

Jade Hochschule
Fachbereich: Bauwesen Geoinformation Gesundheitstechnologie
Studiengang: Angewandte Geodäsie (B. Sc.)



**Webkartenbasierte Visualisierung von Ergebnissen einer Immobilienwertermittlung
mittels Regressionsanalyse nach dem Vergleichswertverfahren**

**Web-based presentation of results of assessment of real estate values determined
via regression analysis**

Bachelorarbeit zur Erlangung des akademischen Grades
„Bachelor of Science“ (B. Sc.)

Eingereicht durch: Clas Christian Borchers



Fachsemester: 7

Erstprüfer: Prof. Dipl.-Ing. Harry Wirth
Jade Hochschule Oldenburg
Ofener Straße 16/19
26121 Oldenburg

Zweitprüfer: Dr.-Ing. Marcel Ziems
Landesamt für Geoinformation und Landesvermessung Niedersachsen
Podbielskistraße 331
30659 Hannover

Oldenburg, 27. Januar 2020

Selbstständigkeitserklärung

Erklärung gemäß § 21 (5) Allgemeiner Teil (Teil A) der Prüfungsordnung für die Bachelor-Studiengänge (BPO) an der Jade Hochschule Wilhelmshaven/Oldenburg/Elsfleth in der Fassung der Bekanntmachung vom 08. Dezember 2004 (VkB1. Nr. 37/2004), zuletzt geändert am 21.10.2014 (VkB1. Nr. 56/2014 vom 24. November 2014).

Die Bachelorarbeit ist eine Einzelarbeit. Ich versichere hiermit, die Bachelorarbeit selbstständig und ohne fremde Hilfe verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt zu haben.

Der digitale Anhang der Bachelorarbeit enthält vertrauliche Informationen, die der Geheimhaltung unterliegen und ist nicht zur Veröffentlichung freigegeben. Ausnahmen können mit Genehmigung des Autors gestattet werden. Die im textlichen Teil der Arbeit enthaltenen vertraulichen Informationen sind unkenntlich gemacht.

Oldenburg, 24.01.2020

Ort, Datum



Unterschrift

Abstract

Die vorliegende Arbeit behandelt die Problemstellung der Visualisierung von Ergebnissen einer Immobilienwertermittlung. Es wird auf die Regressionsanalyse, die dem indirekten Vergleichswertverfahren zugeteilt wird, Bezug genommen. Mit dieser können die Verkehrswerte von verschiedenen Grundstücken bestimmt werden.

Der Grundlagenteil enthält allgemeine Informationen zur Wertermittlung, sowie die für die Umsetzung der Visualisierung erforderlichen Berechnungsschritte und Softwarelösungen. Es werden Methoden zur Klassifizierung von Daten beschrieben, sodass diese in Histogrammen abgebildet werden können. Die Webanwendung wird über das JavaScript-Framework „Angular“ realisiert, dem ein Python-Dienst für die Berechnungen zur Verfügung steht. Der Karteninhalt enthält die in der Ausgleichung verwendeten Kauffallobjekte. Sie sind in Abhängigkeit von der Größe ihres Residuums unterschiedlich eingefärbt. Die Parameter des Bewertungsobjekts, die in die Ausgleichung eingehen, werden in einem beschreibenden Teil der Webanwendung dargestellt. Diesem gegenübergestellt werden die Attribute eines selektierbaren Kauffalls. Es werden die Differenzen zwischen den jeweiligen Attributwerten des Bewertungsobjekts und des Selektionsobjekts berechnet und abgebildet. Außerdem findet die Ermittlung statistischer Kenngrößen statt, welche ebenfalls im beschreibenden Teil der Webanwendung ausgegeben werden. Zusätzlich werden für jedes Attribut die Daten in Graphen dargestellt. Diese enthalten die Histogramme, die Dichtefunktionen der Verteilungen, sowie die Werte des Bewertungs- und des Selektionsobjekts.

Die durchgeführten Methoden stellen eine interaktive Visualisierung der Ergebnisse der Regressionsanalyse dar. Die Objekte lassen sich miteinander vergleichen und durch die Darstellung in Graphen ist der Bezug zum gesamten Datensatz gegeben. Geeignete Kauffälle für die Bestimmung des Vergleichswerts nach dem direkten Vergleichswertverfahren können ermittelt werden. Die Implementierung dieser Berechnungsfunktion in diesem Verfahren ist nicht enthalten.

Inhaltsverzeichnis

Abbildungsverzeichnis	VI
Tabellenverzeichnis	VIII
Codeverzeichnis	IX
1 Einleitung	1
1.1 Motivation	1
1.2 Zielsetzung	1
1.3 Abgrenzung	1
2 Grundlagen	2
2.1 Wertermittlung	2
2.1.1 Kaufpreissammlung	3
2.1.2 Direktes Vergleichswertverfahren: Vergleichspreise und -faktoren	4
2.1.3 Indirektes Vergleichswertverfahren: Regressionsanalyse	6
2.1.4 Exkurs: Attribute der Kauffallobjekte	8
2.1.5 Tool: Automatisiertes Vergleichswertverfahren	8
2.2 Statistik	13
2.2.1 Allgemeine statistische Kenngrößen	13
2.2.2 Diskrete Verteilung	15
2.2.3 Stetige Verteilung	15
2.2.4 Empirische Verteilung	18
2.2.5 Kolmogoroff-Smirnoff-Anpassungstest	19
2.2.6 Empirische Momente	21
2.3 Webanwendung	24
2.3.1 Datengrundlage	25
2.3.2 Vector-Tiles	26
2.3.3 Kartendienst	26
2.3.4 Exkurs: GeoJSON	27
2.3.5 Frontend: Angular	28
2.3.6 Backend-Services	32
3 Konzept	33
4 Durchführung	35
4.1 Berechnung des Vergleichswerts	37

4.2	Darstellung der Kauffälle in der Webkarte	39
4.2.1	Klassifizierung nach Größe des Residuums	40
4.2.2	Erstellen des Layers	41
4.2.3	Erstellen von PopUps	43
4.2.4	Darstellung des Bewertungsobjekts	44
4.3	Darstellung der Parameter des Bewertungsobjekts	45
4.3.1	Verarbeitung in der Komponentenklasse	45
4.3.2	Verarbeitung im Template	45
4.4	Berechnung von statistischen Kenngrößen	47
4.4.1	Verarbeitung im Backend	47
4.4.2	Verarbeitung in der Komponentenklasse	51
4.4.3	Verarbeitung im Template	51
4.5	Darstellung von Histogrammen	53
4.5.1	Verarbeitung in der Komponentenklasse	53
4.5.2	Verarbeitung im Template	54
4.6	Selektion von Objekten	55
4.6.1	Durchführung der Selektion	55
4.6.2	Darstellung des Selektionsobjekts	56
4.6.3	Zurücksetzen der Selektion	59
5	Ergebnisdarstellung und Bewertung	60
5.1	Aufbau der Webanwendung	60
5.2	Vergleich zum Tool „Automatisiertes Vergleichsverfahren“	61
5.2.1	Kartendarstellung	61
5.2.2	Histogramm	62
5.2.3	Selektion	63
6	Fazit	64
7	Ausblick	65
	Literatur	66
A	Report: Bewertung	69
B	Report: Modellbeschreibung	71
C	Digitaler Anhang	73
C.1	Export-Dateien	73
C.2	Quellcode	73

C.3 Beispielvideo	73
C.4 Interne Dokumente	73

Abbildungsverzeichnis

2.1	Kauffalltabellen in der AKS-Datenbank	3
2.2	Vergleichsfaktoren 2015 für Ein- und Zweifamilienhäuser	5
2.3	Ermittlung des Vergleichswerts	5
2.4	Umrechnungskoeffizient Bodenrichtwert	6
2.5	Eingabe der Einflussgrößen	9
2.6	Am besten passende Kauffälle	11
2.7	Visualisierung der Kauffälle in der Karte	11
2.8	Auswirkung von Variationen in den Parametern	12
2.9	Klassifizierung der selektierten Kauffälle	12
2.10	Mittelwert, Modus und Median	14
2.11	Wahrscheinlichkeits- und Verteilungsfunktion einer diskreten Gleichverteilung	15
2.12	Dichte- und Verteilungsfunktion der Standardnormalverteilung	16
2.13	Dichte- und Verteilungsfunktion der Exponentialverteilung	17
2.14	Dichte- und Verteilungsfunktion der Gleichverteilung	18
2.15	K-S-Test am Beispiel von Nüchternblutzucker	21
2.16	Schiefe und Exzess: Abweichungen von der Standardnormalverteilung	23
2.17	Architektur des Prototyps	24
2.18	Verwendete Daten im Prototypen	25
2.19	Screenshot der Webanwendung	25
2.20	Prinzip der Erstellung von Vector-Tiles	26
2.21	Komponenten der Webanwendung	29
2.22	Veränderung des Titels der Webseite	30
3.1	Ablaufdiagramm der durchzuführenden Schritte	33
3.2	Konzeptueller Aufbau der Anwendungskomponenten	34
4.1	Einbinden der ToolStatisticsComponent	36
4.2	Einteilung der Residuen in Klassen	40
4.3	Übermitteln der Daten an die Komponentenklasse „MapboxGlComponent“	41
4.4	Kartendarstellung in verschiedenen Maßstäben	44
4.5	Elementanordnung im Viewport	45
4.6	Ergebnisdarstellung im Overlay	47
4.7	Ablaufdiagramm der Berechnungen im Backend	48
4.8	Beispiel für die Erstellung von Klassen	49
4.9	Darstellung der statistischen Kenngröße am Beispiel des Baujahrs	52
4.10	Darstellung der Graphen am Beispiel des Baujahrs	54
4.11	Inhalt der Selektionsausgabe im Overlay	58

4.12	Darstellung des Graphen nach Einbinden des Selektionsobjekt	59
5.1	Gesamtansicht der Webanwendung mit geöffnetem Statistik-Tool	60
5.2	Darstellung eines Kauffalls mit Flurstücken und Gebäuden	62
5.3	Vergleich der Histogrammdarstellungen	63

Tabellenverzeichnis

2.1	Grundstücksarten	3
2.2	Kritische Werte für den Kolmogoroff-Smirnoff-Test	20
2.3	Eigenschaften der Schiefe	22
2.4	Interpretation des Exzesses	23
2.5	Template-Syntax: Übersicht	30
4.1	Eingabeparameter des Bewertungsobjekts	37
4.2	In der Ausgleichung ermittelte Koeffizienten	38

Codeverzeichnis

2.1	Darstellung des Layers Wald	27
2.2	GeoJSON-Beispiel	27
2.3	Variablenzuordnung in TypeScript	28
2.4	Komponentenklasse des HeaderServices	30
2.5	Interpolation	31
2.6	Property Binding	31
2.7	Event Binding	31
2.8	Strukturdirektive NgIf	31
2.9	Strukturdirektive NgFor	31
2.10	HttpHeaders	32
2.11	„post()“-Methode des HttpClient	32
4.1	Definition des MapTools für die Statistik-Komponente	35
4.2	Zuweisung von Pfad und Komponente	35
4.3	Funktion beim Klick auf Tool	35
4.4	Initialisierung des Python-Dienstes	36
4.5	Funktion zum Übergeben der Geojson-Daten	38
4.6	Funktion zum Übergeben von sonstigen Daten zur weiteren Verarbeitung	39
4.7	Übermitteln der Daten an den Backend-Service	40
4.8	Styling des Layers der Kauffallobjekte	41
4.9	PopUp-Erzeugung bei mouseover	43
4.10	PopUp-Entfernung bei mouseleave	43
4.11	Erstellen des Markers für das Bewertungsobjekt	44
4.12	Initialisierung der Parameter in der Komponentenklasse	45
4.13	Grundgerüst der Anordnung der Card-Elemente	46
4.14	Implementierung der Darstellung des Bewertungsobjekts	46
4.15	Anlegen der Klassen	49
4.16	Berechnung des Werts der empirischen Verteilungsfunktion und der Normalverteilung	50
4.17	Überprüfung auf Überschreitung des Grenzwerts	50
4.18	Berechnung der Dichtefunktion	51
4.19	Zuweisung der im Backend berechneten Daten	51
4.20	Template zur Darstellung der statistischen Kenngrößen	51
4.21	Template zur Darstellung der Schiefe	52
4.22	Festlegen der Optionen zur Darstellung der Graphen	53
4.23	Einbinden der Histogramme im Template	54

4.24	Fallabfrage für das Kennzeichnen des Selektionsobjekts	55
4.25	Färben der Objekte nach Klickselektion	55
4.26	Einbinden einer Funktion in die Ausgabe des Kauffallkennzeichens	56
4.27	Darstellung des selektierten Kauffalls in der Card	57
4.28	Darstellung der Attributwerte und der Differenzen	57
4.29	Anpassung der Optionen der Charts	58
4.30	Button zum Zurücksetzen der Selektion	59
4.31	Funktion zum Zurücksetzen der Selektion	59

1 Einleitung

„Der Wert eines Grundstücks [...] schwankt genauso wie alle anderen Preise und Werte von Handelsgütern in einem marktwirtschaftlichen System“ (BISCHOFF, 2015, S. 495). In der Immobilienwertermittlung werden zur Bestimmung des Verkehrswerts eines Grundstücks verschiedene Verfahren angewandt. Eines ist das Vergleichswertverfahren, welches im indirekten Verfahren das Mittel der Regressionsanalyse darstellt.

1.1 Motivation

Durch das Vorhandensein von Schwankungen in den Ergebnissen der Berechnungen gibt es keine richtige Antwort auf die Frage nach dem tatsächlichen Wert einer Immobilie. Er kann unter Verwendung der anerkannten Verfahren jedoch bestmöglich geschätzt und mit seinen Unsicherheiten angegeben werden. Für die Verbesserung der Ergebnisse sind Objekte zu identifizieren, die dem Bewertungsobjekt möglichst ähnlich sind. Das Finden dieser Kauffallobjekte kann jedoch eine große Schwierigkeit darstellen. Wird die Position betrachtet, fällt auf, dass solche Grundstücke weit von dem zu bewertenden Objekt entfernt liegen können. Es kommt das Problem hinzu, dass die räumliche Lage einen Einfluss auf den Wert haben kann. Für das Lösen dieser Problematik ist eine geeignete Visualisierung hilfreich, die über verschiedenartige Darstellungsformen Hinweise auf geeignete Kauffälle gibt.

1.2 Zielsetzung

Das Ziel dieser Arbeit besteht darin, die Daten, die für die durchzuführende Immobilienwertermittlung notwendig sind, sowie die Ergebnisse der Ausgleichung, in einer Webkarte zu visualisieren. Der Vorteil der Webkarte liegt in der dynamischen Anpassung der darzustellenden Inhalte. Durch eine optimale Visualisierung der Thematik im Karteninhalt und der berechneten Parameter in einem beschreibenden Teil, können weitere Ziele, wie die Suche nach konkreten Kauffällen, verfolgt werden. Dies kann eine Berechnung des Grundstückswerts nach dem direkten Vergleichswertverfahren optimieren.

1.3 Abgrenzung

Die Berechnungen zur Ermittlung des Verkehrswerts eines Bewertungsobjekts werden im Rahmen dieser Arbeit nicht in die Anwendung integriert. Da sie jedoch die Grundlage der durchzuführenden Visualisierung bilden, werden sie in dem bestehenden Tool „Automatisiertes Vergleichswertverfahren“, das an der Technischen Universität Dresden entwickelt wurde, durchgeführt.

2 Grundlagen

Im Kapitel „Grundlagen“ werden die für die Arbeit erforderlichen Verfahren, Methoden und Anwendungen vorgestellt und erläutert. Es wird auf die Grundsätze der Wertermittlung eingegangen, sowie ein konkretes Tool zur Berechnung vorgestellt. Ebenfalls Erwähnung findet die bestehende Webkarte, in der die Visualisierung umzusetzen ist, sowie die statistischen Grundlagen, welche für die Bestimmung von Kenngrößen erforderlich sind.

2.1 Wertermittlung

Die Wertermittlung ist ein Organ zur Bestimmung von Verkehrswerten verschiedenartiger Grundstücke. Gemäß § 194 BAUGB (Baugesetzbuch) handelt es sich beim Verkehrswert um den Wert des Grundstückes, welches es hätte, würde es im üblichen Handel verkauft werden. Es wird unter anderem auf die Aspekte Bezug genommen, in welchem Zustand es sich befindet, wo es räumlich liegt und welche rechtliche Faktoren eine übergeordnete Rolle spielen (§ 194 BAUGB). Den zeitlichen Bezugspunkt stellt der Wertermittlungstichtag dar. Dieser besitzt aufgrund der sich stetig ändernden Marktverhältnisse eine große Relevanz. Er ist in der Immobilienwertermittlungsverordnung (ImmoWertV) verankert und beschreibt den Zeitpunkt, „auf den sich die Wertermittlung bezieht“ (§ 3 ABS. 1 IMMOWERTV). Die zu diesem Tag, der entweder in der Vergangenheit oder in der Gegenwart liegen kann (SOMMER und KRÖLL, 2013, S. 93), „vorherrschenden allgemeinen Wertverhältnisse auf dem Grundstücksmarkt sollen gemäß § 3 ImmoWertV maßgeblich sein“ (SOMMER und KRÖLL, 2013, S. 93). Die Verfahren zur Bestimmung des Verkehrswerts lassen sich in drei Schemata einordnen. Diese sind je nach Art und Beschaffenheit des jeweiligen Grundstücks anzuwenden. Entscheidend ist, ob es sich um bebaute oder unbebaute Grundstücke handelt und ob substanzwert- oder renditeorientierte Faktoren einen Einfluss haben (SIMON und GILICH, 2012, S. 146). Es handelt sich um das Vergleichswertverfahren, das Sachwertverfahren und das Ertragswertverfahren.

Die Erstellung von Verkehrswertgutachten kann von verschiedenen Institutionen durchgeführt werden: von freien Sachverständigen für Grundstückswerte oder von öffentlich bestellten und vereidigten Sachverständigen (BISCHOFF, 2015, S. 501). Dieses erhalten im Gegensatz zu den freien Sachverständigen ihren Titel von einer öffentlich-rechtlichen Kammer und müssen einen prüfungsähnlichen Nachweis und langjährige Berufserfahrung vorweisen (BISCHOFF, 2015, S. 501). Im Weiteren existieren, basierend auf einer Initiative der Europäischen Union, zur einheitlichen Bezeichnung und Kontrolle im EU-Raum, die sogenannten zertifizierten Sachverständigen (BISCHOFF, 2015, S. 501). Der Gutachterausschuss ist ein weiteres zuständiges Organ für die Erstellung von Verkehrswertgutachten (§ 193 BAUGB). Gemäß § 9 ABS. 1 DVO-BAUGB (Niedersächsische Verordnung zur Durchführung des Baugesetzbuches) sind in Niedersachsen im Bereich

jeder Vermessungs- und Katasterbehörde Gutachterausschüsse für Grundstückswerte (GAG) als Landesbehörden zu bilden. Sie haben zusätzlich die Aufgabe die Kaufpreissammlung zu führen und die für die Wertermittlung erforderlichen Daten zu ermitteln (§ 193 Abs. 5 BAUGB). Diese setzen sich aus den Vergleichsfaktoren, den Sachwertfaktoren, den Liegenschaftszinssätzen, sowie den Umrechnungskoeffizienten zusammen (§ 193 Abs. 5 BAUGB).

2.1.1 Kaufpreissammlung

Die laut Baugesetzbuch dem Gutachterausschuss zugeordnete Aufgabe, die Kaufpreissammlung zu führen, wird dadurch realisiert, dass jeder vertragliche Übergang von Eigentum an Grundstücken dem Gutachterausschuss mitzuteilen ist (§ 195 Abs. 1 BAUGB). Dies beinhaltet neben entgeltlichen Kaufverträgen auch solche, die auf der Basis einer Schenkung existieren (§ 195 Abs. 1 BAUGB). Das Ziel ist es, die Vorgänge des Grundstücksverkehrs zu sammeln und auszuwerten (BISCHOFF, 2015, S. 514).

Die **Automatisierte Kaufpreissammlung (AKS)** der Vermessungs- und Katasterverwaltung des Landes Niedersachsen realisiert diese Aufgabe, indem die erfassten Kauffälle in einer Datenbank gespeichert werden (LGLN, 2019b, S. 15). Die folgende Abbildung (2.1) zeigt einen grundlegenden Ausschnitt aus dem Datenbank-Schema der AKS.

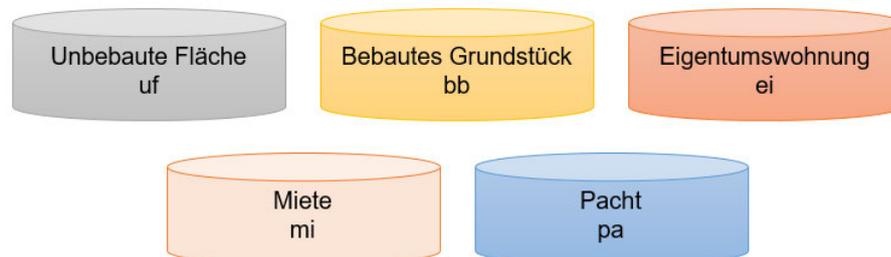


Abbildung 2.1: Kauffalltabellen in der AKS-Datenbank (LGLN, 2019b, S. 15)

In diese fünf übergeordneten Kategorien lassen sich weitere Teilgruppen einordnen:

Name der Grundstücksart	Kurzbezeichnung
bebautes Grundstück	bb
Eigentumswohnung	ei
unbebaute Flächen, unterteilt in:	uf
unbebaute Baufläche	ub
land- und forstwirtschaftliche Fläche	lf
Gemeindebedarfsfläche	gf
sonstige Fläche	sf
Miete	mi
Pacht	pa

Tabelle 2.1: Grundstücksarten (LGLN, 2019b, S. 17).

2.1.2 Direktes Vergleichswertverfahren: Vergleichspreise und -faktoren

Für die Immobilienwertermittlung, also der Bestimmung der Verkehrswerte bebauter Grundstücke, sowie für die Ermittlung des Werts von unbebauten Grundstücken, ist das Vergleichswertverfahren als ein mögliches Verfahren zu nennen. Die Richtlinie zur Ermittlung des Vergleichswerts und des Bodenwerts, kurz Vergleichswertrichtlinie (VW-RL), regelt aufbauend auf der ImmoWertV die Ermittlung des Vergleichswerts und des Bodenwerts (ZIFF 1. ABS. 1 VW-RL). Sie definiert verschiedene Begriffsbezeichnung und skizziert Ablaufstrukturen. Das Vergleichswertverfahren lässt sich in zwei Kategorien gliedern. Neben dem direkten Verfahren, welches als Voraussetzung die konkrete Vergleichbarkeit der Kaufpreise nutzt, kann das indirekte Verfahren angewendet werden (BISCHOFF, 2015, S. 590).

Im direkten Vergleichswertverfahren wird ebenfalls unterschieden: zwischen (1) der Berechnung des Vergleichswerts mittels in ausreichender Anzahl vorliegender Vergleichspreise, sowie (2) der Ermittlung durch geeignete Vergleichsfaktoren (ZIFF. 7 ABS. 1 VW-RL).

(1) Ein **Vergleichspreis** ist der Kaufpreis eines Grundstücks, welches mit dem Wertermittlungsobjekt zum Wertermittlungsstichtag in hinreichend vielen wertbeeinflussenden Grundstücksmerkmalen übereinstimmt (ZIFF. 3 ABS. 1 VW-RL). Diese Merkmale können unter anderem das Baujahr, die Art und die Größe des Gebäudes, der bauliche Zustand oder die Restnutzungsdauer sein (SOMMER und KRÖLL, 2013, S. 31). Nach ZIFF. 3 ABS. 1 VW-RL sind diejenigen Kaufpreise nicht als Vergleichspreise heranzuziehen, die sich in ihrem Wert deutlich von anderen Kaufpreisen mit denselben Eigenschaften unterscheiden. Es handelt sich in solchen Fällen um eine Beeinflussung in ungewöhnlichen oder persönlichen Verhältnissen. Zur Eignung wird der Einfluss der Merkmale auf den Kaufpreis überprüft (ZIFF. 3 ABS. 2 VW-RL). Es werden nur diejenigen Kaufobjekte als Vergleichsobjekte verwendet, bei denen die abweichenden Merkmale einen unerheblichen Einfluss auf den Wert haben, oder bei denen Merkmalsabweichungen berücksichtigt werden können, sodass eine Anpassung möglich ist (ZIFF. 3 ABS. 2 VW-RL).

(2) **Vergleichsfaktoren** sind gemäß ZIFF. 6 ABS. 1 VW-RL Durchschnittswerte eines Normobjekts, also eines Kauffalls mit definierten Grundstücksmerkmalen. Es wird sich auf eine Einheit bezogen (ZIFF. 6 ABS. 1 VW-RL), über die durch Multiplikation der Verkehrswert abgeleitet werden kann (ZIFF. 7 ABS. 1 VW-RL).

Die Abbildung 2.2 zeigt den Vergleichsfaktor in der Einheit $\text{€} / \text{m}^2$ Wohnfläche für den Bereich Friesland und Ostfriesland (ohne Inseln) an. Dieser ist für ein Normobjekt mit den Eigenschaften einer Bodenrichtwertzone von $50 \text{ €} / \text{m}^2$, einer Grundstücksgröße von 1000 m^2 und der

Standardstufe in Abhängigkeit vom Baujahr angegeben. Außerdem handelt es sich um ein nicht unterkellertes Gebäude, welches ein ausgebautes Dachgeschoss besitzt. Die Kurven im Graphen zeigen die Vergleichsfaktoren des jeweiligen modifizierten Baujahres in Abhängigkeit zur Wohnfläche.

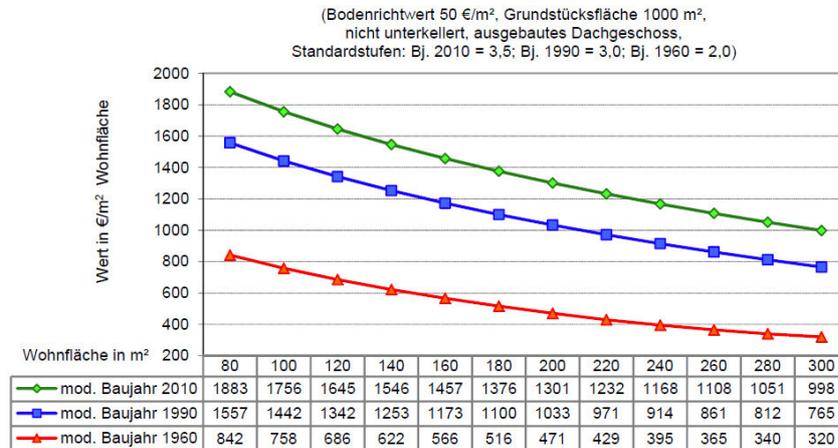
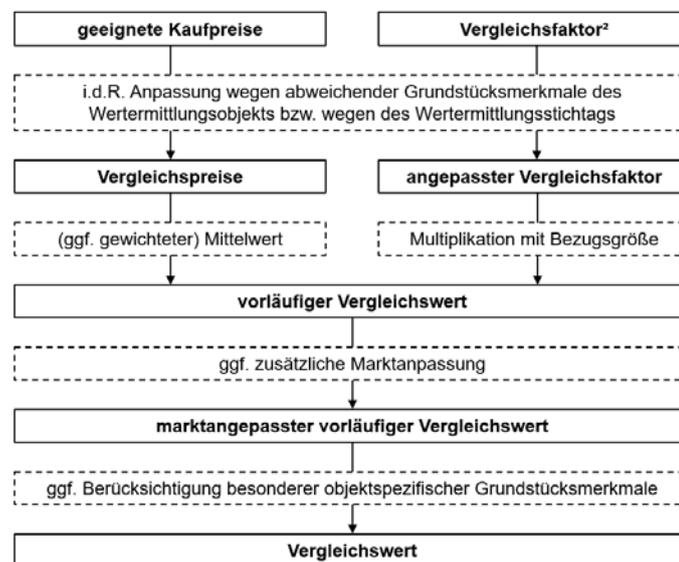


Abbildung 2.2: Vergleichsfaktoren 2015 für Ein- und Zweifamilienhäuser (GAG, 2016, S. 62)

In Abbildung 2.3 wird der Ablauf des direkten Vergleichswertverfahrens mit Vergleichspreisen (links) und mit Vergleichsfaktoren (rechts) schematisch beschrieben.



²bzw. Bodenrichtwert im Rahmen der Bodenwertermittlung

Abbildung 2.3: Ermittlung des Vergleichswerts (ZIFF. 7 ABS. 4 VW-RL)

Unabhängig von der Wahl des Verfahrens werden im ersten Schritt die Kaufpreise bzw. der Faktor an die Abweichungen des Wertermittlungsobjekts an die vorliegenden Werte angepasst, sodass entweder Vergleichspreise oder ein angepasster Vergleichsfaktor entsteht. Findet die Bestimmung des Vergleichswerts über die Vergleichspreise statt, ist an dieser Stelle der Mittelwert zu bilden. Bei der Berechnung über den Weg des Vergleichsfaktors wird nun die Multiplikation mit der

Bezugsgröße durchgeführt, sodass in beiden Fällen der vorläufige Vergleichswert entsteht. Dieser kann bei Bedarf noch mit zusätzlichen Umrechnungskoeffizienten optimiert werden.

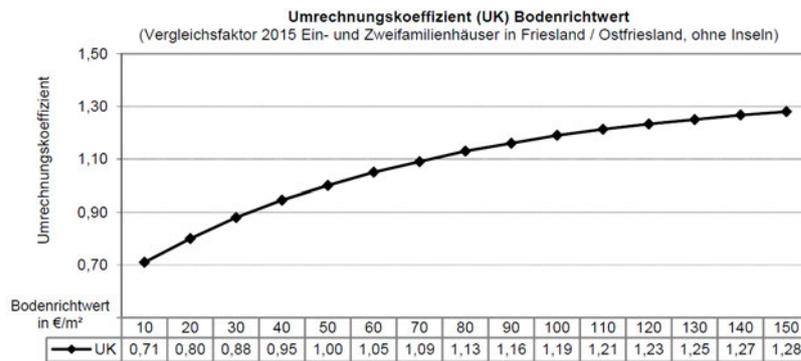


Abbildung 2.4: Umrechnungskoeffizient Bodenrichtwert (GAG, 2016, S. 62)

Eine solche Marktanpassung kann zum Beispiel die des Bodenrichtwerts, also dem Preis pro Quadratmeter Grundfläche, sein. Stimmt dieser nicht zwischen dem Merkmal des Normobjekts und dem Wertermittlungsobjekt überein, so ist er anzupassen (Abbildung 2.4). Der Wert des Umrechnungskoeffizienten des tatsächlichen Bodenrichtwerts ist mit dem des Normobjekts ins Verhältnis zu setzen. Dieser Quotient ist als Faktor bei der Bestimmung des Verkehrswerts anzubringen. Es ergibt sich der marktangepasste vorläufige Vergleichswert. Abschließend sind noch besondere objektspezifische Grundstücksmerkmale durch Zu- oder Abschläge zu berücksichtigen (ZIFF. 8 ABS. 1 VW-RL). Laut ZIFF. 8 VW-RL können dies zum Beispiel Baumängel, Bauschäden, Bodenverunreinigungen oder wirtschaftliche Überalterung sein. Ist diese Anpassung durchgeführt worden, ist das Ergebnis der Vergleichswert des bebauten bzw. unbebauten Grundstücks.

2.1.3 Indirektes Vergleichsverfahren: Regressionsanalyse

Soll ein Vergleichswert über das indirekte Verfahren bestimmt werden, sind die wertbestimmenden Parameter festzustellen und nachzuweisen (BISCHOFF, 2015, S. 590). Eine Möglichkeit einer solchen Feststellung und daraus möglichen Berechnung ist die Regressionsanalyse, die mittels linearen oder nicht linearen Modellen, sowie unter Verwendung einer oder mehrerer Einflussgrößen, durchgeführt werden kann (BISCHOFF, 2015, S. 590). Die gesuchte Größe bei der Berechnung kann zum Beispiel der Kaufpreis pro Quadratmeter Wohnfläche sein (BISCHOFF, 2015, S. 590). Wird der Ansatz der multiplen linearen Regression verwendet, werden zwei oder mehr Regressoren in der Regressionsgleichung angesetzt (ECKSTEIN, 2012, S. 336). Die Form der Geradengleichung lautet:

$$y_i = \beta_0 + \beta_1 \cdot x_{i1} + \beta_2 \cdot x_{i2} + \dots + \beta_k \cdot x_{ik} + \epsilon_i \quad i = 1, \dots, n \quad (2.1)$$

(MITTAG, 2012, S. 245).

Die Zielgröße y wird, da es sich bei der Regressionsanalyse um ein statisches Modell handelt, in der Ausgleichung geschätzt (URBAN und MAYERL, 2011, S. 17). Der Parameter β_0 ist die Regressionskonstante (BISCHOFF, 2015, S. 593). Bei den Parametern $\beta_1, \beta_2, \dots, \beta_k$ handelt es sich um die Regressionskoeffizienten, die abhängigen Variablen drücken sich in den Parametern x_1, x_2, \dots, x_k aus (BISCHOFF, 2015, S. 593). Außerdem liegt eine Störvariable vor, die mit ϵ bezeichnet wird (MITTAG, 2012, S. 245). Den Regressoren wird unterstellt, dass sie „untereinander nicht korreliert sind“ (ECKSTEIN, 2012, S. 336). Dieser funktionale Zusammenhang stellt das mathematische Modell der Ausgleichung dar. Werden die Attribute der einzelnen Kauffälle als Beobachtungswerte in die Formel eingesetzt, ergibt sich ein Gleichungssystem mit n Gleichungen (MITTAG, 2012, S. 245). Unter Verwendung von Matrizen lautet die Formel des Modells:

$$y = X\beta + \epsilon \quad (2.2)$$

(MITTAG, 2012, S. 246).

Die Ausgleichung soll bewirken, dass die Störgrößen $\hat{\epsilon} = y - \hat{y}$ (Residuen) minimal sind (SACHS und HEDDERICH, 2006, S. 566). Es wird nach der Methode der kleinsten Quadrate verfahren (MITTAG, 2012, S. 248). Die Minimierungsaufgabe lautet:

$$\sum_{i=1}^n \hat{\epsilon}_i = \sum_{i=1}^n (y_i - \hat{\beta}_0 - \hat{\beta}_1 x_{i1} - \hat{\beta}_2 x_{i2} - \dots - \hat{\beta}_k x_{ik})^2 \rightarrow Min. \quad (2.3)$$

(MITTAG, 2012, S. 249).

Über die Ableitung dieser Funktion nach β lässt sich das Ausgleichungsproblem lösen, sodass sich unter Verwendung der Hilfsmatrix $H = X(X'X)^{-1}X'$ folgende Zusammenhänge ergeben:

$$\begin{aligned} \hat{\beta} &= (X'X)^{-1}y && \text{Schätzung für } \beta && (2.4) \\ \hat{y} &= Hy = X\hat{\beta} && \text{Schätzung für } y \\ \hat{\epsilon} &= y - X\hat{\beta} = y - \hat{y} = (I - H)y && \text{Schätzfehler} \\ \hat{\epsilon}'\hat{\epsilon} &= y'(I - H)y && \text{Summe der Abweichungsquadrate,} \\ &&& \text{RSS (residual sum of squares)} \end{aligned}$$

(SACHS und HEDDERICH, 2006, S. 567).

Für die Schätzung der Varianz wird folgender Ansatz verwendet:

$$\hat{\sigma}^2 = \frac{RSS}{n - k - 1} \quad (2.5)$$

(SACHS und HEDDERICH, 2006, S. 567).

Der Erwartungswert $E(\epsilon)$ ergibt als Folgerung der Ausgleichung den Wert „0“ und die Varianz wird über $Var(\epsilon) = \sigma^2 I$ ausgedrückt (SACHS und HEDDERICH, 2006, S. 567). Wird die Normalverteilung mit den Parametern μ und σ betrachtet, ist diese proportional zu den Residuen selbst:

$$\epsilon \sim N(0, \sigma^2 I) \quad (2.6)$$

(SACHS und HEDDERICH, 2006, S. 567).

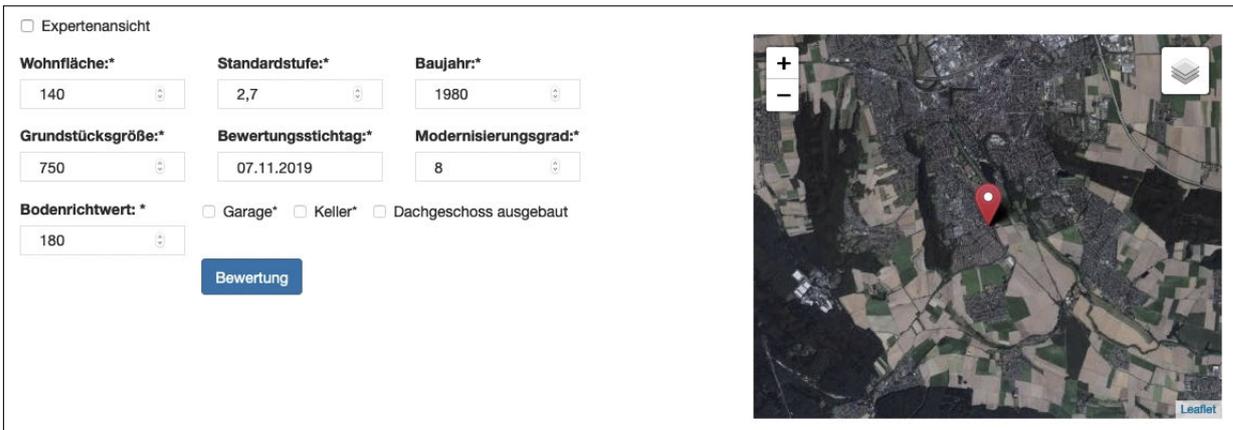
2.1.4 Exkurs: Attribute der Kauffallobjekte

Im Modell der NHK 2010, welches die Normalherstellungskosten bezogen auf das Basisjahr 2010 widerspiegelt (SOMMER und KRÖLL, 2013, S. 55), wird bei den jeweiligen Gebäudearten zwischen verschiedenen Standardstufen unterschieden (SOMMER und KRÖLL, 2013, S. 57). Gemäß ZIFF. 4.1.1.2 ABS. 2 SW-RL (Richtlinie zur Ermittlung des Sachwerts, kurz Sachwertrichtlinie) sind bei der Einordnung des Objekts in eine dieser Stufen neben der Qualität der verwendeten Baumaterialien u. a. auch energietechnische Eigenschaften von Bedeutung. Die Standardstufe ist ein Maß für die Qualität des Bauwerks. Da den Bauwerken eine „gleichmäßige Wertminderung zugrunde zu legen“ (§ 23 IMMOWERTV) ist, wird die Alterswertminderung durchgeführt. Sie gibt den zeitlichen Aspekt des Wertverlustes wieder und wird über das Verhältnis der Restnutzungsdauer zur Gesamtnutzungsdauer bestimmt (§ 23 IMMOWERTV). Die Gesamtnutzungsdauer beschreibt den Zeitraum, in welchem das Objekt, entsprechend seiner Zweckbestimmung, theoretisch nutzbar ist (SOMMER und KRÖLL, 2013, S. 68). Sie ist in der Anlage 3 der Sachwertrichtlinie beschrieben und wird je nach Art des Gebäudes unterschiedlich angegeben. Ein freistehendes Einfamilienhaus mit einer Ausstattung, die der Standardstufe 3 entspricht, hat hiernach eine Gesamtnutzungsdauer von 70 Jahren (ANLAGE 3 SW-RL). Die Restnutzungsdauer ist die Differenz aus der Gesamtnutzungsdauer und dem Alter des Gebäudes. Werden zum Beispiel Modernisierungen durchgeführt, kann die Restnutzungsdauer des Objekts verlängert werden (BISCHOFF, 2015, S. 658). In diesem Fall wird von der modifizierten Restnutzungsdauer gesprochen. Dessen konkreter Wert ist abhängig vom Modernisierungsgrad. In der Anlage 4 der Sachwertrichtlinie ist das Modell zur Berücksichtigung der durchgeführten Baumaßnahmen beschrieben. Für verschiedene Arten an Modernisierungselementen werden Punkte vergeben, anhand derer der Zustand in die Modernisierungsgrade 1 (nicht modernisiert) bis 5 (umfassend modernisiert) eingeteilt wird (ANLAGE 4 SW-RL).

2.1.5 Tool: Automatisiertes Vergleichswertverfahren

Das Tool „Automatisiertes Vergleichswertverfahren“, welches an der Technischen Universität Dresden entwickelt wurde, dient der Bestimmung von Vergleichswerten (SOOT und WEITKAMP, 2019, S. 2).

Das zu bewertende Objekt wird einem Teilmarkt (Einfamilienhäuser, Doppelhaushälften, Reihenhäuser oder Eigentumswohnungen) zugeordnet, auf dessen Basis der Vergleichswert im hedonischen Modell und über Vergleichspreise bestimmt wird (SOOT und WEITKAMP, 2019, S. 2). Das hedonische Modell ist ein Verfahren zur Immobilienbewertung unter Verwendung von realen Transaktionsdaten, für die der durchschnittliche Quadratmeterpreis bestimmt wird (DEMARY, 2009, S. 3). In diesem Modell wird der Vergleichswert über die hedonische Regression bestimmt, bei der ein funktionaler Zusammenhang zwischen dem Immobilienpreis und den Eigenschaften der Immobilie eingeführt wird (DEMARY, 2009, S. 4). Dieses Verfahren wird entsprechend der Regressionsanalyse angewandt.



The image shows a web interface for entering influence factors. On the left, there is a form with the following fields and values:

- Expertenansicht
- Wohnfläche*: 140
- Standardstufe*: 2,7
- Baujahr*: 1980
- Grundstücksgröße*: 750
- Bewertungsstichtag*: 07.11.2019
- Modernisierungsgrad*: 8
- Bodenrichtwert*: 180
- Garage*
- Keller*
- Dachgeschoss ausgebaut

A blue button labeled 'Bewertung' is located below the form. To the right of the form is a satellite map with a red location pin and a 'Leaflet' logo in the bottom right corner.

Abbildung 2.5: Eingabe der Einflussgrößen (EIGENE)

Die Definition des Bewertungsobjekts kennzeichnet sich durch die Angabe der Lage, die Auswahl des Teilmarktes, sowie der in Abbildung 2.5 aufgezeigten Grundstücksmerkmale (SOOT und WEITKAMP, 2019, S. 4 f.). Die für die Berechnung herangezogenen Objekte werden im ersten Schritt in einer Selektionsauswahl zusammengefasst (SOOT und WEITKAMP, 2019, S. 11). Von den möglichen Grundstücken wird der Wertabstand zum Bewertungsobjekt berechnet, der sich aus den Abweichungen aller Einflussfaktoren zusammensetzt (SOOT und WEITKAMP, 2019, S. 11). Eine veränderliche Selektionsschranke sorgt dafür, dass der Vorgang iterativ abläuft, da in jedem Schritt der Betrag vergrößert wird, bis der Umfang der Stichprobe mindestens 300 Kauffälle umfasst (SOOT und WEITKAMP, 2019, S. 12). Mit dieser Auswahl wird anschließend die Regressionsanalyse durchgeführt, dessen zu untersuchende Parameter sich zum Beispiel für den Teilmarkt Einfamilienhäuser aus folgenden Größen zusammensetzt: Wohnfläche, Baujahr, Datum des Vertrages, Bodenrichtwert, Restnutzungsdauer, Grundstücksfläche, Modernisierungsgrad, Standardstufe, Garage und Balkon (SOOT und WEITKAMP, 2019, S. 13 f.). Es werden für die Durchführung nur diejenigen Parameter verwendet, die in mehr als 60 % der Fälle belegt sind (SOOT und WEITKAMP, 2019, S. 14).

Zur Überprüfung, ob die Parameter in der Regressionsanalyse signifikant bestimmt wurden, wird der Student-Test (auch t-Test) herangezogen (SOOT und WEITKAMP, 2019, S. 15). Dieser erlaubt es, einzelne Koeffizienten gezielt zu überprüfen (SACHS und HEDDERICH, 2006, S. 572). Über die Teststatistik

$$\hat{t}_i = \frac{\hat{\beta}_i}{se(\hat{\beta}_i)} \quad (2.7)$$

(SACHS und HEDDERICH, 2006, S. 572),

mit se als Standardfehler des Koeffizienten, kann die Nullhypothese $H_0 : \beta_i = 0$ (SACHS und HEDDERICH, 2006, S. 572) überprüft werden. Ist der Freiheitsgrad $f \geq 200$, gilt ein Wert von 1,97 als kritischer Wert mit einer Irrtumswahrscheinlichkeit von 5 % (SOOT und WEITKAMP, 2019, S. 15). Ist der Betrag von \hat{t}_i größer als der kritische Wert, kann die Nullhypothese abgelehnt und die Alternativhypothese ($H_A : \beta_i \neq 0$ (SACHS und HEDDERICH, 2006, S. 572)) angenommen werden (SOOT und WEITKAMP, 2019, S. 15). In diesem Fall ist der Parameter „mit hoher Sicherheit signifikant bestimmt“ (SOOT und WEITKAMP, 2019, S. 15). Die Ermittlung von Ausreißern wird mit Hilfe der Cook-Distanz durchgeführt (SOOT und WEITKAMP, 2019, S. 18). Es werden zwischen den auf den Einzelbeobachtungen basierenden Werten \hat{y}_i und dem ausgeglichenen Wert \hat{y} Differenzen gerechnet, die anschließend quadriert und mit der geschätzten Varianz $\hat{\sigma}^2$ gewichtet werden (FAHRMEIR et al., 2009, S. 178). Außerdem geht der Parameter k in die Gewichtung ein, der ein Maß für die Anzahl der Koeffizienten darstellt. Die Formel zur Berechnung der Cook-Distanz lautet:

$$D_i = \frac{(\hat{y}_{(i)} - \hat{y})'(\hat{y}_{(i)} - \hat{y})}{k * \hat{\sigma}^2} \quad (2.8)$$

(FAHRMEIR et al., 2009, S. 178).

Die im Tool verwendete Funktion zur Ausreißerererkennung wendet dieses Verfahren iterativ an (SOOT und WEITKAMP, 2019, S. 18). Die Kauffälle mit auffällig hohen Cook-Distanzen werden anhand der Quantile solange eliminiert, bis die folgende Bedingung erfüllt ist:

$$Max_{D_i} < D_{i_{q75\%}} + 3 \cdot D_{i_{IQR}} \quad (2.9)$$

(SOOT und WEITKAMP, 2019, S. 18).

In der Anwendung dargestellt wird das Ergebnis der Regressionsanalyse in einem Feld mit der Bezeichnung „Vergleichswert hedonisches Modell“. Es wird neben dem absoluten Wert auch der Kaufpreis pro Quadratmeter Wohnfläche angegeben.

Eine Auflistung von insgesamt zehn Objekten mit ihren Attributen wird tabellarisch dargestellt (Abbildung 2.6). Über den ermittelten Wertabstand wird geprüft, um welche Kauffälle es sich dabei handelt (SOOT und WEITKAMP, 2019, S. 19): je kleiner der Wertabstand ist, „desto besser

passt das Vergleichsobjekt zum Bewertungsobjekt“ (SOOT und WEITKAMP, 2019, S. 19). Von diesen Kauffällen können mehrere markiert werden, was zur Folge hat, dass das Feld „Vergleichswert aus Objekten“ den Mittelwert der aktuellen Selektion berechnet und ausgibt (SOOT und WEITKAMP, 2019, S. 19).

id	Gemeinde	Straße	Bodenrich...	Standards...	Wohnfläche	Baujahr	Datum	Kaufpreis/...
056bb20161362	Itzum	Sensburger Ring	175	2	140	1982	2016-12-28	1.900
056bb20161351	Itzum	Elsa-Brändströ...	175	2.5	150	1985	2016-11-10	1.800
056bb20150475	Hildesheim	Arndtstraße	185	2.5	150	1981	2015-02-06	1.800
056bb20171324	Itzum	Am Nottbohm	165	3.6	135	1985	2017-09-28	1.800
056bb20150481	Ochtersum	An der Renne	180	2.9	155	1994	2015-02-10	1.800
056bb20150551	Ochtersum	Plötzenstraße	160	2.8	150	1978	2015-04-09	1.800
056bb20150541	Ochtersum	Am Wallhof	170	2.5	160	1980	2015-03-05	1.800
056bb20160719	Hildesheim	Neidenburger ...	170	2	143	1963	2016-08-30	1.800
056bb20150494	Ochtersum	Wunramstraße	170	2.4	154	1962	2015-07-07	1.800
056bb20150120	Hildesheim	Halberstädter ...	170	2.5	135	1960	2015-03-06	1.800

Abbildung 2.6: Am besten passende Kauffälle (EIGENE)

Die grafische Ausgabe (Abbildung 2.7) der Daten beschränkt sich auf eine Hintergrundkarte. Neben dem Wertermittlungsobjekt, das durch einen roten Marker gekennzeichnet ist, sind die selektierten Objekte der Ausgabetabelle durch blaue Marker dargestellt.

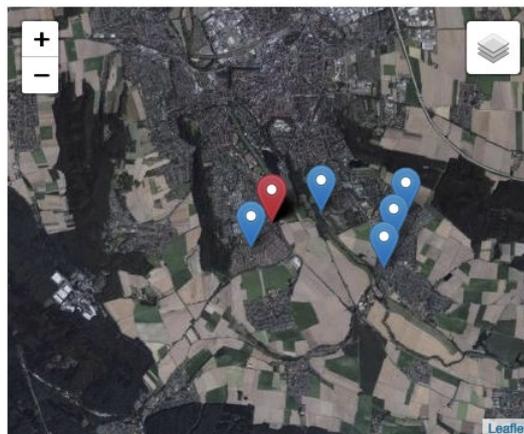


Abbildung 2.7: Visualisierung der Kauffälle in der Karte (EIGENE)

Zusätzliche Ergebnisdokumente (Bewertung und Modell) können für die durchgeführte Analyse erstellt werden. Diese sind, neben den folgenden Auszügen, in Anhang A und B abgebildet.

(1) Der Report **Bewertung** stellt einen Überblick der Ergebnisse mit den zugehörigen Unsicherheiten dar. Es findet eine tabellarische Auflistung der Vergleichskauffälle statt, die für die Bestimmung des Vergleichswerts herangezogen wurden. Zusätzlich wird eine Berechnung durchgeführt und abgebildet, die die Auswirkungen von Variationen in den Parametern beschreibt (SOOT und WEITKAMP, 2019, S. 20). Den Werten des Vergleichsobjekts wird ein Zu- und ein Abschlag aufgetragen und mit diesem der entsprechende Vergleichswert bestimmt. Tendenzen

für Veränderungen in den Parametern auf den Vergleichswert werden verdeutlicht. Dargestellt ist dies in der Abbildung 2.8.

Parameter	Baujahr	Bodenrichtwert [€/qm]	Restnutzungsdauer [a]	Modernisierungsgrad	Standardstufe
Ausprägung	2000	176	49	7	2.2
Vergleichswert	281000	238000	199000	222000	237000
Ausprägung	1980	144	40	5	1.8
Vergleichswert	172000	215000	249000	231000	216000

Abbildung 2.8: Auswirkungen von Variationen in den Parametern (EIGENE)

(2) Der Report **Modellbeschreibung** stellt die Dokumentation des Analyseprozesses dar. In einer Tabelle werden die Koeffizienten, die aus der Ausgleichung hervorgehen, abgebildet. Sie beschreiben das Regressionsmodell. Im weiteren werden Histogramme der Einflussgrößen visualisiert (SOOT und WEITKAMP, 2019, S. 20). Die Daten beziehen sich auf die durchgeführte Selektion. Es handelt sich um mindestens 300 zu klassifizierende und abzubildende Kauffälle. Zu diesen werden, neben der reinen Darstellung, für jedes Attribut die Werte Minimum, Maximum, Mittelwert und Median bestimmt. Die Abbildung 2.9 zeigt die Art Visualisierung der Histogramme und der berechneten Werte im Report.

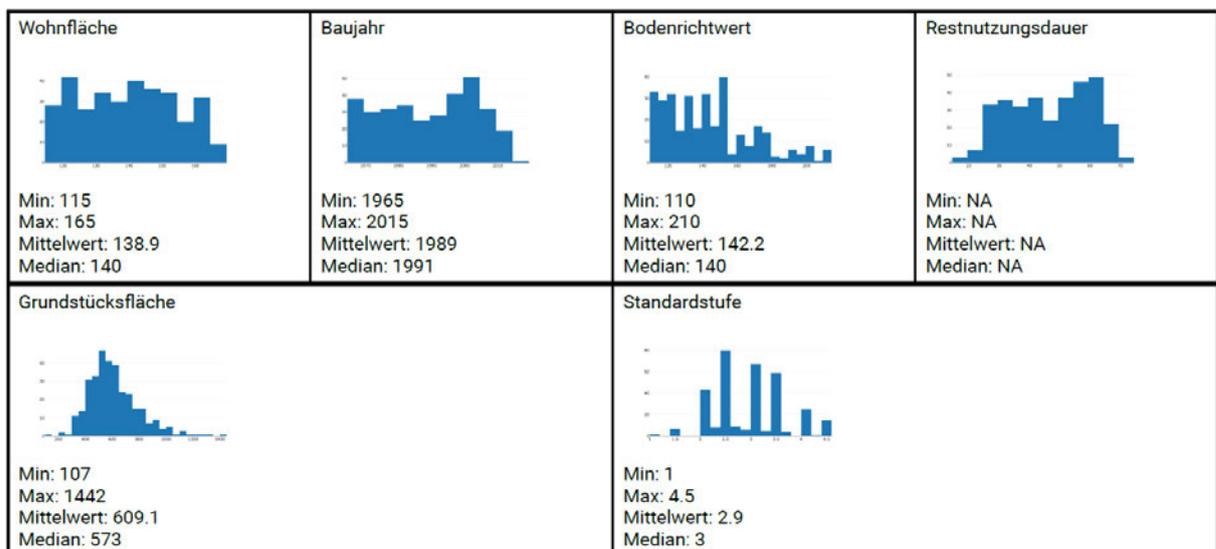


Abbildung 2.9: Klassifizierung der selektierten Kauffälle (EIGENE)

Abschließend lassen sich drei JSON-Dateien exportieren. Die erste beinhaltet die gesamte Kauffallselektion. Die zweite Datei repräsentiert die in der Ausgleichung verwendeten Kauffälle und gibt diese im GeoJSON-Format mit ihren Attributen, sowie den Ergebnissen der Ausgleichung (Residuen und Cook-Distanzen) an. Eine Beschreibung des GeoJSON-Formates ist in Kapitel 2.3.4 zu finden. In der dritten Datei sind die Parameter der Ausgleichung aufgeführt. Dies sind zum Beispiel die Regressionskoeffizienten oder die Anzahl der Iterationsschritte.

Die Dateien sind im digitalen Anhang C.1 abgelegt.

2.2 Statistik

Werden Daten erhoben und sollen diese auf statistische Merkmale überprüft werden, lassen sich neben den allgemeinen statistischen Kenngrößen auch die empirischen Momente angeben. Außerdem kann die Art der Verteilung von empirischen Daten auf Übereinstimmung mit einer theoretischen Verteilung geprüft werden, was durch den Kolmogoroff-Smirnoff-Anpassungstest ermöglicht wird.

2.2.1 Allgemeine statistische Kenngrößen

Für statistisch auszuwertende Datensätze lassen sich unter anderem vier Kenngrößen bestimmen: der Mittelwert, der Modus, der Median und die Varianz.

(1) Der **arithmetische Mittelwert** \bar{x} gibt die durchschnittliche Größe der Beobachtungswerte wieder. Er berechnet sich nach der Funktion:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.10)$$

(SACHS und HEDDERICH, 2006, S. 68).

(2) Der **Modus** D , auch Modalwert, bezeichnet die häufigste Merkmalsausprägung (SACHS und HEDDERICH, 2006, S. 56). In einem Datensatz können mehrere Beobachtungswerte dem Modus zugeordnet werden (BÖSELT, 1999, S. 47).

(3) Der **Median** \tilde{x} , auch Zentralwert (BÖSELT, 1999, S. 32), der Beobachtungsreihe spezifiziert sich dadurch, dass er, im Gegensatz zum Mittelwert, die tatsächliche mittlere Lage des sortierten Datensatzes berücksichtigt (BÖSELT, 1999, S. 32). Somit ist dieser Wert, bei einer ungeraden Länge der Reihe, der tatsächlich erhobene Wert einer Beobachtung. Ist die Anzahl der Elemente gerade, wird das Mittel der beiden zentralen Werte bestimmt und für den Wert des Medians angegeben (SACHS und HEDDERICH, 2006, S. 63). Es wird sichergestellt, dass beide Beobachtungshälften (links und rechts des Medians) 50% der Elemente erhalten (SACHS und HEDDERICH, 2006, S. 63). Der Median \tilde{x} ist definiert durch:

$$\tilde{x} = \begin{cases} x_{\frac{n+1}{2}}^* & \text{falls } n \text{ ungerade} \\ \frac{1}{2}(x_{\frac{n}{2}}^* + x_{\frac{n}{2}+1}^*) & \text{falls } n \text{ gerade} \end{cases} \quad (2.11)$$

(BÖSELT, 1999, S. 32).

Die Werte x_i^* stellen die Elemente x_i in nach Größe sortierter Reihenfolge dar (BÖSELT, 1999, S. 32).

Die folgende Grafik zeigt beispielhaft den Zusammenhang von Mittelwert, Modus und Median an einem Datensatz ($n = 80$) auf. Der Median und der Modus nehmen den gleichen Wert (2) an. Der Median ist nach Definition bei $n = 80$ Elementen an der 40,5. Stelle der geordneten Datenreihe. Der zugeordnete Wert liegt ebenfalls an der Stelle 2. Der Mittelwert der Daten liegt bei 2,5 und befindet sich somit rechts neben dem Median und dem Modus auf der X-Achse.

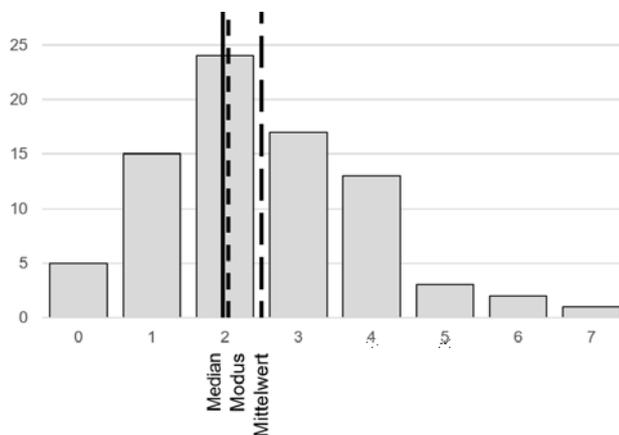


Abbildung 2.10: Mittelwert, Modus und Median (EIGENE)

(4) Für die Berechnung der **empirischen Varianz** ist die Information des Mittelwerts erforderlich. Aus diesem kann mit dem aktuellen Wert x_i der Reihe die Verbesserung $v_i = \bar{x} - x_i$ (GRUBER und JOECKEL, 2009, S. 163) berechnet werden. Die quadrierten und anschließend aufsummierten Verbesserungen werden mit der Anzahl der Stichproben ins Verhältnis gesetzt. Es ergibt sich folgender funktionaler Zusammenhang:

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n (v_i)^2 \quad (2.12)$$

(BÖSELT, 1999, S. 36).

Der Subtrahend von -1 ergibt sich an dieser Stelle, da der Mittelwert \bar{x} als Schätzwert aus dem arithmetischen Mittel angenommen wird (GRUBER und JOECKEL, 2009, S. 163). Die positive Wurzel der Varianz wird außerdem als **empirische Standardabweichung** s bezeichnet:

$$s = +\sqrt{s^2} \quad (2.13)$$

(BÖSELT, 1999, S. 36).

Die Standardabweichung und die Varianz sind Maße für die Streuung der Daten (SACHS und HEDDERICH, 2006, S. 70). Da die Summe der Verbesserungen (v_i) Null ergäbe, werden diese quadriert (SACHS und HEDDERICH, 2006, S. 70), sodass das Vorzeichen in jedem Fall positiv ist und sich die Verbesserungen nicht gegenseitig aufheben.

2.2.2 Diskrete Verteilung

Ist eine Verteilung diskret, so lässt sich „die Anzahl ihrer Ausprägungen abzählen“ (MITTAG, 2012, S. 173). Die Menge der Ausprägungen ist auf endlich viele Einheiten begrenzt, das Maximum liegt bei abzählbar unendlich vielen Ausprägungen (MITTAG, 2012, S. 149). Dies bedeutet, dass den Merkmalen Klassen zugeordnet werden, die einen konkreten (diskreten) Wert besitzen. Die Gleichverteilung besitzt die Eigenschaft, dass sie an den definierten Stellen die gleiche Wahrscheinlich besitzt:

$$P(X = k) = \frac{1}{m} \quad \text{für } k = 1, 2, \dots, m \quad (2.14)$$

(SACHS und HEDDERICH, 2006, S. 163).

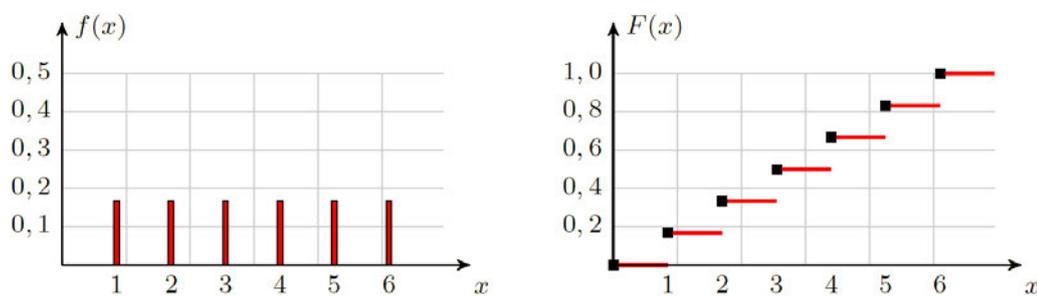


Abbildung 2.11: Wahrscheinlichkeits- und Verteilungsfunktion einer diskreten Gleichverteilung (MITTAG, 2012, S. 151)

Die Abbildung 2.11 zeigt eine **Gleichverteilung** am Beispiel des Würfels. In der Wahrscheinlichkeitsfunktion $f(x)$ (links) ist die Wahrscheinlichkeit der einzelnen Ausprägungen ($p = \frac{1}{6}$) abgetragen. Der rechte Teil der Grafik zeigt die Verteilungsfunktion $F(x)$, die durch Aufsummieren der Einzelwahrscheinlichkeiten entsteht (SACHS und HEDDERICH, 2006, S. 146). Sie ist definiert als:

$$F(x) = \sum_{x_i \leq x} f(x_i) \quad (2.15)$$

(SACHS und HEDDERICH, 2006, S. 147).

2.2.3 Stetige Verteilung

Kann die Zufallsgröße X in hinreichend kleinen Intervallen jeden Wert annehmen und ist dessen Verteilungsfunktion außerdem durch eine reelle Dichtefunktion $f(t) \geq 0$ mit $t \in \mathbb{R}$ gegeben, so wird sie stetig genannt (ECKSTEIN, 2012, S. 164). Die Dichtefunktion $f(t)$ ist diejenige Funktion, über die mittels Integration Wahrscheinlichkeiten bestimmt werden können (SACHS und HEDDERICH, 2006, S. 147). Diese nimmt keine negativen Werte an (MITTAG, 2012, S. 174) und besitzt außerdem die Eigenschaft, dass das Integral $\int_{-\infty}^{+\infty} = 1$ ist (MITTAG, 2012, S. 174).

Der Zusammenhang von Dichte- und Verteilungsfunktion wird durch die Definition der Verteilungsfunktion beschrieben:

$$F(x) = P(X \leq x) = \int_{-\infty}^x f(t) dt \quad (2.16)$$

(SACHS und HEDDERICH, 2006, S. 147).

Die Wahrscheinlichkeit, dass die Zufallsvariable X einen Wert im Intervall $a < X \leq b$ annimmt, wird durch folgenden Zusammenhang angegeben:

$$P(a < X \leq b) = F(b) - F(a) = \int_a^b f(t) dt \quad (2.17)$$

(SACHS und HEDDERICH, 2006, S. 147).

Normalverteilung

Die Normalverteilung ist durch die Parameter des Erwartungswerts μ und der Standardabweichung σ gekennzeichnet. Diese sind maßgebend für die Lage und die Form der Kurve. Die Notation wird in der Form $N(\mu, \sigma)$ angegeben. Die Gleichung der Dichtefunktion lautet:

$$f(x) = \frac{1}{\sigma \cdot \sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2} \quad (2.18)$$

(BECKER, 1993, S. 258).

Das Maximum der Kurve befindet sich an der Stelle $x = \mu$ und die Wendepunkte befinden sich jeweils an den Stellen $\mu - \sigma$ und $\mu + \sigma$, sodass sich eine sogenannte Glockenkurve ergibt (SACHS und HEDDERICH, 2006, S. 192). Die Abbildung 2.12 zeigt die Dichtefunktion (links), sowie die Verteilungsfunktion der Standardnormalverteilung $N(0, 1)$. Neben den Funktionen ist außerdem jeweils an der Stelle $x = -0,8$ der zugehörige Wert der Verteilungsfunktion ($F(-0,8)$) dargestellt.

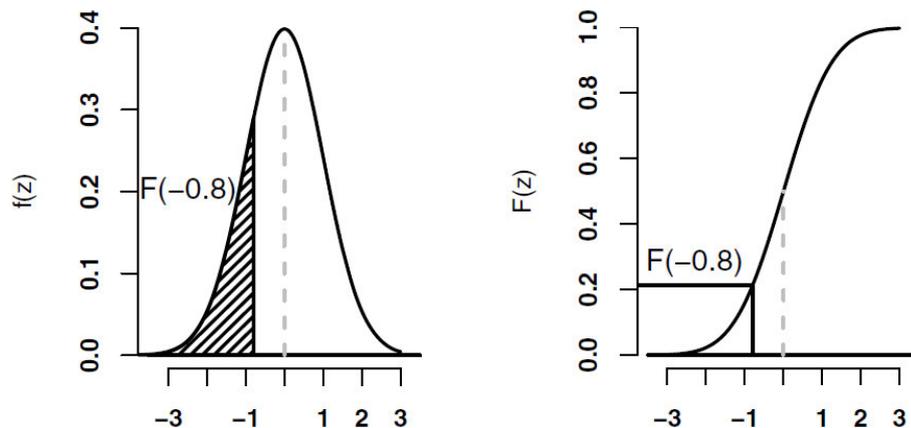


Abbildung 2.12: Dichte- und Verteilungsfunktion der Standardnormalverteilung $N(0, 1)$ (SACHS und HEDDERICH, 2006, S. 194)

Exponentialverteilung

Die Exponentialverteilung für eine stetige Zufallsgröße X wird über den Parameter λ gesteuert (ECKSTEIN, 2012, S. 190). Die Dichtefunktion ergibt sich durch:

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases} \quad (2.19)$$

(SACHS und HEDDERICH, 2006, S. 207).

Die Parameter μ und λ stehen in folgendem Zusammenhang:

$$\mu = \frac{1}{\lambda} \quad (2.20)$$

(SACHS und HEDDERICH, 2006, S. 208).

Dargestellt werden in Abbildung 2.13 drei verschiedene Exponentialfunktionen mit $\lambda = 1; 5; 10$. Links sind die Dichtefunktionen aufgezeigt, rechts die Verteilungsfunktionen.

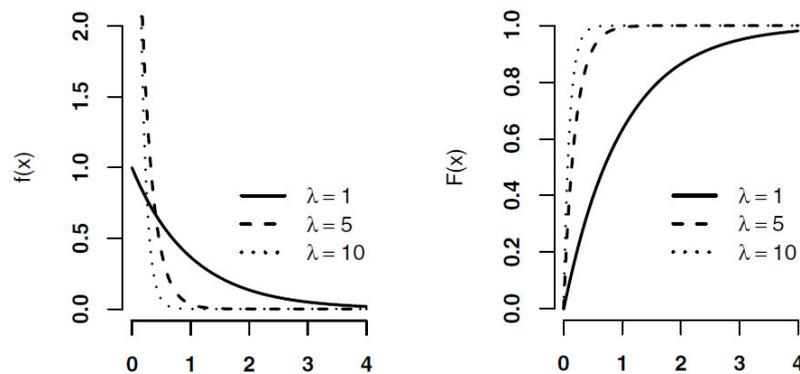


Abbildung 2.13: Dichte- und Verteilungsfunktion der Exponentialverteilung (SACHS und HEDDERICH, 2006, S. 208)

Die Verteilungsfunktion definiert sich nach:

$$F(x) = P(X \leq x) = \begin{cases} 1 - e^{-\lambda x} & \text{für } x \geq 0 \\ 0 & \text{für } x < 0 \end{cases} \quad (2.21)$$

(SACHS und HEDDERICH, 2006, S. 208).

Gleichverteilung

Die stetige Gleichverteilung wird durch die Parameter a und b beschrieben:

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{für } a < x < b \\ 0 & \text{für } x \leq a \text{ oder } x \geq b \end{cases} \quad (2.22)$$

(SACHS und HEDDERICH, 2006, S. 190).

Bestimmt werden können die Werte für μ und σ^2 aus dem Zusammenhang mit a und b :

$$\begin{aligned} \text{Erwartungswert : } \mu &= \frac{a+b}{2} \\ \text{Varianz : } \sigma^2 &= \frac{(b-a)^2}{12} \end{aligned} \quad (2.23)$$

(SACHS und HEDDERICH, 2006, S. 191).

Die Abbildung 2.14 zeigt links die Dichtefunktion $f(x)$, sowie rechts die Verteilungsfunktion $F(x)$. Die Gleichverteilung wird auch Rechteckverteilung genannt (MITTAG, 2012, S. 174). Diese ist nach der Formel 2.22 definiert. Da der Quotient $\frac{1}{b-a}$ von x unabhängig ist und nur für den Bereich $a < x < b$ bestimmt wird, ergibt sich, dass der Funktionswert $f(x)$ im definierten Bereich konstant ist. Außerhalb dieses Bereiches ist er laut Definition „0“.

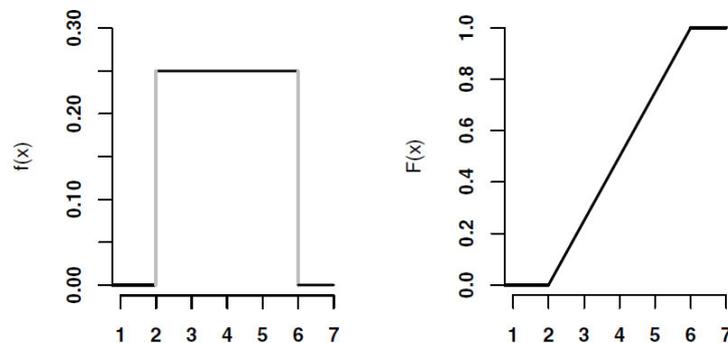


Abbildung 2.14: Dichte- und Verteilungsfunktion der Gleichverteilung (SACHS und HEDDERICH, 2006, S. 191)

Für die Verteilungsfunktion bedeutet dies, dass der Funktionswert im Bereich von Werten x_i , die kleiner als a sind, gleich „0“ und für Werte, die größer als b sind, gleich „1“ sein muss. Der Bereich zwischen a und b verläuft linear. Ausgedrückt wird der Verlauf der Kurve in folgender Funktion:

$$F(x) = \begin{cases} 0 & \text{für } x < a \\ \frac{x-a}{b-a} & \text{für } a \leq x \leq b \\ 1 & \text{für } x > b \end{cases} \quad (2.24)$$

(MITTAG, 2012, S. 175).

2.2.4 Empirische Verteilung

Sind Daten erhoben worden und sollen diese ausgewertet werden, wird von einer empirischen Verteilung gesprochen. Diese kann entweder durch die absolute Häufigkeit

$$h_i := h(a_i) \quad i = 1, 2, 3, \dots, k \quad (2.25)$$

(MITTAG, 2012, S. 38)

oder durch die relative Häufigkeit angegeben werden (MITTAG, 2012, S. 52 f.):

$$f_i := f(a_i) = \frac{h(a_i)}{n} \quad i = 1, 2, 3, \dots, k \quad (2.26)$$

(MITTAG, 2012, S. 39)

Die absolute Häufigkeit ist die ganzzahlige Anzahl der in einem Stichprobenumfang der Größe n erhobenen Werte in einer Urliste (ECKSTEIN, 2012, S. 72). Bei der relativen Häufigkeit wird dieser Wert anschließend mit dem Stichprobenumfang ins Verhältnis gesetzt (ECKSTEIN, 2012, S. 72). Visualisiert werden können diese Werte zum Beispiel in einem Balkendiagramm, oder bei stetigen Merkmalen in einem Histogramm (MITTAG, 2012, S. 53). Es werden k äquidistante Klassen erstellt (MITTAG, 2012, S. 53), denen die Einzelmessungen zugeordnet werden. Die Klassenbreite h_n der einzelnen Klassen ist mit Zuhilfenahme der Standardabweichung σ über

$$h_n = 3,49 \frac{\sigma}{\sqrt[3]{n}} \quad (2.27)$$

(SCOTT, 1979, S. 608)

zu bestimmen. Da diese Art der Berechnung außerdem die Größe des Stichprobenumfangs n berücksichtigt, wird eine optimale Breite der Klassen bestimmt, sodass neben dem Informationsgehalt auch die Lesbarkeit und die Interpretierbarkeit der dargestellten Daten erhalten bleibt.

Die empirische Verteilungsfunktion wird mit

$$E_n = \frac{n(i)}{n} \quad (2.28)$$

(SACHS und HEDDERICH, 2006, S. 337)

angegeben. Der Wert $n(i)$ spiegelt die Anzahl derjenigen Beobachtungen wider, die kleiner als die aktuell betrachtete Beobachtung ist (SACHS und HEDDERICH, 2006, S. 337). Es wird also eine Kumulation durchgeführt.

2.2.5 Kolmogoroff-Smirnoff-Anpassungstest

Liegen erhobene Daten vor und soll überprüft werden, ob diese einer theoretischen stetigen Verteilung (z. B. Normalverteilung, Exponentialverteilung oder Gleichverteilung) zugeordnet werden können, empfiehlt sich der Kolmogoroff-Smirnoff-Anpassungstest (K-S-Test) (SACHS und HEDDERICH, 2006, S. 337). Dieser prüft zu einem vorher festgelegten Signifikanz-Niveau α , ob das theoretische Verteilungsmodell als Beschreibung für die unbekannte Verteilung der Stichprobe herangezogen werden kann (ECKSTEIN, 2012, S. 253).

Es sind zu den jeweiligen Signifikanzniveaus Schranken (Grenzwerte) für die Testgröße D unter Zuhilfenahme des Umfangs der Stichprobe tabelliert:

Schranken für D	Signifikanzniveau α
$1,037/\sqrt{n}$	0,20
$1,138/\sqrt{n}$	0,15
$1,224/\sqrt{n}$	0,10
$1,358/\sqrt{n}$	0,05
$1,517/\sqrt{n}$	0,02
$1,628/\sqrt{n}$	0,01

Tabelle 2.2: Kritische Werte für den Kolmogoroff-Smirnoff-Test (SACHS und HEDDERICH, 2006, S. 338)

Sind Stichproben im Umfang von $n > 35$ vorhanden, so kann die Tabelle 2.2 als Bezug für die Annahme oder Ablehnung der Nullhypothese verwendet werden (SACHS und HEDDERICH, 2006, S. 338). Die Nullhypothese H_0 beinhaltet die Aussage, dass die überprüften Daten der gewählten Verteilung entsprechen. Dagegen sagt die Alternativhypothese H_A aus, dass die Daten der Verteilung nicht folgen (SACHS und HEDDERICH, 2006, S. 337).

Zur Berechnung der Testgröße \hat{D} werden im ersten Schritt die Elemente der Stichprobe der Größe nach sortiert, sodass sich die Werte x_i ergeben (BEHREND, 2013, S. 346). Anschließend wird die Differenz aus dem Wert der theoretischen Verteilungsfunktion an der Stelle x_i und dem entsprechenden Wert der empirischen Verteilung, welche in der Formel mit $\frac{i}{N}$ beschrieben wird, berechnet, wobei das i die kumulierte Anzahl der sortierten Daten darstellt. Von dieser Differenz wird daraufhin der Betrag genommen. Dieses Verfahren wird mit sämtlichen Daten durchgeführt, sodass sich am Ende ein Maximum für den gesamten Datensatz ergibt. Dieses wird mit dem kritischen Wert (s. Tabelle 2.2) verglichen und es wird überprüft, ob dieser den zulässigen Wert übersteigt (SACHS und HEDDERICH, 2006, S. 337). Ist dies der Fall, wird die Nullhypothese abgelehnt und die Alternativhypothese angenommen (SACHS und HEDDERICH, 2006, S. 338). Die konkrete Formel der Teststatistik lautet:

$$\hat{D} = \max_{1 \leq i \leq N} |F(x_i) - \frac{i}{N}| \quad (2.29)$$

(SACHS und HEDDERICH, 2006, S. 337).

Die folgende Abbildung (2.15) zeigt ein Beispiel der Überprüfung einer Stichprobe auf eine Verteilung. Hier wurde getestet, ob die erhobenen Daten (Nüchternblutzuckerwerte) normalverteilt sind. Die stetige Kurve stellt die Verteilungsfunktion der Normalverteilung $N(\mu, \sigma)$ dar. Zusätzlich zu erkennen ist eine Treppenfunktion: die empirische Verteilungsfunktion. An den Punkten der Klassenmitte wird die Differenz zur Verteilungsfunktion der Normalverteilung bestimmt und mit der Teststatistik \hat{D} geprüft.

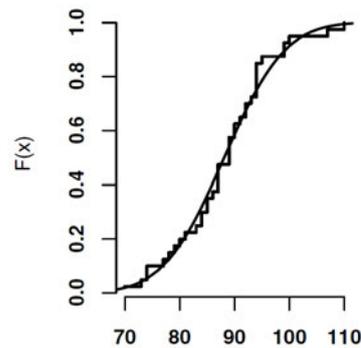


Abbildung 2.15: K-S-Test am Beispiel von Nüchternblutzucker [mg/dl] (SACHS und HEDDERICH, 2006, S. 338)

2.2.6 Empirische Momente

Für die qualitative Beschreibung von Werten einer stetigen oder diskreten Zufallsvariable sind die Funktionalparameter (SACHS und HEDDERICH, 2006, S. 150), „die bestimmte Eigenschaften einer Wahrscheinlichkeitsverteilung kennzeichnen“ (SACHS und HEDDERICH, 2006, S. 150), heranzuziehen. Diese sind zum Beispiel die Lage, die Streuung und die Form (SACHS und HEDDERICH, 2006, S. 150). Ausgedrückt werden diese Parameter in den vier Momenten der Verteilung: (1) Erwartungswert, (2) Varianz, (3) Schiefe und (4) Exzess. Diese leiten sich aus den Potenzen X^k ab, sodass der Erwartungswert dieser Potenzen k -tes Moment der Zufallsvariablen genannt wird (SACHS und HEDDERICH, 2006, S. 154). Es gilt:

$$\mu'_k = E[X^k] \quad (2.30)$$

(SACHS und HEDDERICH, 2006, S. 154).

Wird eine Abweichung auf den Erwartungswert bezogen, wird von einem zentralen Moment gesprochen (SACHS und HEDDERICH, 2006, S. 154).

1. Moment: Erwartungswert

Für das erste Moment ($k = 1$) ergibt sich also, dass $\mu'_1 = E[X] = \mu$ ist, was dem Erwartungswert entspricht (SACHS und HEDDERICH, 2006, S. 154). Dieser gibt ein Maß für den Ausgang des Zufallsexperimentes an (SACHS und HEDDERICH, 2006, S. 152).

Er wird durch folgende Gleichung definiert:

1. Erwartungswert für diskrete Zufallsvariablen X : $E[X] = \mu = \sum_{i=1}^n x_i \cdot f(x_i)$ (2.31)
2. Erwartungswert für stetige Zufallsvariablen X : $E[X] = \mu = \int_{-\infty}^{+\infty} x \cdot f(x) dx$

(SACHS und HEDDERICH, 2006, S. 151).

Die Funktion f entspricht der Dichtefunktion.

2. zentrales Moment: Varianz

Mit Hilfe der Varianz soll eine zu erwartende Streuung im Ausgang des Zufallsexperiments beschrieben werden (SACHS und HEDDERICH, 2006, S. 152). Die Definitionen für die Varianz bei diskreten und stetigen Zufallsvariablen lauten:

1. Varianz für diskrete Zufallsvariablen X : $Var[X] = \sigma^2 = \sum_{i=1}^n (x_i - \mu)^2 \cdot f(x_i)$ (2.32)
2. Varianz für stetige Zufallsvariablen X : $Var[X] = \sigma^2 = \int_{-\infty}^{+\infty} (x - \mu)^2 \cdot f(x) dx$

(SACHS und HEDDERICH, 2006, S. 152).

3. zentrales Moment: Schiefe

Über die Schiefe können Angaben zur „Richtung und Größenordnung der Asymmetrie einer Verteilung“ (STRUNZ, 2012, S. 218) getroffen werden. Ist der Wert der Schiefe (γ_1) größer als 0, handelt es sich um positive Schiefe (BÖSELT, 1999, S. 49). Diese Art der Schiefe wird auch linkssteil genannt (BÖSELT, 1999, S. 49). Rechtssteile Verteilungen kennzeichnen sich durch $\gamma_1 < 0$ (SACHS und HEDDERICH, 2006, S. 155). Ist der Wert $\gamma_1 = 0$, wird von einer symmetrischen Verteilung gesprochen (SACHS und HEDDERICH, 2006, S. 154). Die Formeln zur Berechnung der Schiefe bei diskreten und stetigen Verteilungen sind definiert durch:

1. Schiefe für diskrete Zufallsvariablen X : $\gamma_1 = \frac{\sum_{i=1}^n (x_i - \mu)^3 \cdot f(x_i)}{\sigma^3}$ (2.33)
2. Schiefe für stetige Zufallsvariablen X : $\gamma_1 = \frac{\int_{-\infty}^{+\infty} (x - \mu)^3 \cdot f(x) dx}{\sigma^3}$

(SACHS und HEDDERICH, 2006, S. 154).

Die Division mit σ^3 geschieht, da so die „dimensionslose Kennzeichnung der Schiefe“ (SACHS und HEDDERICH, 2006, S. 154) ermöglicht werden kann.

Die Schiefe besitzt folgende Zusammenhänge mit dem Mittelwert, dem Modus und dem Median:

γ_1	Art der Verteilung	Modus (D) und Mittelwert (\bar{x})	Median (\tilde{x}) und Mittelwert (\bar{x})	Modus (D) und Median (\tilde{x})
> 0	linkssteile Verteilung	$D < \bar{x}$	$\tilde{x} < \bar{x}$	$D < \tilde{x}$
$= 0$	symmetrische Verteilung	$D = \bar{x}$	$\tilde{x} = \bar{x}$	$D = \tilde{x}$
< 0	rechtssteile Verteilung	$D > \bar{x}$	$\tilde{x} > \bar{x}$	$D > \tilde{x}$

Tabelle 2.3: Eigenschaften der Schiefe (SACHS und HEDDERICH, 2006, S. 156)

4. zentrales Moment: Exzess

Über den Exzess als viertes zentrales Moment kann eine Aussage über die Überhöhung oder die Stauchung der Verteilung getroffen werden (STRUNZ, 2012, S. 219). Er ist ein Maß für die

Wölbung bzw. die Steilheit der Kurve (STRUNZ, 2012, S. 219). Die Formeln zur Berechnung des Exzesses lauten:

$$\begin{aligned} 1. \text{ Exzess für diskrete Zufallsvariablen } X : \quad \gamma_2 &= \frac{\sum_{i=1}^n (x_i - \mu)^4 \cdot f(x_i)}{\sigma^4} - 3 & (2.34) \\ 2. \text{ Exzess für stetige Zufallsvariablen } X : \quad \gamma_2 &= \frac{\int_{-\infty}^{+\infty} (x - \mu)^4 \cdot f(x) dx}{\sigma^4} - 3 \end{aligned}$$

(SACHS und HEDDERICH, 2006, S. 155).

Auch dieser Parameter wird mit σ^k dividiert, was zu einer dimensionslose Angabe des Werts führt (SACHS und HEDDERICH, 2006, S. 155). Die Subtraktion um -3 erfolgt, da dies der Wert des Exzesses der Normalverteilung ist, die als Referenz angegeben wird (SACHS und HEDDERICH, 2006, S. 155). So wird für die Wölbung direkt ein Maß für die Abweichung zur normalverteilten Glockenkurve ermittelt.

Die nachstehende Tabelle zeigt die verschiedenartigen Wölbungen, ausgedrückt durch den Exzess γ_2 , an.

γ_2	Art der Wölbung	Bezeichnung der Wölbung
> 0	stark gewölbt	leptokurtisch
$= 0$	mittelmäßig gewölbt	mesokurtisch
< 0	flach gewölbt	platykurtisch

Tabelle 2.4: Interpretation des Exzesses (SACHS und HEDDERICH, 2006, S. 156)

Für den Fall, dass es sich um eine symmetrische Verteilung handelt, sind die Werte des Mittelwerts, des Modus und des Medians identisch, sodass gilt: $\bar{x} = D = \tilde{x}$ (BÖSELT, 1999, S. 50). Ist die Verteilung dagegen unsymmetrisch, so liegt der Mittelwert \bar{x} gegenüber dem Median \tilde{x} in Richtung des „Tals“ der Verteilung verschoben (BÖSELT, 1999, S. 50). Der Modus liegt weiterhin an der maximalen Ausprägung der Kurve (BÖSELT, 1999, S. 50).

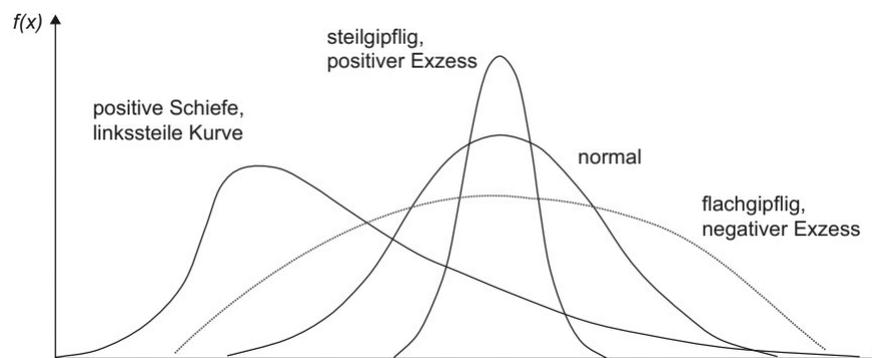


Abbildung 2.16: Schiefe und Exzess: Abweichungen von der Standardnormalverteilung (STRUNZ, 2012, S. 218)

Die Abbildung 2.16 zeigt im Vergleich zur Standardnormalverteilung drei unterschiedlich ausgeprägte Kurven. Auf der linken Seite ist eine Verteilung mit positiver Schiefe zu sehen, dessen Modus weiter links als der der Standardnormalverteilung liegt. Zentral, aber mit höherem Wert $f(x)$, befindet sich eine steilgipflige Kurve. Eine im Vergleich zur normalverteilten Glockenkurve spitzere Verteilung, wie sie hier vorliegt, nennt sich auch positiv exzessiv (BÖSELT, 1999, S. 49). Rechts abgebildet (gestrichelt) befindet sich eine Kurve mit negativem Exzess. Sie ist deutlich flacher gewölbt als die anderen Verteilungen.

2.3 Webanwendung

Die Projektgruppe „Basisvisualisierung II“ des Landesamtes für Geoinformation und Landesvermessung Niedersachsen (LGLN) hat mit dem Ziel, eine „Webanwendung zur individuellen Anpassung der Vektorkarte“ (LGLN, 2019a, S. 4) zu erstellen, die Aufgabe, einen Prototypen zu implementieren (LGLN, 2019a, S. 4). Dieser Prototyp soll auf den Daten des AAA-Modells basieren (AFIS-ALKIS-ATKIS) und Vector-Tile-Dienste mit Kartenstylings für die Darstellung verwenden (LGLN, 2019a, S. 4).

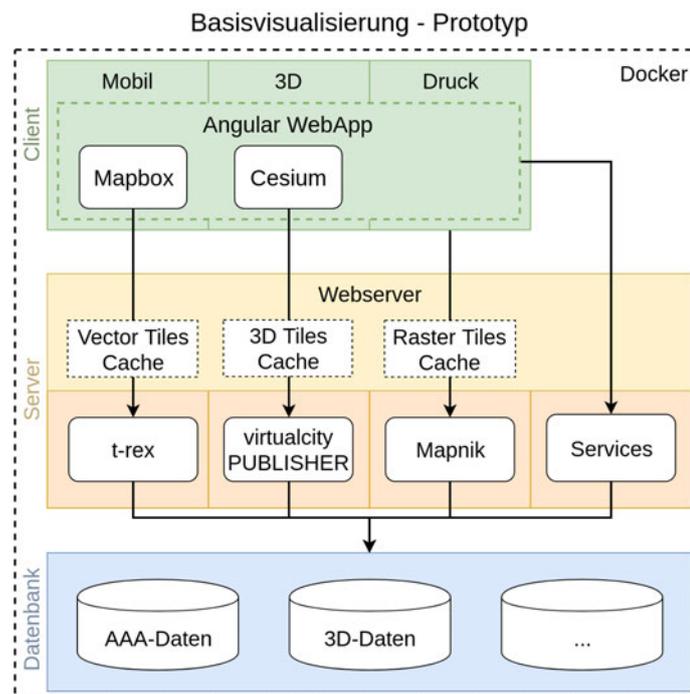


Abbildung 2.17: Architektur des Prototyps (ZIEMS et al., 2019, S. 11)

Die Abbildung 2.17 zeigt die Architektur des Prototyps und visualisiert die Zusammenhänge der einzelnen Bestandteile. Der Teilbereich „Mobil“, auf den sich hier konzentriert wird, wird durch die Verwendung des Dienstes „Mapbox“ am Client realisiert. Als Server wird der „t-rex-Tileserver“ verwendet und die Datenbank besteht u. a. aus den Daten des AAA-Modells.

2.3.1 Datengrundlage

Die darzustellenden Daten bestehen aus Teilbereichen ATKIS (Amtliches Topographisch-Kartographisches Informationssystem) und ALKIS (Amtliches Liegenschaftskatasterinformationssystem), sowie aus den Kauffällen der AKS. Die ATKIS-Daten ergeben durch die flächendeckende Nutzung die Kartengrundlage. Aus dem Bereich ALKIS werden Flurstücke und Gebäude eingebunden. Die AKS-Daten sind eine Ergänzung zu den Daten des AAA-Modells und geben die Lage von Grundstückskäufen an. Die Reihenfolge der dargestellten Layer ist der Abbildung 2.18 zu entnehmen.

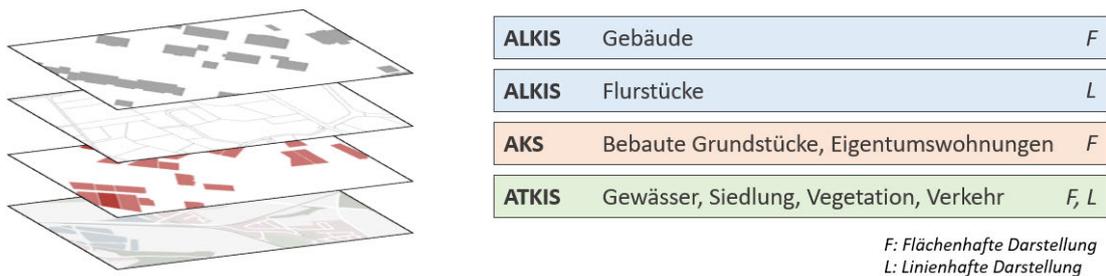


Abbildung 2.18: Verwendete Daten im Prototypen (EIGENE)

Die Abbildung 2.19 zeigt das Ergebnis der Darstellung der Layer, sowie den Aufbau der Webanwendung. Bei den Gebäuden handelt es sich um 3D-Gebäude des Detaillierungsgrads Level of Detail (LoD) 1. Bei diesem sogenannten „Klötzchenmodell“ wird für alle Gebäude ein Flachdach dargestellt (ADV, 2018, S. 33).



Abbildung 2.19: Screenshot der Webanwendung (EIGENE)

Das verwendete Datenbankmanagementsystem ist „PostgreSQL“ mit der Erweiterung „PostGIS“, welche es ermöglicht, räumliche Daten zu speichern und zu verarbeiten (POSTGIS, 2019, o. S.).

2.3.2 Vector-Tiles

Bei den in der Datenbank gespeicherten Objekten handelt es sich um Vektordaten. Werden diese in Kartendiensten dargestellt, wird der gesamte Datensatz geladen, also auch jener Teil, der außerhalb des betrachteten Sichtfensters liegt (GAFFURI, 2012, S. 91). Die Durchführung der Kachelung (Tiling) der Vektordaten bewirkt die Erstellung von Kacheln bzw. Tiles. Diese entsprechen der rechteckigen Repräsentation der in ihr liegenden Geodaten (CAVAZZI, 2018, o. S.). Es ergibt sich der Vorteil, dass nur noch die im aktuellen Kartenausschnitt liegenden Tiles übergeben werden müssen, was zu einer verbesserten Performance der Webkarte führt (LI et al., 2018, S. 4).

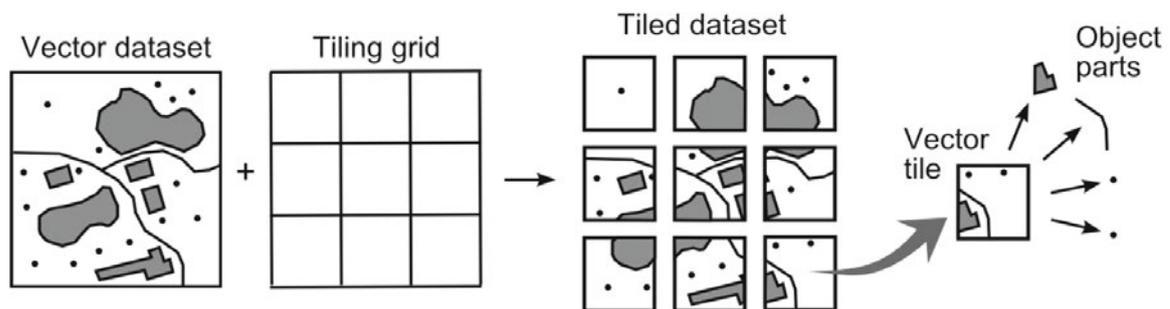


Abbildung 2.20: Prinzip der Erstellung von Vector-Tiles (GAFFURI, 2012, S. 91)

Die Abbildung 2.20 zeigt den Zusammenhang zwischen dem Vektordatensatz und der Kachelmatrix, die bei der Verschneidung die rechts dargestellten Tiles ergeben. Innerhalb einzelner Kacheln werden die Daten, die über den Rand hinausgehen, abgeschnitten, sodass Teile eines einzelnen Objekts in mehreren Tiles liegen können (GAFFURI, 2012, S. 91).

Die Webanwendung nutzt für diese Berechnung den Tileserver „t-rex“, welcher die Daten direkt aus einer PostGIS-Datenbank umformatiert (KALBERER, 2019, o. S.).

2.3.3 Kartendienst

Der in der Webanwendung verwendete Kartendienst zur Darstellung der Daten wird durch die JavaScript-Bibliothek „Mapbox GL JS“ bestimmt. Es stehen grundlegende Kartenfunktionen wie das Verschieben der Karte, das Zoomen, das Drehen (360°) oder das Neigen (0 - 60°) zur Verfügung (MAPBOX, 2019a, o. S.). Durch die von der Programmierschnittstelle (API - application programming interface) zur Verfügung gestellten Methoden zur Erweiterung der Kartenfunktionen, lässt sich die Webkarte beliebig verändern und durch zum Beispiel dynamisches Ergänzen von Layern individuell anpassen. Die Bibliothek erlaubt es interaktive Karten auf der Grundlage von Vector-Tiles und eigenen Stylings zu erzeugen (MAPBOX, 2019a, o. S.). Das Styling ist für die Visualisierung der Karte verantwortlich (MAPBOX, 2019b, o. S.). Es legt fest, welche Daten verwendet, wie sie dargestellt und in welcher Reihenfolge sie abgebildet werden (MAPBOX, 2019a, o.

S.). Es können über die API vordefinierte Stylings ausgetauscht, aber auch Eigenschaften bereits geladener Stylings angepasst werden (MAPBOX, 2019a, o. S.). Das Styling wird im JSON-Format (JavaScript Object Notation) angegeben (MAPBOX, 2019a, o. S.). Dieses definiert sich durch die Zuordnung von Key/Value- bzw. Schlüssel/Wert-Paaren (JSON.ORG, 2019, o. S.). Ein Beispiel für die Darstellung eines Layers „Wald“ zeigt sich in folgendem Ausschnitt:

```
1 {
2     "id": "Wald",
3     "type": "fill",
4     "source": "ATKIS",
5     "source-layer": "veg02_f",
6     "paint": {
7         "fill-color": "#e0e3de"
8     }
9 }
```

Code 2.1: Darstellung des Layers Wald (EIGENE).

Die Angabe der Keys „source“ und „source-layer“ steuert, welche Daten für den jeweiligen Layer genutzt werden. Über den Schlüssel „type“ wird die Art der Darstellung bestimmt, die in diesem Beispiel eine Flächenfüllung ist. Die farblichen Festsetzungen werden über den Key „paint“ definiert. Die „id“ erlaubt den späteren Zugriff auf den Layer.

2.3.4 Exkurs: GeoJSON

Im Kartendienst können neben den Vector-Tiles auch Daten im GeoJSON-Format verwendet werden (MAPBOX, 2019a, o. S.). Dieses ermöglicht die Kodierung von Geodaten in der JavaScript Object Notation (BUTLER et al., 2016, S. 3). Ein GeoJSON-Objekt kann u. a. eine räumliche Einheit, ein Feature oder mehrere räumliche Objekte (eine FeatureCollection) repräsentieren (BUTLER et al., 2016, S. 3). Außerdem ist es möglich, Attribute zu den einzelnen Objekten zu hinterlegen (BUTLER et al., 2016, S. 1), die beim Aufruf abgegriffen und verarbeitet werden können. Es werden die Geometrietypen Punkt, Linie, Polygon, MultiPunkt, MultiLinie, und MultiPolygon unterstützt (GEOJSON.ORG, 2019, o. S.). Die Objekte werden im Koordinatenreferenzsystem WGS84 angegeben (BUTLER et al., 2016, S. 1). Ein Minimalbeispiel für ein GeoJSON-Feature sieht wie folgt aus:

```
1 {
2   "type": "Feature",
3   "geometry": {
4     "type": "Point",
5     "coordinates": [125.6, 10.1]
6   },
7   "properties": {
8     "name": "Dinagat_Islands"
9   }
10 }
```

Code 2.2: GeoJSON-Beispiel (GEOJSON.ORG, 2019, o. S.).

2.3.5 Frontend: Angular

Die Umsetzung der Frontendkomponente, also des darzustellenden Teils der Webanwendung, wird durch das JavaScript-Framework „Angular“ realisiert. Angular ermöglicht die Erstellung von komponentenbasierten Single-Page-Applikationen (WOIWODE et al., 2018, S. vii) unter Verwendung von TypeScript (WOIWODE et al., 2018, S. ix).

TypeScript

Die Programmiersprache TypeScript wurde von Microsoft entwickelt und stellt eine Obermenge von JavaScript dar (WOIWODE et al., 2018, S. 27). Dies bedeutet, dass sämtlicher JavaScript-Code auch in TypeScript Validität besitzt (WOIWODE et al., 2018, S. 28). Der TypeScript-Code wird vor der Verarbeitung im Browser in reines JavaScript kompiliert, sodass die Lauffähigkeit sichergestellt ist (WOIWODE et al., 2018, S. 28). Eine der Kernideen bei der Entwicklung von TypeScript war die Einführung von Typsicherheit (HÖLLER, 2017, S. 727). Durch die statischen Datentypen kann den Variablen direkt bei der Initialisierung die Art zugeordnet werden. Mögliche Typen sind zum Beispiel „boolean“ (Wahrheitswerte), „number“ (Zahlenwerte), „string“ (Zeichenketten) oder „Array“ (typisierte Listen) (HÖLLER, 2017, S. 728 ff.). Das Array kennzeichnet sich dadurch, dass den Elementen ebenfalls Typen zugeordnet werden:

```
1 const strings: Array<string> = ['Yes', 'No'];
```

Code 2.3: Variablenzuordnung in TypeScript (HÖLLER, 2017, S. 731).

Die Typisierung der Variablen findet durch den Doppelpunkt statt. Die Angabe innerhalb der Winkelklammern spezifiziert den Typ der Elemente des Arrays.

Eine besonderer Basistyp ist „any“. Wird dieser als Angabe für eine Variable verwendet, wird die Typenüberprüfung eingestellt und der Wert der Variablen kann sich innerhalb des Programmcodes ändern (HÖLLER, 2017, S. 734). Das Verhalten entspricht in diesem Fall dem einer Variablen in JavaScript (WOIWODE et al., 2018, S. 34).

Node.js und npm

Die Erweiterung der Anwendung um diverse Module wird durch die Verwendung von Node.js und npm (Node Package Manager) ermöglicht (WOIWODE et al., 2018, S. 6 f.). Node.js ist die Plattform, die diese Pakete zur Verfügung stellt (WOIWODE et al., 2018, S. 6). Der Paketmanager npm wird für die Verwaltung und Installation der Software verwendet, die in JavaScript implementiert werden soll (HÖLLER, 2017, S. 25 f.). In der Datei „app.module.ts“ ist das modulare System „NgModule“ verankert, in welchem festgelegt wird, welche Bibliotheken der Anwendung zur Verfügung gestellt werden (GOOGLE, 2019d, o. S.).

Komponenten

Die Webanwendungen, die mit dem Framework Angular erstellt wurden, besitzen eine Modularität, die es erlaubt, einzelne Komponenten auszutauschen und in anderen Projekten wiederzuverwenden. Die Komponenten übernehmen jeweils Teilaufgaben und ergeben im Zusammenspiel die gesamte Anwendung (WOIWODE et al., 2018, S. 64). Sie bestehen standardmäßig aus vier Teilen (HÖLLER, 2017, S. 69). Die **Komponentenklasse** stellt die Logik der Anwendung durch eine TypeScript-Klasse dar und ermöglicht Interaktionen mit dem Nutzer (WOIWODE et al., 2018, S. 64). Das **Template** ist der darzustellende Teil der Anwendung und wird als HTML (Hypertext Markup Language)-Code implementiert (WOIWODE et al., 2018, S. 66). Die **Stylesheet-Datei** (CSS - Cascading Style Sheet) beeinflusst das Aussehen einer Komponente (WOIWODE et al., 2018, S. 68). Der **rudimentäre Testfall** wird hier nicht weiter erläutert. Er wird im Rahmen der Durchführung dieser Arbeit nicht benötigt.

Services

Services stellen in einer Angular-Anwendung, neben den Komponenten, einen weiteren Kernbestandteil dar (HÖLLER, 2017, S. 252). Die Hauptaufgabe liegt, neben der Kapselung der Applikationslogik, darin, Daten zu speichern, die in mehreren Komponenten benötigt werden (HÖLLER, 2017, S. 252). Es wird die Möglichkeit geboten, global verfügbare Variablen und Funktionen zu implementieren, die über die Laufzeit einer Komponente hinaus existieren.

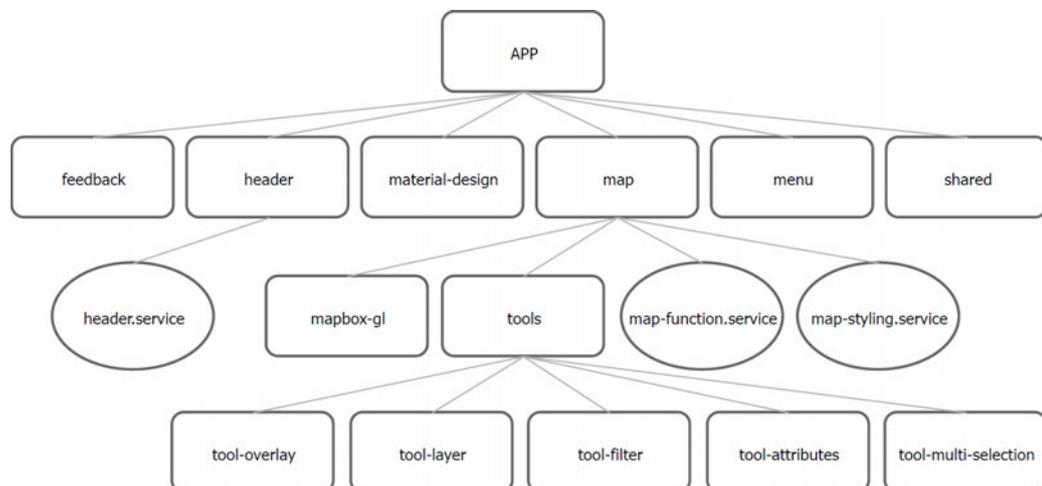


Abbildung 2.21: Komponenten der Webanwendung (EIGENE)

Die Abbildung 2.21 zeigt einen Ausschnitt aus der Struktur der Webanwendung. Übergeordnet befindet sich die Wurzel-Komponente „APP“ mit den ihr untergeordneten Komponenten (Rechtecke) und Services (Kreise). Ein Beispiel für einen Service stellt der „HeaderService“ dar. Im Header selbst wird der Titel der aktuell geladenen Komponente angezeigt, sodass dem Nutzer die Information über die Position, an der er sich innerhalb der Anwendung befindet, stetig angezeigt wird.

In Abbildung 2.22 wird die Änderung des Header-Titels beim Öffnen der Komponente „ToolAttributesComponent“, die Attributtabelle der selektierten Kauffälle zeigt, dargestellt.

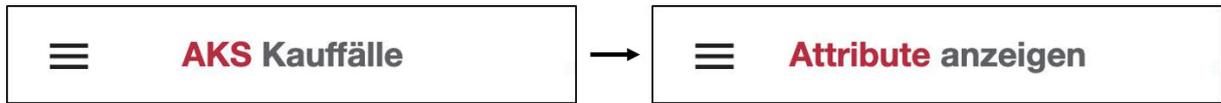


Abbildung 2.22: Veränderung des Titels der Webseite (EIGENE)

In Code 2.4 ist der Aufbau der Komponenteklasse des HeaderServices dargestellt. Der Variable „titleChanged“ wird ein EventEmitter zugewiesen, der über die Methode „emit()“ Werte übergeben kann (GOOGLE, 2019b, o. S.).

```

1 /* header.service.ts */
2 export class HeaderComponent {
3     title: string;
4     titleChanged = new EventEmitter<string>();
5     constructor() {}
6     changeTitle(title: string) {
7         this.title = title;
8         this.titleChanged.emit(this.title);
9     }
10 }

```

Code 2.4: Komponenteklasse des HeaderServices (EIGENE)

Die ebenfalls dem EventEmitter zugeordnete Methode „subscribe()“ fängt anschließend die vom „emit()“ übergebenen Werte ab und verarbeitet sie (GOOGLE, 2019b, o. S.). Dies geschieht in diesem Beispiel in der Komponenteklasse des Headers. Das Template stellt den dort abgefangenen Wert dar. Ausgelöst wird ein solches Event zum Beispiel beim Öffnen von Komponenten. Beim Initialisieren wird die „changeTitle()“-Methode des HeaderServices aufgerufen, sodass in der Folge der darzustellende Titel verändert wird.

Template Syntax

Durch die Template Syntax ermöglicht Angular eine Schnittstelle zwischen der TypeScript-Logik und dem darzustellenden Template (HÖLLER, 2017, S. 92). Die für diese Anwendung wichtigsten Bestandteile werden in der Tabelle 2.5 dargestellt.

Zeichen	Bezeichnung	Funktion
{{}}	Interpolation	Daten im Template anzeigen
	Property Binding	Eigenschaften eines DOM-Elements setzen
()	Event Binding	Ereignisse abfangen und behandeln
*	Strukturdirektiven	Eigenschaften lesen und Ergebnisse abfangen

Tabelle 2.5: Template-Syntax: Übersicht (WOIWODE et al., 2018, S. 79)

Durch die **Interpolation** werden Ausdrücke in Zeichenketten umgewandelt (GOOGLE, 2019f, o. S.). Der Template-Ausdruck kann aus Variablen der Komponentenkasse (Code 2.5), sowie aus Arithmetik bestehen (WOIWODE et al., 2018, S. 71). Änderungen der Daten bewirken eine automatische Aktualisierung des dargestellten Inhalts (WOIWODE et al., 2018, S. 71).

```
1 <p>{{title}}</p>
```

Code 2.5: Interpolation (GOOGLE, 2019f, o. S.)

Beim **Property Binding** wird der Wert eines Ausdrucks in die Eigenschaft eines DOM-Elements (Document Object Model) geschrieben (HÖLLER, 2017, S. 93). Folglich entsteht der Zugriff auf alle Eigenschaften des zugeordneten Ausdrucks (HÖLLER, 2017, S. 93). Ein Beispiel für das Property Binding ist im Code 2.6 dargestellt.

```
1 <img [src]="imageUrl">
```

Code 2.6: Property Binding (GOOGLE, 2019f, o. S.)

Das **Event Binding** erlaubt es, auf Ereignisse, wie zum Beispiel das Klicken eines Buttons, zu reagieren (WOIWODE et al., 2018, S. 72). Die Richtung des Datenflusses ist somit entgegengesetzt zu der des Property Bindings: die Information wird vom Template an die Komponentenkasse übergeben (WOIWODE et al., 2018, S. 72). Es können Informationen des Events über die Variable „\$event“ (GOOGLE, 2019f, o. S.), sowie durch eigene Parameter übergeben werden. Ein Beispiel eines Event Bindings ist in Code 2.7 dargestellt.

```
1 <button (click)="onSave($event)">Save</button>
```

Code 2.7: Event Binding (GOOGLE, 2019f, o. S.)

Die **Strukturdirektiven NgIf** und **NgFor** ermöglichen ein dynamisches Erzeugen von DOM-Elementen im Template (HÖLLER, 2017, S. 107). Die Verwendung von NgIf lässt eine Abfrage zu, sodass die Elemente nur in Abhängigkeit des Wahrheitswerts dem DOM-Baum angehängt werden (WOIWODE et al., 2018, S. 75). Über die Direktive NgFor wird über ein Array aus der Klassenkomponente iteriert und für jedes Element ein DOM-Element erzeugt (WOIWODE et al., 2018, S. 75). Ein Beispiel für die Verwendung von NgIf und NgFor ist in folgenden Codebeispielen (2.8 und 2.9) aufgeführt.

```
1 <div *ngIf="currentCustomer">Hello, {{currentCustomer.name}}</div>
```

Code 2.8: Strukturdirektive NgIf (GOOGLE, 2019f, o. S.)

```
1 <div *ngFor="let_item_of_items">{{item.name}}</div>
```

Code 2.9: Strukturdirektive NgFor (GOOGLE, 2019f, o. S.)

2.3.6 Backend-Services

Backend-Services werden unter anderem für das Verwalten serverbasierter Datenbanken verwendet (WOIWODE et al., 2018, S 169). Die am häufigsten verwendeten Methoden werden durch die „CRUD“-Operationen ausgedrückt: Create, Read, Update und Delete (WOIWODE et al., 2018, S. 169). In der Webanwendung wird dieses Verfahren über die Angular-Klasse „HttpClient“ realisiert. Ein Beispiel für einen Backend-Service ist der „rating-service“. Dieser speichert Einträge der Feedback-Funktion. Es handelt sich um einen Python-Dienst, der parallel zur Anwendung läuft, aber im Unterschied zu dieser über eine zentrale Datenbank verfügt, dessen Laufzeit die der Anwendung übersteigt. So findet eine persistente Datenhaltung statt. Die Klasse „HttpHeaders“ ermöglicht die Eingabe von Zugriffsdaten und die Deklarierungen der Datentypen der Rückgabewerte (GOOGLE, 2019c, o. S.). Das Nachfolgende Code-Beispiel (2.10) zeigt ein Beispiel für eine mögliche Implementierung einer Variablen dieser Klasse.

```

1 const httpOptions = {
2   headers: new HttpHeaders({
3     'Content-Type': 'application/json',
4     'Authorization': 'my-auth-token'
5   })
6 };

```

Code 2.10: HttpHeaders (GOOGLE, 2019c, o. S.)

Durch die Methode „HttpClient.post()“ können Daten unter Angabe der URL und der vorher definierten Optionen an den Service übergeben werden (GOOGLE, 2019c, o. S.), der die weitere Verarbeitung durchführt. Für die Ausführung der Anfrage an den Server, muss eine Subscription an der Variablen aufgerufen werden (GOOGLE, 2019c, o. S.), die eventuelle Rückgabewerte der serverseitigen Berechnung abfängt und diese in einer zugewiesenen Variablen speichern kann.

```

1 addHero (hero: Hero): Observable<Hero> {
2   return this.http.post<Hero>(this.heroesUrl, hero, httpOptions)
3     .pipe(
4       catchError(this.handleError('addHero', hero))
5     );
6 }
7 this.heroesService
8   .addHero(newHero)
9   .subscribe(hero => this.heroes.push(hero));

```

Code 2.11: „post()“-Methode des HttpClient (GOOGLE, 2019c, o. S.)

In Code 2.11 ist die Methode „addHero()“ zum Anlegen eines neuen „hero“ dargestellt (GOOGLE, 2019c, o. S.). Die „post()“-Methode mit den entsprechenden Parametern wird aufgerufen und über die Subscription ausgelöst. Der Rückgabewert wird im Array „this.heros“ gespeichert.

3 Konzept

Die Aufgabe dieser Arbeit besteht darin, die Ergebnisse einer Immobilienwertermittlung mittels Regressionsanalyse nach dem Vergleichswertverfahren in einer Webkarte zu visualisieren. Es sollen, neben der Darstellung der Kauffallobjekte und der Berechnung von statistischen Kenngrößen, gezielte Informationsabfragen ermöglicht werden, die bei der Suche nach geeigneten Vergleichsobjekten helfen.

Nachdem in Kapitel 2 eine Einführung in die Grundlagen der einzelnen Bestandteile gegeben wurde, folgt in diesem Kapitel das darauf aufbauende Konzept. Es soll ein Grundgerüst für die Umsetzung der Aufgabe geschaffen werden, das die Zusammenhänge der Komponenten beschreibt und bei der Implementierung durch strukturelle Verbindungen dient. Die Abbildung 3.1 zeigt das Ablaufschema der durchzuführenden Schritte.

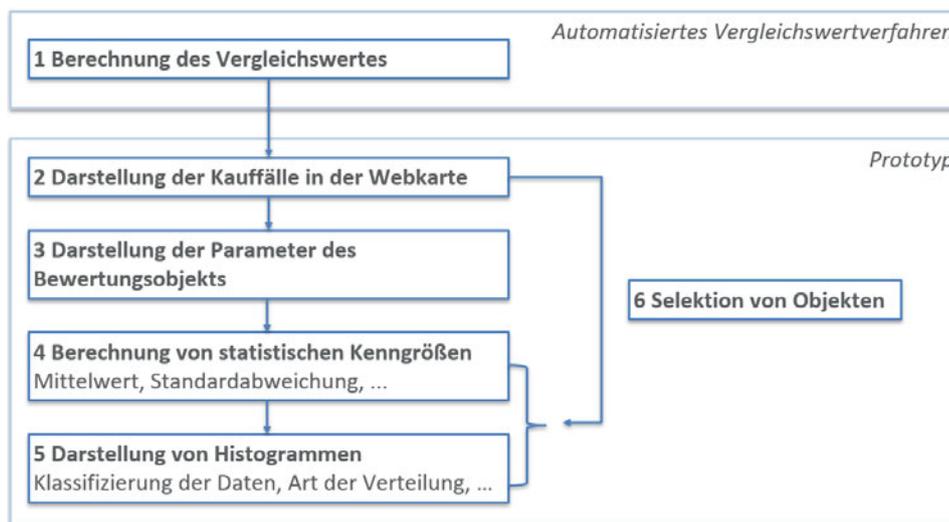


Abbildung 3.1: Ablaufdiagramm der durchzuführenden Schritte (EIGENE)

(1) Die Immobilienwertermittlung für das Bewertungsobjekt wird nach dem Vergleichswertverfahren mittels Regressionsanalyse durchgeführt. Verwendet werden soll das Tool „Automatisiertes Vergleichswertverfahren“.

(2) Die in der Webkarte zu visualisierenden Daten bestehen aus den Kauffällen, die in der Ausgleichung verwendet werden. Diese können entweder punkt- oder flächenförmig (z. B. entsprechend der Form des Flurstücks) dargestellt werden. Es kann durch unterschiedliche Färbungen auf unterschiedlich große Werte hingewiesen werden. Es bieten sich numerische Attribute (z. B. der Kaufpreis pro Quadratmeter Wohnfläche) oder Parameter der Ausgleichung (z. B. die Residuen) an. Als Webkarte soll der Prototyp der Projektgruppe „Basisvisualisierung II“ verwendet werden. Dieser bietet durch die gegebene Flexibilität (z. B. das Erweitern um diverse

Layer oder das Einbinden von zusätzlichen JavaScript-Bibliotheken) gute Voraussetzungen für die Darstellung der Kauffalldaten. Die Abbildung 3.2 zeigt links eine mögliche Zusammensetzung der Kartenelemente.

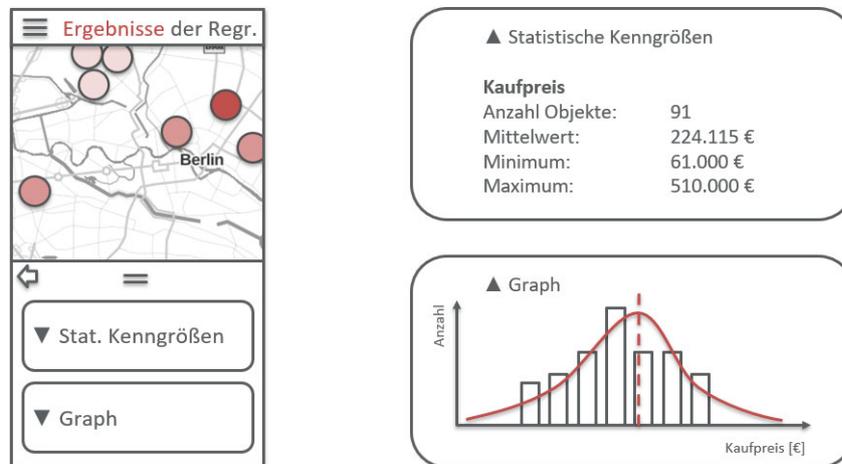


Abbildung 3.2: Konzeptueller Aufbau der Anwendungskomponenten (EIGENE)

(3) Die Eingabeparameter des Bewertungsobjekts (z. B. Wohnfläche, Baujahr, ...) sollen dargestellt werden, sodass Vergleiche zu anderen Objekten vorgenommen werden können. Außerdem sind die Ergebnisse der Regressionsanalyse (Kaufpreis, bzw. Kaufpreis pro Quadratmeter Wohnfläche) anzugeben.

(4) Zu den Kauffällen sollen statistische Daten berechnet und visualisiert werden. In diesem Fall bieten sich die Attribute und die Ergebnisse der Ausgleichung als mögliche Daten an. Es sind Größen wie zum Beispiel die Standardabweichung oder der Mittelwert der jeweiligen Datensätze zu ermitteln (Abbildung 3.2, rechts oben).

(5) Zusätzlich zu den statistischen Kennwerten sollen die Daten in Histogrammen veranschaulicht werden (Abbildung 3.2, rechts unten). Zur Umsetzung müssen die Kauffälle entsprechend der jeweiligen Attribute klassifiziert werden. Es soll im Graphen die Art der Verteilung der Daten wiedergegeben werden. In den Histogrammen ist der Wert des Bewertungsobjekts zu kennzeichnen, sodass eine Einordnung in die Gesamtheit durchgeführt werden kann.

(6) Über die Möglichkeit der Selektion von Objekten, können die Attribute dieses Kauffalls denen des Bewertungsobjekts gegenübergestellt werden. Die Einordnung des selektierten Kauffalls in die Gesamtheit kann über das Histogramm geschehen. In diesem sollen bei durchgeführter Selektion zwei Werte dargestellt werden: der des Bewertungsobjekts und der des selektierten Kauffalls.

4 Durchführung

Für die Implementierung der in Kapitel 3 beschriebenen Ansätze ist das Erstellen einer Angular-Komponente für die Verarbeitung und Darstellung der Inhalte in der Webkarte erforderlich. Außerdem wird ein Backend-Service für die Durchführung von Berechnungen benötigt.

Vorbereitende Maßnahmen: Frontend

Über den Befehl „ng generate component tool-statistics“ auf der Ebene „APP - map - tools“ wird die „ToolStatisticsComponent“ erzeugt. Sie fügt sich damit in die Komponentenstruktur (Abbildung 2.21) an der Stelle der übrigen Tools ein. Die Einbindung in die Webanwendung geschieht in der Komponentenklasse „MapComponent“. Es werden die am unteren Rand der Anwendung darzustellenden Buttons (MapTools) definiert. Diese setzen sich aus dem Namen eines Icons des Material-Design-Katalogs (Code 4.1, Zeile 2), sowie dem Pfad, an den bei durchgeführtem Klick innerhalb der Anwendung navigiert werden soll (Zeile 3), zusammen.

```
1 new MapTool(
2     'show_chart',
3     'statistics'
4 )
```

Code 4.1: Definition des MapTools für die Statistik-Komponente (EIGENE)

Dem Pfad wird über die sogenannten „Routes“ in der Moduldatei der Anwendung (app.module.ts) die zugehörige Komponente zugewiesen (Code 4.2). Diese wird beim Aufrufen des angegebenen Links geladen (GOOGLE, 2019e, o. S.).

```
1 const appRoutes: Routes = [
2     ...
3     {path: 'map', component: MapComponent, children: [
4         ...
5         {path: 'statistics', component: ToolStatisticsComponent}
6     ]},
7     ...
8 ];
```

Code 4.2: Zuweisung von Pfad und Komponente (EIGENE)

Der Klick auf einen Button in der Toolbar löst in der MapComponent eine (bereits implementierte) Funktion aus (Code 4.3). Diese navigiert über das Routing zu dem Pfad des jeweiligen Tools und setzt die Variable „showToolOverlay“ auf „true“.

```
1 onToolbarButtonClick(tool: MapTool) {
2     this.router.navigate(['map', tool.link]);
3     this.showToolOverlay = true;
4 }
```

Code 4.3: Funktion beim Klick auf Tool (EIGENE)

Das Template zur MapComponent beschreibt den darzustellenden Teil. Es wird der Komponenteninhalt des jeweiligen Tools überhalb der Karte dargestellt. Es ist die Strukturdirektive NgIf mit der Variable „showToolOverlay“ verwendet worden, sodass das Overlay nur dann geöffnet wird, wenn auf ein Icon in der Toolbar geklickt wurde.

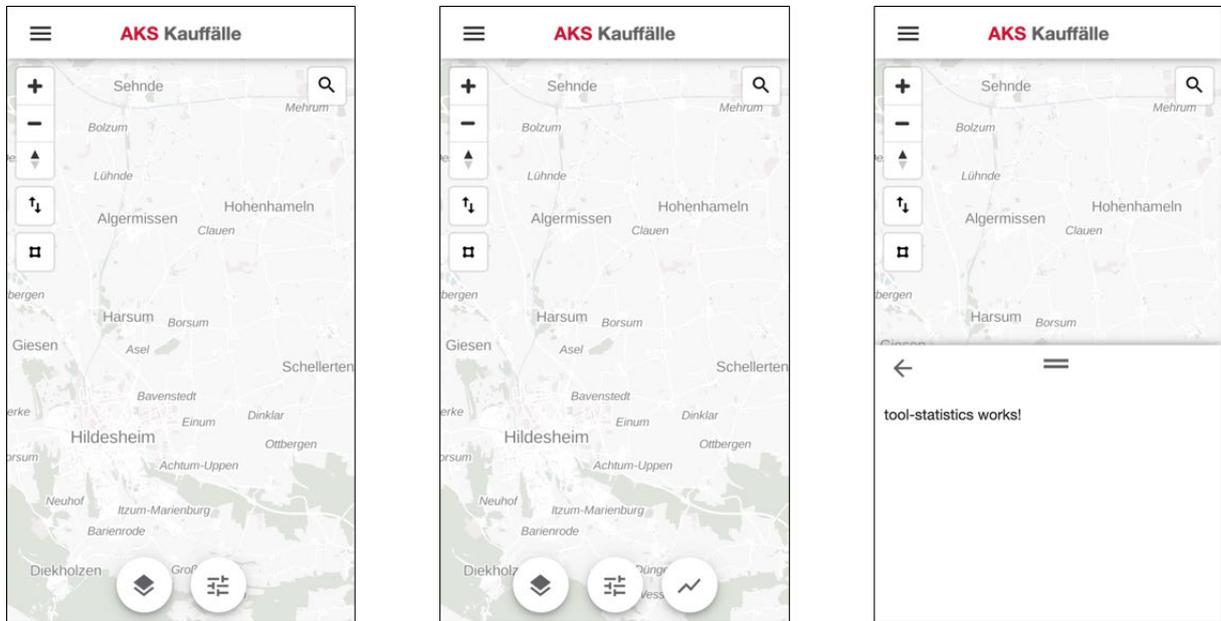


Abbildung 4.1: Einbinden der ToolStatisticsComponent (EIGENE)

Nach der Durchführung dieser Schritte ist eine leere Komponente in die Anwendung eingebunden, die über einen Klick auf das entsprechende Item in der Webanwendung aufgerufen werden kann. Es wird der automatisch generierte Text des Templates dargestellt. In der Abbildung 4.1 ist der Verlauf des Einbindens der Komponente in die Webanwendung zu sehen. Der linke Teil der Darstellung zeigt die Anwendung vor dem Erstellen der Komponente. In der mittleren Abbildung sind sämtliche beschriebene Schritte durchgeführt worden. Es ist das Icon in der Toolbar zu erkennen. Der rechte Teil der Grafik zeigt die geöffnete Komponente im Overlay. Dieser Teil ist in den folgenden Schritten mit Inhalten zu füllen. Außerdem ist die Kartenansicht um die Lage der in der Ausgleichung verwendeten Kauffälle zu ergänzen.

Vorbereitende Maßnahmen: Backend

Für die Erstellung des Backend-Services ist ein Python-Dienst mit dem Web-Framework „Flask“ verwendet worden. Dieser Service kann von der Frontend-Anwendung über eine URL („root_url“ in Verbindung mit „root_path“) angesprochen werden. Es ist die Angular-Klasse „HttpClient“ in der Komponentenklasse „ToolStatisticsComponent“ implementiert worden. Der Code 4.4 zeigt die Initialisierung des Python-Dienstes „calculation-service“. Die Methode „calculate“ kann über den entsprechenden Pfad in der URL aufgerufen werden.

```

1 service = Flask(__name__)
2 CORS(service)
3 root_url = 'http://localhost:5000'
4 root_path = '/calculation-service'
5 @service.route(root_path + "/calculate", methods=['POST'])
6
7 def calculate():
8     if not request.json:
9         abort(400)
10    else:
11        data = request.json
12        response_data = data
13        return jsonify(response_data)

```

Code 4.4: Initialisierung des Python-Dienstes (EIGENE)

In der Variable „request“ sind die von der Anwendung übermittelten Daten gespeichert, die weiter verarbeitet werden sollen. Sind diese nicht vorhanden, wird eine Fehlermeldung ausgegeben. In diesem Code-Ausschnitt werden die Eingangsdaten in derjenigen Variable gespeichert, die anschließend zurückgegeben wird. An dieser Stelle ist die Logik zur Berechnung der gesuchten Werte zu implementieren.

Durch das Erstellen der Komponente im Frontend und des Anlegens des Backend-Services sind die Grundlagen für die Berechnungen und die anschließenden Visualisierungen der Daten in der Webanwendung vorhanden. Die folgenden Kapitel beschreiben die Durchführung dieser Schritte nach dem in Kapitel 3 beschriebenen Konzept.

4.1 Berechnung des Vergleichswerts

Zur Berechnung des Vergleichswerts wird das Tool „Automatisiertes Vergleichswertverfahren“ verwendet. Es ist ein zufälliges Objekt in einem Wohngebiet in Hildesheim gewählt worden, das dem Teilmarkt Einfamilienhäuser zugeordnet wird. Die für die Berechnung erforderlichen Kenndaten des Grundstücks lauten:

Attribut	Wert
Lage	Hildesheim, XXXXXXXXXX
Wohnfläche	140 m ²
Standardstufe	2
Baujahr	1990
Grundstücksfläche	700 m ²
Wertermittlungsstichtag	12. November 2019
Modernisierungsgrad	6
Bodenrichtwert	160 € / m ²

Tabelle 4.1: Eingabeparameter des Bewertungsobjekts (EIGENE).

Die iterative Suche nach ähnlichen Kauffällen im Bereich des Bewertungsobjekts hat eine Selektion im Umfang von 331 Kauffällen ergeben. Bei der Durchführung der Regressionsanalyse

werden nur diejenigen Attribute verwendet, die in mehr als 60 % der Fälle belegt sind und dessen Parameter signifikant bestimmt werden können. Diese Bedingung und die Durchführung von Ausreißertests mit Hilfe der Cook-Distanzen ergibt eine Reduktion auf 142 Kauffälle. Mit diesen wurden in der Ausgleichung folgende Koeffizienten ermittelt:

Attribut	Koeffizient	Einheit
Kaufpreis / Wohnfläche	-75.316,23	€ / m^2
Baujahr	38,85	€ / m^2
Bodenrichtwert	5,01	1
Restnutzungsdauer	-39,57	€ / ($m^2 \cdot a$)
Modernisierungsgrad	-29,89	€ / m^2
Standardstufe	378,02	€ / m^2

Tabelle 4.2: In der Ausgleichung ermittelte Koeffizienten (EIGENE).

Der ermittelte Kaufpreis pro Quadratmeter Wohnfläche beträgt 1.618,25 € / m^2 , was einem Vergleichswert des Grundstücks von 226.554,48 € entspricht.

Die Export-Datei „geojson.json“ beinhaltet die in der Ausgleichung verwendeten Kauffälle mit den jeweiligen Attributen, sowie den aus der Ausgleichung hervorgehenden Cook-Distanzen und Residuen. Außerdem sind Punktkoordinaten für die einzelnen Kauffälle angegeben. Es handelt sich um eine GeoJSON-Datei, welche die einzelnen Objekte in einer FeatureCollection zusammenfasst. Für die Verwendung und Verarbeitung dieser Daten in der Webanwendung ist in der Komponentenkategorie „ToolStatisticsComponent“ eine Funktion implementiert worden, die ein GeoJSON-Objekt zurückgibt (Code 4.5). Die Daten dieser Berechnung sind somit statisch in der Anwendung integriert und werden in der Variable „geoJsonData“ gespeichert.

```

1 getGeoJsonData() {
2     return {
3         "type": "FeatureCollection",
4         ...,
5         "features": [{
6             "type": "Feature",
7             "id": 1,
8             "properties": {
9                 "kpwofl": 1006.0606,
10                ...,
11            },
12            "geometry": {
13                "type": "Point",
14                "coordinates": [ 9.9..., 52.2... ]
15            }
16        },
17        ...
18    ]
19 };
20 }

```

Code 4.5: Funktion zum Übergeben der Geojson-Daten (EIGENE)

Aus dem Inhalt der Exportdatei „Ergebnis.json“ ist ein JSON-Objekt zusammengesetzt worden, das über die Funktion „getInputData()“ abgerufen werden kann (Code 4.6). Es enthält Angaben zum Bewertungsobjekt, sowie die in der Ausgleichung verwendeten Parameter mit Angaben zu Einheiten und der Anzahl signifikanter Ziffern. Diese Informationen sind für die spätere Darstellung im Template, sowie für die Berechnung von Kenngrößen wichtig. Die Anzahl an signifikanten Ziffern bestimmt zum Beispiel mit wie vielen Nachkommastellen die Werte ausgegeben werden. Die Speicherung geschieht in der Variable „inputData“.

```
1 getInputData() {
2     return {
3         "coordinates": [9.9887955, 52.123371],
4         "sonstige": {
5             "labe": "████████████████████",
6             "gmna": "Hildesheim",
7             "prei": 226554.4828
8         },
9         "parameter":{
10            "kpwofl": {
11                "name": "Kaufpreis_/_Wohnflaeche",
12                "einheit": "Eur/m2",
13                "wert": 1618.2463,
14                "sign_ziff": 2
15            },
16            ...
17        }
18    };
19 }
```

Code 4.6: Funktion zum Übergeben von sonstigen Daten zur weiteren Verarbeitung (EIGENE)

4.2 Darstellung der Kauffälle in der Webkarte

Die Darstellung der Kauffälle kann punkt- oder flächenhaft umgesetzt werden. Sollen Flächen visualisiert werden, bieten sich die Flurstücke an, die das Grundstück bilden. Es ist präzise zu erkennen, welcher Teil der Erdoberfläche verkauft wurde und welche Bauwerke betroffen sind. Die Darstellung funktioniert jedoch nur in großen Maßstäben. Sind ganze Ortschaften oder Gemeinden abzubilden, können einzelne Flurstücke nicht mehr erkennbar sein. Die Art der Darstellung könnte maßstabsabhängig verändert werden, sodass ab einer festgelegten Maßstabszahl die Flächen in Punkte umformatiert werden, deren dargestellte Größe im Viewport konstant ist. Im hier betrachteten Fall, der möglichen Darstellung von über 300 Kaufobjekten, ist die Lage das Hauptmerkmal. Die Information über die Form ist nicht primär von Interesse, sodass die punktförmige Darstellung hinreichend ist.

Um das Bewertungsobjekt von den Kauffällen zu unterscheiden, wird ein Marker anstatt eines andersfarbigen Kreises verwendet. Der Nutzer kann direkt interpretieren, um was es sich bei den dargestellten Elementen handelt.

4.2.1 Klassifizierung nach Größe des Residuums

Durch verschiedene farbliche Darstellungen der Punktobjekte kann die Information über die Größe des jeweiligen Residuums direkt wiedergegeben werden. Es wird der Ansatz gewählt, dass Objekte mit kleinen Residuen farblich stärker dargestellt werden, als solche mit großen. Eine Klassifizierung der Daten in Abhängigkeit von der Größe des Residuums ist notwendig. Über die Angular-Klasse „HttpClient“ werden die GeoJSON-Daten und die Daten des Bewertungsobjekts an den Backend-Service übergeben.

```

1 var postData = {"geojson": this.geoJsonData, "attributes": this.inputData.parameter
  };
2 this.http.post('http://localhost:5002/calculation-service/calculate', postData,
  options).subscribe((response: any) => {
3     this.result = response;
4     this.mapStylingService.statisticsOn.emit([
5         this.result.geojson,
6         this.result.koordinaten,
7         this.inputData
8     ]);
9     ...
10 });

```

Code 4.7: Übermitteln der Daten an den Backend-Service (EIGENE)

Der Code 4.7 zeigt das Übermitteln der Daten an den „calculation-service“. Mit der Methode „calculate“ werden die Klassen für die Darstellung der Kauffalldaten berechnet und zurückgegeben. Gespeichert wird das JSON-Objekt in der Variable „result“. Der Backend-Dienst sucht im ersten Schritt das betragsmäßig größte Residuum. Dieser Wert wird anschließend mit dem Divisor 5 geteilt, woraus sich das Maß b für die Bestimmung der Klassengrößen ergibt. Eine Klasse entspricht der Größe von $2b$. Da das betragsmäßig größte Residuum als Bezug für die Klassenbreite herangezogen wird, ist entweder das Minimum oder das Maximum entsprechend der äußeren Klassengrenze der Klasse 3. In diesem Beispiel ist dies das Minimum.

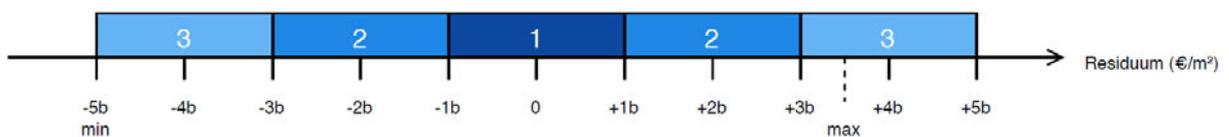


Abbildung 4.2: Einteilung der Residuen in Klassen (EIGENE)

Es folgt eine Iteration über sämtliche Kauffälle und es wird in jedem Schritt die Zuordnung in eine der drei Klassen vorgenommen. Die zugeordnete Klasse wird als weiteres Attribut an das jeweilige Feature des GeoJSON-Objekts angehängt. Das gesamte, angepasste GeoJSON-Objekt wird anschließend über das Rückgabeobjekt des Backend-Services an die Webanwendung übergeben. In Code 4.7 ist dies durch die über den EventEmitter „statisticsOn“ an den MapStylingService übermittelten Parameter „result.geojson“ gekennzeichnet. Die Abbildung 4.2 zeigt die für die Klassen zu verwendenden Farben.

4.2.2 Erstellen des Layers

Ein weiterer im Backend bestimmter Parameter ist „result.koordinaten“. Dieser beinhaltet die minimalen und maximalen Werte der geographischen Länge und Breite der Kaufobjekte: die Bounding Box.



Abbildung 4.3: Übermitteln der Daten an die Komponenteklasse „MapboxGlComponent“ (EIGENE)

Die Abbildung 4.3 zeigt den Weg der Daten von der `ToolStatisticsComponent`, über den `MapStylingService` zur `MapboxGlComponent`. In dieser Komponenteklasse ist das Einbinden des Layers implementiert. Durch die Subscription ausgelöst, werden im ersten Schritt diejenigen Kartenlayer ausgeblendet, die die generelle Darstellung der Kauffälle der AKS beinhalten, sodass sich diese Objekte nicht mit den hinzukommenden überschneiden. Es wird nur ein Thema in der Karte dargestellt, was für eine bessere Lesbarkeit sorgt. Über die durch den Kartendienst Mapbox zur Verfügung stehende Methode „`fitBounds()`“ wird der Kartenausschnitt im Viewport durch das Verwenden des Parameters „result.koordinaten“ auf das Gebiet geschwenkt und gezoomt (MAPBOX, 2019a, o. S.), in welchem die Kauffälle liegen. Durch das manuelle Anpassen der Eckpunktkoordinaten um Summanden bzw. Subtrahenden wird der Ausschnitt so verändert, dass dieser auch bei geöffnetem Overlay innerhalb des sichtbaren Kartenfensters liegt. Anschließend wird die Funktion „`createNewLayerStatistics()`“ aufgerufen. Diese erzeugt eine neue GeoJSON-Datenquelle und verwendet diese bei der Initialisierung des neuen Layers (Code 4.8).

```

1 this.map.addLayer({
2     id: 'statistics_lyr',
3     source: 'statistic_src',
4     type: 'circle',
5     paint: {
6         "circle-radius": {
7             "stops": [[12, 7], [22, 180]],
8             "base": 1.75
9         },
10    "circle-opacity": 0.8,
11    "circle-pitch-alignment": "map",
12    "circle-color": [
13        "case",
14        ["==", ["get", "resClass"], 1], "rgba(13, 71, 161, 1)",
15        ["==", ["get", "resClass"], 2], "rgba(30, 136, 229, 1)",
16        ["==", ["get", "resClass"], 3], "rgba(100, 181, 246, 1)",
17        "rgba(255, 0, 0, 1)",
18        ...
19    ]
20 });
  
```

Code 4.8: Styling des Layers der Kauffallobjekte (EIGENE)

Der mit dem Wert „circle“ festgelegte Schlüssel „type“ definiert, dass die Objekte als Kreise dargestellt werden sollen. Durch das Angeben von unterschiedlichen Farbeigenschaften „paint“ werden beim Rendern der Objekte die dort bestimmten Festsetzungen angewandt (MAPBOX, 2019b, o. S.).

circle-radius

Die Größe der Kreise wird über die Angabe der Eigenschaft „circle-radius“ gesteuert (MAPBOX, 2019b, o. S.). Durch die Verwendung von „stop“ und „base“ wird die Darstellung in Abhängigkeit von der Zoomstufe hergestellt. Die Zoomstufe ist proportional zum Maßstab. Der Parameter „base“ gibt die exponentielle Basis der Interpolationskurve zwischen den in „stop“ festgelegten Grenzen an. Der Radius wird zwischen 7 und 180 Pixel bei einer minimalen Zoomstufe von 12 und einer maximalen Zoomstufe von 22 angegeben. Außerhalb der Grenzen findet keine Interpolation statt. Es wird dort der Grenzwert verwendet.

circle-opacity

Die Eigenschaft „circle-opacity“ steuert die Deckkraft der Objekte (MAPBOX, 2019b, o. S.). Der Wert liegt zwischen 0 (Sichtbarkeit: 0 %) und 1 (Sichtbarkeit: 100 %) (MAPBOX, 2019b, o. S.). Der hier gewählte Wert von 0,8 sorgt für Transparenz, sodass an denjenigen Stellen, an denen mehrere Kauffälle in einem kleinen Gebiet liegen, eine Unterscheidbarkeit weiterhin gewährleistet ist. Der Verdeckung von Objekten durch Überlappung wird auf diese Weise entgegengewirkt. Außerdem können unter den Kreisen liegende Gebäude wahrgenommen werden.

circle-pitch-alignment

Das „circle-pitch-alignment“ sorgt mit dem Attribut „map“ dafür, dass die darzustellenden Kreise in der Kartenebene liegen und somit der veränderlichen Neigung der Karte folgen (MAPBOX, 2019b, o. S.).

circle-color

Die farbliche Festlegung der Kreisdarstellung wird über das Attribut „circle-color“ gesteuert (MAPBOX, 2019b, o. S.). Es wird ein „case“-Ausdruck verwendet, der die Nutzung von Bedingungen zulässt (MAPBOX, 2019b, o. S.). Die getätigte Implementierung überprüft das Attribut „resClass“ der einzelnen Kauffälle aus der GeoJSON-Datenquelle, in welchem die Klassifizierung nach den Residuen hinterlegt ist, auf Gleichheit mit den Werten 1, 2 und 3. Ist die Übereinstimmung gegeben, wird der zugeordnete Farbwert für die Darstellung verwendet. Die Farben und Klassen entsprechen dem Schema nach Abbildung 4.2. Ergibt keine der Bedingungen einen wahren Rückgabewert, wird der am Ende definierte „default“-Wert verwendet (MAPBOX, 2019b, o. S.).

4.2.3 Erstellen von PopUps

Für eine erste Informationsvermittlung sind PopUps beim Überfahren der Objekte mit der Maus zu erstellen. Zur Einordnung sollen neben dem Kauffallkennzeichen die Werte des Kaufpreises pro Quadratmeter Wohnfläche, sowie die Größe des Residuums ausgegeben werden.

```

1 this.map.on("mousemove", "statistics_lyr", (event: any) => {
2     ...
3     hoveredObjId = event.features[0].id;
4     popUp = new mapboxgl.Popup(...)
5         .setLngLat(event.lngLat)
6         .setHTML(
7             '<p_style="font-weight:_bold">' +
8             event.features[0].properties['kfkz'] + '</p>' +
9             '<p>_Kaufpreis_/_Wohnflaeche:_ ' +
10            event.features[0].properties['kpwofl'].toFixed(2) + ' _Eur/m2
11            </p>' +
12            '<p>_Residuum:_ ' +
13            event.features[0].properties['residuen'].toFixed(2) + ' _Eur/
14            m2</p>' )
15            .addTo(this.map);
16     ...
17 });

```

Code 4.9: PopUp-Erzeugung beim mouseover (EIGENE)

Der Code 4.9 zeigt einen Ausschnitt aus der Implementierung zur Erzeugung der PopUps. Das „mousemove“-Event wird auf den Layer „statistics_lyr“ angewendet. Es wird durchgehend überprüft, ob sich die Maus bei einer Bewegung über einem oder mehreren Objekten des Layers befindet. Wird das Event ausgelöst, wird das erste Objekt der Liste verwendet und für dieses ein PopUp mit den darzustellenden Informationen an der Stelle der Maus im Viewport erzeugt. Das Event wird jedoch auch dann ausgelöst, wenn sich die Maus bereits auf einem Objekt befindet und sich weiterbewegt, aber überhalb des Kreises bleibt. In diesen Fällen muss über eine Hilfsvariable die Information gespeichert werden, ob aktuell ein PopUp dargestellt wird oder nicht. Die Überprüfung wird über das Speichern der ID der jeweiligen Kauffälle realisiert.

```

1 this.map.on("mouseleave", "statistics_lyr", function() {
2     if (hoveredObjId) {
3         popUp.remove();
4     }
5     ...
6 });

```

Code 4.10: PopUp-Entfernung beim mouseleave (EIGENE)

Das „mouseleave“-Event wird dann ausgelöst, wenn sich die Maus von einem Objekt des Layers herunterbewegt. In diesem Fall wird das aktuell dargestellte PopUp entfernt (Code 4.10). Wird die Maus von einem Objekt direkt zum nächsten bewegt, ohne dass sie vorher ein „mouseleave“-Event ausgelöst hat, wird das PopUp entfernt und ein neues mit den aktuellen Werten erzeugt. Auch hier findet die Fallprüfung über die Verwendung der Objekt-ID statt.

4.2.4 Darstellung des Bewertungsobjekts

Für die bessere Unterscheidbarkeit von realen Kauffällen und dem zu bewertenden Objekt, wird für diesen ein Marker, statt einer kreisförmigen Darstellung, verwendet. Der Kartendienst Mapbox bietet die Möglichkeit, Marker über die Angabe eines Koordinatenpaares der Karte anzuhängen (MAPBOX, 2019a, o. S.). Außerdem kann die Verbindung zu einem PopUp geschaffen werden, welches den zugewiesenen HTML-Code darstellt (MAPBOX, 2019a, o. S.). Dieser ist in Code 4.11, Zeile 6 beschrieben. Als Koordinaten werden die über die Variable „statisticsOn“ übergebenen Werte des Bewertungsobjekts genutzt. Diese sind nach der Subscription auf dem EventEmitter in der Variable „data“ hinterlegt. Das PopUp erscheint bei einem Klick auf den Marker und wird durch einen weiteren Klick geschlossen.

```

1 var popUpVglObj = new mapboxgl.Popup({
2     closeButton: false,
3     anchor: "top-left",
4     offset: 10
5 })
6 .setHTML('<p_style="font-weight:_bold">_Bewertungsobjekt_</p>');
7
8 this.marker = new mapboxgl.Marker()
9     .setLngLat(data[2].coordinates)
10    .addTo(this.map)
11    .setPopup(popUpVglObj);

```

Code 4.11: Erstellen des Markers für das Bewertungsobjekt (EIGENE)

Die Abbildung 4.4 zeigt die in der Karte eingebundenen Kauffälle. Der linke Teil der Abbildung ist in einem kleineren Maßstab dargestellt als der rechte Teil. Zusätzlich zu den Kauffällen sind die PopUp-Elemente eingeblendet: links das des Bewertungsobjekts und rechts das eines Kauffalls. Durch die Darstellung von Ankern kann die Zugehörigkeit der PopUps zu den entsprechenden Objekten intuitiv interpretiert werden.



Abbildung 4.4: Kartendarstellung in verschiedenen Maßstäben (EIGENE)

4.3 Darstellung der Parameter des Bewertungsobjekts

Unter die darzustellenden Parameter des Bewertungsobjekts fallen neben den Eingabeparametern auch die Ergebnisse der Regressionsanalyse.

4.3.1 Verarbeitung in der Komponentenkasse

In der Komponentenkasse werden die Werte der darzustellenden Parameter in der Variable „inputData“ gespeichert. Über die Hilfsvariable „inputData_usedAttributes“ wird ein Objekt angelegt, das aus Schlüssel-Wert-Paaren besteht. Die Elemente sind dabei die in der Ausgleichung verwendeten Parameter. Der Schlüssel ist das Kürzel des Attributs, der Wert ist der Name. Dieses Objekt enthält somit nicht alle Eingabeparameter des Bewertungsobjekts, jedoch diejenigen, die tatsächlich verwendet wurden und somit vergleichbar mit eventuell selektierten Objekten sind. Die Angabe über die Anzahl der verwendeten Kauffälle ist an dieser Stelle wiederzugeben. Dieser Wert entspricht dem der in der Karte dargestellten Objekte. In der Variable „result“ ist die Angabe über die Anzahl der Kauffälle enthalten. Diese wurde im Backend-Service berechnet und übergeben. Durch die globale Initialisierung der beschriebenen Parameter in der Komponentenkasse „ToolStatisticsComponent“ (Code 4.12) kann im Template auf die Variablen zugegriffen werden.

```

1 export class ToolStatisticsComponent implements OnInit, OnDestroy {
2     inputData: any;
3     inputData_usedAttributes: any;
4     result: any;
5     ...
6 }

```

Code 4.12: Initialisierung der Parameter in der Komponentenkasse (EIGENE)

4.3.2 Verarbeitung im Template

Die in der Anwendung darzustellenden Inhalte werden in einem regelmäßigen Raster in sogenannten „Cards“ angeordnet, welche als Container für Text, Fotos und Aktionen dienen (GOOGLE, 2019a, o. S.). Einen schematischen Aufbau der Elemente zeigt die Abbildung 4.5.

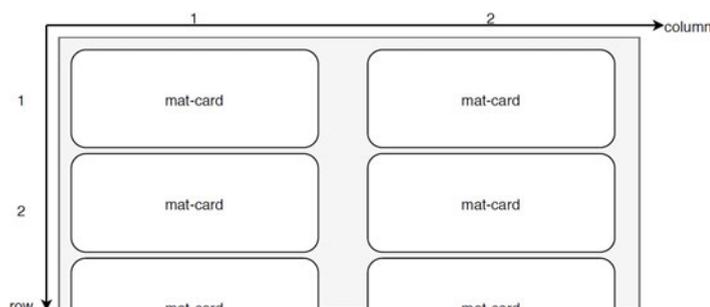


Abbildung 4.5: Elementanordnung im Viewport (EIGENE)

Die Cards werden von Angular durch das „Material Design“ zur Verfügung gestellt. Der Code 4.13 zeigt die Anordnung der Elemente im HTML-Template. Durch das Zuordnen einer Klasse an die Card können die Dimensionen und die Anordnung im Overlay durch ein Styling (SCSS) festgelegt werden. Zusätzlich wird an dieser Stelle eine weitere Angabe zur Darstellung getroffen: benötigt der Inhalt in einem Container in vertikaler Richtung mehr Platz als durch die festgelegte Größe vorhanden ist, ist das Scrollen innerhalb der Card möglich.

```

1 <div class="row">
2   <div class="col">
3     <mat-card class="mc501" style="overflow-y:_scroll;">
4       <mat-card-content>
5         ...
6       </mat-card-content>
7     </mat-card>
8   </div>
9   ...
10 </div>

```

Code 4.13: Grundgerüst der Anordnung der Card-Elemente (EIGENE)

Das erste Card-Element (oben links, Reihe und Spalte 1) wird mit den Angaben zum Bewertungsobjekt gefüllt. Unter der Überschrift „Ermittelter Vergleichswert“ werden die Angaben zur Lage des Objekts, sowie die Ergebnisse der Ausgleichung dargestellt (Code 4.14). Zur Ausgabe der Eingabeparameter wird die Strukturdirektive NgFor genutzt, unter dessen Verwendung über die Elemente des Objekts „inputData_usedAttributes“ iteriert wird. Ein Iterationsschritt ergibt dabei jeweils eine Zeile in der Darstellung. Diese werden in zwei Spalten unterteilt. Die linke Spalte beinhaltet den Namen des Attributs, die rechte den Wert des Parameters, sowie die Einheit. Der Wert wird dabei nur mit der vorher festgelegten Anzahl an signifikanten Ziffern wiedergegeben. Es wird die Interpolation zur Umformatierung der Variablen in darstellbaren Text verwendet.

```

1 <h2 class="example-h2">Ergebnisse</h2>
2 ...
3 <h4>Ermittelter Vergleichswert</h4>
4 <div class="row">
5   <div class="col">Gemeinde:</div>
6   <div class="col">{{ inputData.sonstige.gmna }}</div>
7 </div>
8 ...
9 <h4>Eingabeparameter des Bewertungsobjekts</h4>
10 <div *ngFor="let_item_of_inputData_usedAttributes_|_keyvalue">
11   <div class="row">
12     <div class="col">{{ item.value.name }}</div>
13     <div class="col">{{ item.value.wert.toFixed(item.value.sign_ziff) }}
14       {{item.value.einheit }}</div>
15   </div>
16 </div>

```

Code 4.14: Implementierung der Darstellung des Bewertungsobjekts (EIGENE)

Die Abbildung 4.6 zeigt das Resultat der Implementierung der Daten des Bewertungsobjekts im Overlay der Webanwendung. Links ist ein Screenshot der gesamten Anwendung zu sehen. Im Overlay sind zwei Cards angelegt. Eine für die hier beschriebene Darstellung der Informationen zum Bewertungsobjekt und eine, die die Überschrift „Selektion“ trägt. Sie ist an dieser Stelle Platzhalter für eine später durchführbare Selektion und die Darstellung der dort relevanten Informationen. Auf der rechten Seite der Grafik findet sich eine Zusammenstellung aller in der Card dargestellten Inhalte.

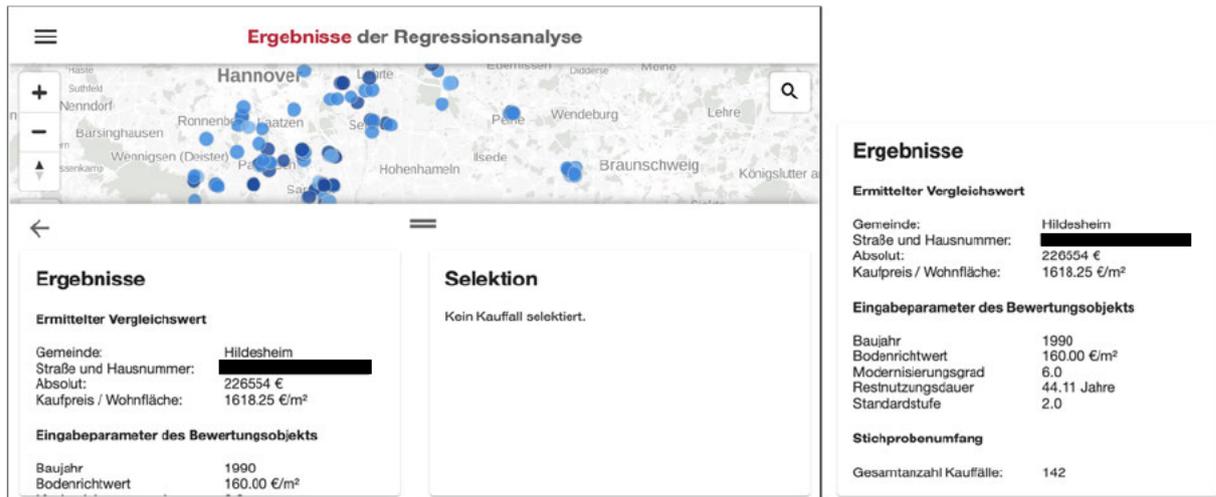


Abbildung 4.6: Ergebnisdarstellung im Overlay (EIGENE)

4.4 Berechnung von statistischen Kenngrößen

Die Berechnung der statistischen Kenngrößen wird für die in der Ausgleichung verwendeten Attribute, sowie für die Residuen und die Cook-Distanzen durchgeführt. Diese Informationen sind Teil der GeoJSON-Datei. Sie werden an den Backend-Service übergeben, in welchem die Berechnungen durchgeführt werden. Die Ergebnisse werden in der Komponentenkategorie „ToolStatisticsComponent“ verarbeitet und anschließend, über die Implementierung im Template, in der Anwendung dargestellt.

4.4.1 Verarbeitung im Backend

Da für sämtliche Parameter die gleichen Daten erfasst werden sollen, wird mittels for-Schleife über ein Objekt iteriert, welches die entsprechenden Parameter beinhaltet. Die Ergebnisse einer Iteration werden einem Dictionary angehängt, welches am Ende in ein JSON-Objekt umformatiert und zurückgegeben wird. Zu Beginn der Berechnungen wird in sämtlichen Kauffällen nach dem kleinsten und größten Wert der Datenreihe gesucht. Zu diesen wird die ID und das Kauffallkennzeichen gespeichert. Das Kauffallkennzeichen und der jeweilige Extremwert sind für die Darstellung in der Anwendung relevant, die ID wird bei weiteren Interaktionen benötigt. So

kann in der Komponentenkategorie auf das entsprechende Element des GeoJSON-Objekts zugegriffen werden. Anschließend werden die Kenngrößen Mittelwert, Standardabweichung und Median der ordinären Daten ermittelt.

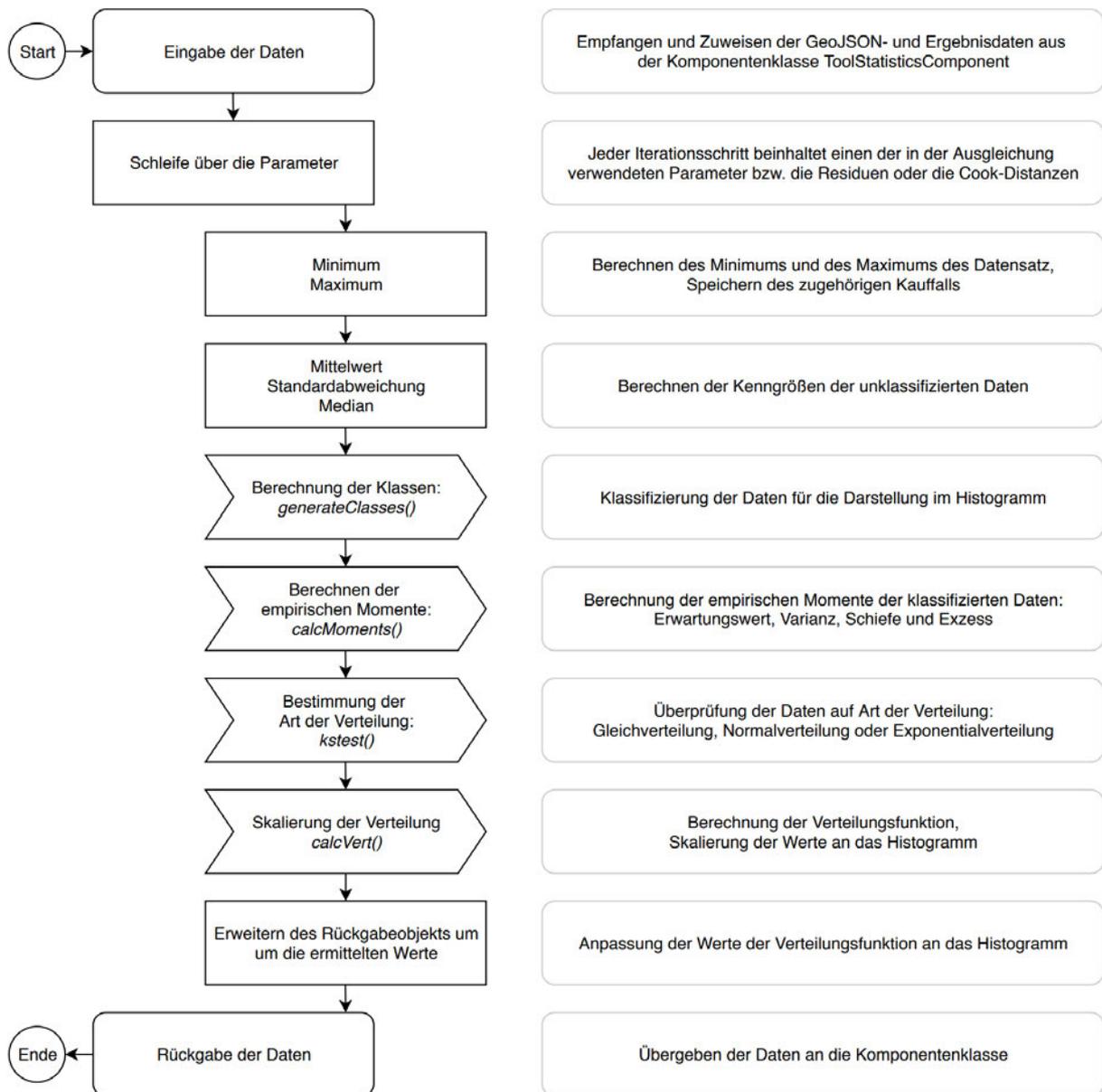


Abbildung 4.7: Ablaufdiagramm der Berechnungen im Backend (EIGENE)

Die Abbildung 4.7 zeigt das Ablaufschema der Berechnungen im „calculation-service“, sowie kurze, erläuternde Texte zu den jeweiligen Schritten. Es ist um die in den vorherigen Kapitel bereits erwähnten Prozesse reduziert.

generateClasses()

Bei der Einteilung der Daten in Klassen ist die jeweilige Breite „ h_n “ nach der Formel 2.27 bestimmt worden. Unter Verwendung der Differenz des maximalen und des minimalen Werts des

jeweiligen Attributs, sowie der bestimmten Klassenbreite, kann, neben der Anzahl der Klassen „hn_anz“, der Überschuss bestimmt werden.

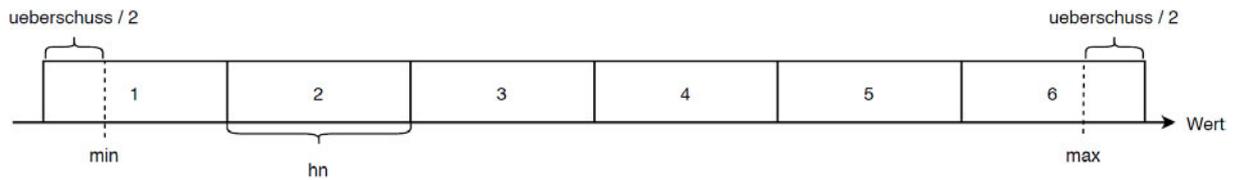


Abbildung 4.8: Beispiel für die Erstellung von Klassen (EIGENE)

Der Überschuss muss beim Anlegen der Klassen berücksichtigt und gleichmäßig auf die linke und die rechte Grenze verteilt werden. Realisiert wird dies, indem für den Anfangswert das um den halben Überschuss reduzierte Minimum verwendet wird und die folgenden Klassen in Schritten der Breite „hn“ erzeugt werden. Beispielfhaft dargestellt wird dieses Verfahren in Abbildung 4.8. Der Code 4.15 zeigt die Umsetzung im Backend-Service.

```

1 hn = 3.49 * sigma * anz ** (-1 / 3)
2 hn_anz = int((max - min) / hn) + 1
3 ueberschuss = hn * hn_anz - (max-min)
4
5 klassenHist = {}
6 p = min - ueberschuss / 2
7 while p < max:
8     klassenHist[p + hn/2] = 0
9     p += hn

```

Code 4.15: Anlegen der Klassen (EIGENE)

Das Dictionary „klassenHist“ enthält die Informationen über die Klassen. Die Schlüssel spiegeln die Klassenmitten wieder. Die zugeordneten Werte, die beim Anlegen des Dictionaryes mit dem Wert „0“ belegt werden, sind die ganzzahligen Häufigkeiten der vorkommenden Elemente des GeoJSON-Objekts. In einer Iteration über dieses findet die Zuordnung statt. Außerdem kann überprüft werden, welche Klasse die meisten Elemente enthält. Diese wird dem Modus zugeschrieben.

calcMoments()

Die Berechnung der empirischen Momente folgt den Formeln 2.31 bis 2.34 für diskrete Zufallsvariablen. Ermittelt werden die Werte des Erwartungswerts, der Varianz, der Schiefe, sowie des Exzesses. Die aus der Varianz abgeleitete Standardabweichung wird ebenfalls bestimmt.

kstest()

Es wird die am besten zu den Daten passende Verteilung gesucht. Die Funktion „kstest()“ führt den Kolmogoroff-Smirnoff-Anpassungstest für die klassifizierte Daten durch, dessen Parameter sich als Kriterium für die Zuordnung der Daten zu einer Verteilung eignen.

Durch die Verwendung des zuvor ermittelten Erwartungswerts und der Standardabweichung werden die Parameter λ (Exponentialverteilung), a und b (Gleichverteilung) bestimmt. In der Iteration über die Elemente des Dictionaries, in der die Klassengrenzen und Häufigkeiten abgetragen sind, wird der Wert der empirischen Häufigkeitsverteilung „empVertKum“ ermittelt und mit den Werten der verschiedenen Verteilungen in Differenz gesetzt. Die Durchführung bei der Bestimmung der Werte der Normalverteilung ist in Code 4.16 dargestellt. Die Variable „nv_xi“ stellt den Wert der Normalverteilung an der Stelle $x_i (= key)$ dar. Der Betrag der Differenz wird in der Variable „d_nv“ ausgedrückt.

```

1 empVertKum = 0
2 for key, value in hist.items():
3     empVertKum += value / n
4     nv_xi = scipy.stats.norm(my, sigma).cdf(key)
5     d_nv = abs(nv_xi - empVertKum)
6     test_nv.append(d_nv)

```

Code 4.16: Berechnung des Werts der empirischen Verteilungsfunktion und der Normalverteilung (EIGENE)

Dieses Schema wird für sämtliche Verteilungen durchgeführt. Die Listen „test_nv“, „test_ex“ und „test_gl“ enthalten die Werte der jeweiligen Testgrößen. Von diesen wird jeweils das Maximum bestimmt und überprüft, ob dieses kleiner als der Grenzwert D (Tabelle 2.2) ist. In diesem Fall werden die Art der zutreffenden Verteilung, sowie Informationen über die Testgröße und den Grenzwert zurückgegeben (Code 4.17).

```

1 if max_nv <= d_schranke:
2     max_nv_d = "Die_Daten_folgen_der_Verteilung"
3     max_nv_d2 = "(" + str(round(max_nv, 2)) + " <=" + str(round(d_schranke, 2))
4     + ")"
5     result = ["Normalverteilung", max_nv_d, max_nv_d2]

```

Code 4.17: Überprüfung auf Überschreitung des Grenzwerts (EIGENE)

Wird in keinem Fall der Grenzwert unterschritten, ist die Summe aller Werte D einer Verteilung zu bilden. Die Verteilung, dessen Summe den kleinsten Wert ergibt, wird dem Datensatz zugeordnet. Da die Summe verwendet wird, können einzelne Ausreißer nicht direkt zum Ausschluss der Verteilung führen. Zusätzlich wird die Information übermittelt, dass die Daten der Verteilung nicht folgen.

calcVert()

In der Funktion „calcVert()“ wird die skalierte Dichtefunktion der Verteilungen berechnet. Die Skalierung der Funktion ist für die Anpassung an die darzustellenden Histogramme notwendig. Es wird das Dictionary „klassenVert“ angelegt, dessen Elemente die Werte der Funktion beinhalten. Berechnet werden die Daten im Intervall $[min - 2 * hn; max + 2 * hn]$, sodass die Darstellung auch außerhalb des Histogramms erfolgen kann.

Der Code 4.18 zeigt die Implementierung am Beispiel der Normalverteilung.

```

1 u = min - 2 * hn
2 if artVerteilung == "Normalverteilung":
3     while u <= max + 2 * hn:
4         value = scipy.stats.norm(my, sigma).pdf(u)
5         klassenVert[u] = value * f
6         u += hn / 2

```

Code 4.18: Berechnung der Dichtefunktion (EIGENE)

Sind sämtliche Werte berechnet worden, wird das Rückgabeobjekt um diese erweitert. Nach der Wiederholung dieses Vorgangs für alle Attribute, wird das Objekt an die Komponentenkasse übergeben.

4.4.2 Verarbeitung in der Komponentenkasse

Die vom Backend-Service empfangenen Daten werden an die Funktion „generateChartData()“ übergeben. Dies geschieht iterativ für jedes Attribut. In der Funktion findet eine Zuweisung der berechneten Parameter an das global zur Verfügung stehende Objekte „chartDataset“ statt. Der Code 4.19 zeigt dieses Vorgehen. Dem Schlüssel „data“ werden die im „calculation-service“ ermittelten Daten des jeweiligen Attributs zugewiesen.

```

1 this.chartDataset[attr] = {
2     "title": name,
3     ...
4     "data": this.result.attributes[attr],
5     "einheit": einheit,
6     "sign_ziff": sign_ziff
7 }

```

Code 4.19: Zuweisung der im Backend berechneten Daten (EIGENE)

4.4.3 Verarbeitung im Template

Im Template werden iterativ Cards erstellt, dessen Inhalt die statistischen Kenngrößen der jeweiligen Attribute sind. Die Cards werden innerhalb der Anwendung (beginnend in Zeile 2) in Spalte 1 des Rasters dargestellt, in der Spalte 2 folgt die Visualisierung der Histogramme.

```

1 <div *ngFor="let_item_of_chartDataset_|_keyvalue">
2     <div class="row">
3         <div class="col">
4             <mat-card class="mc501" style="overflow-y:_scroll;">
5                 <mat-card-content>
6                     <h2 class="example-h2">{{ item.value.title
                          }}</h2>

```

Code 4.20: Template zur Darstellung der statistischen Kenngrößen (EIGENE)

Die in der Komponentenkategorie gespeicherten Daten werden durch die Interpolation in darstellbaren Text umformatiert (Code 4.20).

Bei der Darstellung der Schiefe und des Exzesses wird zusätzlich eine Fallabfrage durch die Verwendung der Strukturdirektive NgIf durchgeführt. Der Code 4.21 zeigt die Implementierung am Beispiel der Schiefe. Es wird die Überprüfung der Werte auf das Vorzeichen, bzw. auf die Gleichheit mit „0“ ausgeführt. Die in der Tabelle 2.3 zugeordnete Eigenschaft wird als weitere Angabe in der Anwendung dargestellt.

```

1 <div *ngIf="item.value.data.klassifiziert.schiefe_>_0" class="col">
2     {{ item.value.data.klassifiziert.schiefe.toFixed(2) }} (linkssteile
3     Verteilung)
4 </div>
5 <div *ngIf="0_>_item.value.data.klassifiziert.schiefe" class="col">
6     {{ item.value.data.klassifiziert.schiefe.toFixed(2) }} (rechtssteile
7     Verteilung)
8 </div>
9 <div *ngIf="item.value.data.klassifiziert.schiefe_==_0" class="col">
10    {{ item.value.data.klassifiziert.schiefe.toFixed(2) }} (symmetrische
11    Verteilung)
12 </div>

```

Code 4.21: Template zur Darstellung der Schiefe (EIGENE)

Nach durchgeführter Implementierung der Darstellung der statistischen Kenngrößen ergibt sich die in der Abbildung 4.9 (links) dargestellte Ansicht der Anwendung. Auf der rechten Seite ist der komplette Inhalt der Card dargestellt. Es werden das Minimum und das Maximum unter Angabe des zugehörigen Kauffalls gezeigt. Außerdem sind die berechneten Kenngrößen der unklassifizierten, sowie der klassifizierten Daten dargestellt. Zu diesen sind zusätzliche Informationen zum Histogramm und zur Verteilungsform abgebildet.

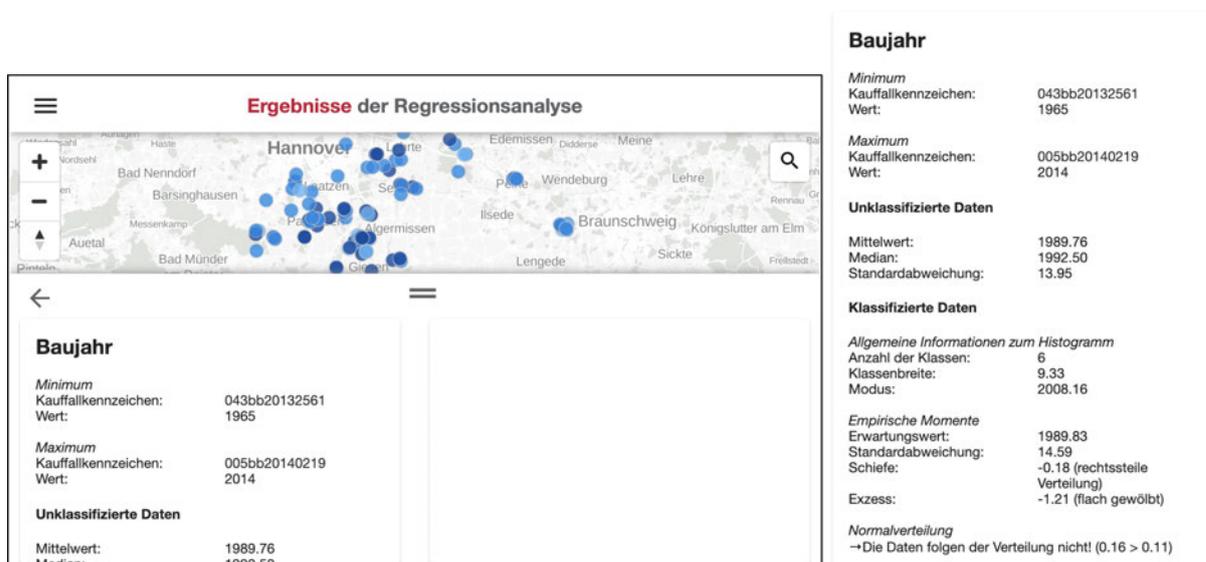


Abbildung 4.9: Darstellung der statistischen Kenngröße am Beispiel des Baujahrs (EIGENE)

4.5 Darstellung von Histogrammen

Die im calculation-service klassifizierten Datensätze sollen in Histogrammen dargestellt werden. Zusätzlich sind die zugeordneten Verteilungen zu visualisieren. Für die Einbindung von interaktiv bedienbaren Graphen wird die JavaScript-Bibliothek „ngx-echarts“ verwendet. Diese ermöglicht das Erzeugen von Grafiken, sowie das dynamische Erweitern um zusätzliche Datenbestände (THE APACHE SOFTWARE FOUNDATION, 2019, o. S.).

4.5.1 Verarbeitung in der Komponenteklasse

Für die Bestimmung, der für die Visualisierung erforderlichen Parameter, wird die Funktion „generateChartData()“ für jedes Attribut aufgerufen. Da das Datenformat bei der Zuweisung vom Datentyp Array sein muss, werden die Werte der klassifizierten Daten, sowie die der Verteilungen, umformatiert. Bei den Verteilungen wird zusätzlich eine Sortierung durchgeführt. Dies ist erforderlich, da die einzelnen Werte durch eine Linie verbunden werden sollen und die richtige Reihenfolge sichergestellt sein muss. Zusätzlich werden bei den Verteilungsfunktionen Exponentialverteilung und Gleichverteilung Datenwerte für den Übergang auf den Wert „0“ eingefügt, sodass die Kanten der Dichtefunktionen dargestellt werden.

```
1 chartOption = {
2     legend: {data: ["Histogramm", "Dichtefunktion", 'Bewertungsobjekt']},
3     tooltip: {},
4     xAxis: {name: einheit, type: 'value', ...},
5     yAxis: {name: "Anzahl", type: 'value', ...},
6     series: [
7         {name: "Histogramm", type: 'bar', data: dataHist},
8         {name: "Dichtefunktion", type: 'line', data: dataVert, ...},
9         {name: 'Bewertungsobjekt', type: 'line', data: [
10            [this.inputData.parameter[attr].wert, 0],
11            [this.inputData.parameter[attr].wert, this.result.attributes
12                [attr].verteilung.maxAnzahl * 1.25]}
13     ]
14 };
```

Code 4.22: Festlegen der Optionen zur Darstellung der Graphen (EIGENE)

Die im Graphen darzustellenden Daten und die Eigenschaften des Charts werden in einem Objekt („chartOption“, Code 4.22) festgelegt, das anschließend im Template eingebunden wird. Es enthält die Angaben zur Beschriftung der Legende („legend“), sowie dem darzustellenden Bereich und der Beschriftung der Achsen („xAxis“ und „yAxis“). Die Einführung der Eigenschaft „tooltip“ erzeugt PopUps mit Informationen zum aktuellen Wert der Daten im Graphen bei einem „mouseover“-Event über dem entsprechenden Datensatz. Durch die Zuweisung am Parameter „series“ werden die Daten eingebunden. Es wird zusätzlich der Typ der Darstellung festgelegt. Neben dem Histogramm und der Dichtefunktion wird der Wert des Bewertungsobjekts des jeweiligen Attributs im Graphen abgebildet. Dies wird durch eine vertikale Liniendarstellung realisiert, die

an der Stelle des Werts die 1,25-fache Höhe der maximal ausgeprägten Klasse besitzt. Der globalen Variable „chartDataset“ werden die festgelegten Chart-Optionen zugewiesen, auf die im Template zugegriffen werden kann. Die Bestimmung der Daten bei den Residuen und den Cook-Distanzen unterscheidet sich von den übrigen Attributen. In diesen Fällen sind keine Werte des Bewertungsobjekts vorhanden, sodass die Histogramme einen Datensatz weniger enthalten. Gespeichert werden diese Angaben in der global verfügbaren Variable „chartDatasetReg“.

4.5.2 Verarbeitung im Template

Die Histogramme der Attribute werden, wie die übrigen Elemente im Overlay der Anwendung, in Cards dargestellt. Diese befinden sich jeweils auf der gleichen Zeilenebene wie die Informationen zu den statistischen Kenngrößen. Es wird die zweite, rechte Spalte genutzt, sodass sich die darzustellenden Elemente eines Attributs nebeneinander anzeigen lassen.

```
1 <div echarts [options]="item.value.chartOption" (chartInit)="onChartInit($event, _
  item.key)" class="chart"></div>
```

Code 4.23: Einbinden der Histogramme im Template (EIGENE)

Die Einbindung der Graphen im Template wird über den Code 4.23 ausgeführt. Bei der Initialisierung wird die Funktion „onChartInit()“ in der Komponenteklasse „ToolStatisticsComponent“ ausgeführt. Es wird ein Event, sowie der Name des Attributs der aktuell darzustellenden Daten übergeben. Das Event beinhaltet die Instanz des Charts, die dem Objekt „chartDataset“ (bzw. „chartDatasetReg“) zugewiesen wird. Dies ermöglicht einen späteren Zugriff auf den Graphen und erlaubt Manipulationen. Benötigt wird diese Möglichkeit bei der Selektion von Objekten und der zusätzlichen Darstellung dieser in der Grafik.

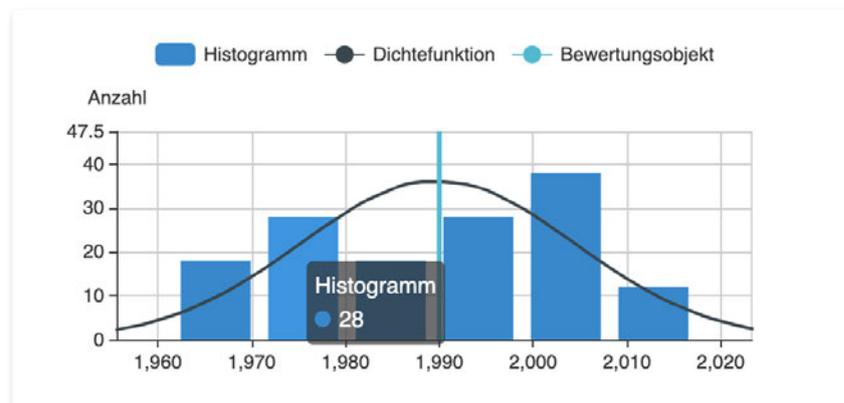


Abbildung 4.10: Darstellung der Graphen am Beispiel des Baujahrs (EIGENE)

Die Abbildung 4.10 zeigt das Ergebnis der Einbindung der Graphen am Beispiel des Baujahrs. In blau dargestellt ist das Histogramm, in Hellblau der Wert des Bewertungsobjekts und in Dunkelblau die Verteilung (Dichtefunktion) der klassifizierten Daten. Außerdem ist ein PopUp mit der Information zur Anzahl der Klassenelemente abgebildet.

4.6 Selektion von Objekten

Neben der Darstellung der Gesamtergebnisse der Regressionsanalyse sollen zusätzlich einzelne Kauffälle selektiert werden können, dessen Werte denen des Bewertungsobjekts entgegengestellt werden. Es werden interaktiv bedienbare Elemente eingeführt, die gezielt Vergleiche ermöglichen.

4.6.1 Durchführung der Selektion

Die Selektion kann innerhalb der Kartendarstellung oder innerhalb der Ausgabe der statistischen Kenngrößen durchgeführt werden. Um das selektierte Objekt in der Karte zu kennzeichnen, wird dies durch eine gesonderte farbliche Darstellung umgesetzt.

```
1 ["boolean", ["feature-state", "selected"], false, "rgba(251, 192, 45, 1)", ...]
```

Code 4.24: Fallabfrage für das Kennzeichnen des Selektionsobjekts (EIGENE)

Der Code 4.24 zeigt die Voraussetzung für das Färben einzelner Objekte des GeoJSON-Layers. Es wird eine zusätzliche Fallabfrage an das Styling („paint“ - „circle-color“) angehängt. Die Eigenschaft „selected“ wird an den Elementen als „feature-state“ eingeführt. Diese ist standardmäßig auf den Wahrheitswert „false“ gesetzt. Der feature-state kann im Anwendungsverlauf verändert bzw. abgefragt werden (MAPBOX, 2019b, o. S.). Für den Fall, dass er für ein Objekt auf den Wert „true“ gesetzt wird, ändert sich die Darstellung auf den angegebenen Farbwert. Einer der in der Karte visualisierten Kreise besitzt in diesem Fall eine Wiedergabe in der Farbe Gelb.

Selektion in der Karte

In der Komponentenklasse „MapboxGlComponent“, in der die Kartenfunktionen definiert werden, werden bei einem Klick in die Karte Objekte des Layers „statistics_lyr“ abgefangen und in dem Array „statFeatures“ gespeichert. Die Registrierung von Treffern löst die weitere Verarbeitung aus.

```
1 this.mapStylingService.statisticsClicked.emit(statFeatures[0]);
2 if (lastSelected) {
3     this.map.setFeatureState(
4         {source: 'statistic_src', id: lastSelected},
5         {selected: false}
6     );
7 }
8 this.map.setFeatureState(
9     {source: 'statistic_src', id: statFeatures[0].id},
10    {selected: true}
11 );
12 lastSelected = statFeatures[0].id;
```

Code 4.25: Färben der Objekte nach Klickselektion (EIGENE)

Der Code 4.25 zeigt die Verarbeitung der gefundenen Kauffälle. Über den MapStylingService wird das erste Objekt des Arrays an die Komponentenklasse „ToolStatisticsComponent“ übergeben.

Die dortige Ausführung der Funktion „addDataToCart“ steuert die Ausgabe der Eigenschaften des Selektionsobjekts. Sie wird im Kapitel 4.6.2 näher beschrieben. Weiter wird überprüft, ob die Variable „lastSelected“ einen Wahrheitswert besitzt. Dies ist der Fall, wenn bereits ein Objekt selektiert wurde. In diesem Fall wird über die Festlegung des feature-states die Eigenschaft „selected“ dieses Objekts auf „false“ gesetzt. Im Anschluss folgt die Zuweisung von „true“ an die Eigenschaft des aktuell selektierten Objekts. Dies bewirkt die entsprechende Färbung des Kartenelements. Abschließend wird die ID des Objekts in der Variable „lastSelected“ gespeichert, sodass die Informationen des zuletzt selektierten Objekts bei einer folgenden Selektion vorliegen und weiterverarbeitet werden können.

Selektion in der Ausgabe der statistischen Kenngrößen

Die Card zur Ausgabe der statistischen Kenngrößen gibt für die minimalen und maximalen Werte des jeweiligen Attributs das zugehörige Kauffallkennzeichen des Kaufobjekts an. Der Code 4.26 zeigt den darzustellenden Teil des Kauffallkennzeichens im Template des Statistik-Tools. Über die Interpolation wird das Attribut ausgegeben. Ein Klick auf dieses Element führt die Funktion „setSelected()“ in der Komponentenklasse aus. Dargestellt wird das Kauffallkennzeichen in einer anderen Farbe und mit einem „pointer-cursor“, der bei einem „mouseover“-Event erscheint, sodass der Nutzer die Klickbarkeit des Objekts erkennt.

```

1 <div class="col">Kauffallkennzeichen:</div>
2 <div class="col" (click)="setSelected(item.value.data.unklassifiziert.minimum.id) "
   style="cursor:_pointer;_color:_#01579b">
3     {{ item.value.data.unklassifiziert.minimum.kfkz }}
4 </div>

```

Code 4.26: Einbinden einer Funktion in die Ausgabe des Kauffallkennzeichens (EIGENE)

Der Funktion „setSelected()“ wird die ID des selektierten Kauffalls übergeben. Diese wird anschließend über den MapStylingService an die Komponentenklasse „MapboxGlComponent“ übermittelt. In dieser wird überprüft, ob bereits ein Kauffall selektiert ist und dessen Darstellung angepasst werden muss. Es wird die Kennzeichnung des selektierten Kauffalls durchgeführt. Anschließend übergibt die „setSelected()“-Methode das Kauffallobjekt an die Funktion „addDataToChart()“.

4.6.2 Darstellung des Selektionsobjekts

Die Selektion bewirkt Veränderungen an zwei Stellen innerhalb der Anwendung. Es wird eine Card erzeugt, in welchem die einzelnen Attributwerte des Selektionsobjekts aufgelistet und denen des Bewertungsobjekts gegenübergestellt werden. Außerdem werden die zu den Attributen gehörigen Graphen um die jeweilige Information erweitert.

Ausgabe in der Card „Selektion“

Dargestellt werden die Attributwerte des selektierten Objekts in der Card „Selektion“. Diese stellt standardmäßig den Hinweis dar, dass kein Kauffall selektiert ist. Wird diese jedoch durchgeführt, ändert sich die Überschrift in das Kauffallkennzeichen des ausgewählten Objekts. Umgesetzt ist dies, wie in Code 4.27 dargestellt, über die Strukturdirektive „NgIf“ und die Verwendung der Variable „selectedObj“. Diese speichert das selektierte Objekt in der Komponentenkategorie „Tool-StatisticComponent“ und dadurch dessen Eigenschaften, auf die mittels Interpolation zugegriffen werden kann. Ist keine Selektion aktiv, entspricht der Wert der Variable „null“.

```

1 <h2 *ngIf="!selectedObj" class="example-h2">Selektion</h2>
2 <h2 *ngIf="selectedObj" class="example-h2">{{ selectedObj.properties["kfz"] }}</h2>
3 <div *ngIf="!selectedObj">
4     <p>Kein Kauffall selektiert.</p>
5 </div>
6 <div *ngIf="selectedObj">
7     <h4>Allgemeine Angaben</h4>
8     ...
9 </div>

```

Code 4.27: Darstellung des selektierten Kauffalls in der Card (EIGENE)

Es findet eine Auflistung der Daten des Selektionsobjekts statt. Diese ist so sortiert, dass sie der Ausgabe der Attribute des Bewertungsobjekts entspricht. Werte gleicher Attribute können direkt miteinander verglichen werden. Zusätzlich wird die Differenz dieser beiden Werte berechnet und dargestellt. Die Angabe findet in der dritten Spalte innerhalb der Card statt.

```

1 <div class="row">
2     <div class="col-sm-6">
3         Kaufpreis:
4     </div>
5     <div class="col-sm-3">
6         {{ selectedObj.properties["prei"].toFixed(0) }} Eur
7     </div>
8     <div class="col-sm-3" *ngIf="selectedObj.properties['prei']_<_inputData.
9         sonstige.prei_>=_0">
10        +{{ (selectedObj.properties["prei"] - inputData.sonstige.prei).
11           toFixed(0) }} Eur
12    </div>
13    <div class="col-sm-3" *ngIf="0_>_selectedObj.properties['prei']_<_inputData.
14        sonstige.prei">
15        {{ (selectedObj.properties["prei"] - inputData.sonstige.prei).
16           toFixed(0) }} Eur
17    </div>
18 </div>

```

Code 4.28: Darstellung der Attributwerte und der Differenzen (EIGENE)

Der Code 4.28 zeigt die Zeile für die Ausgabe der Werte beim Attribut „Kaufpreis“. Es wird bei der Bestimmung der Differenzen mittels NgIf überprüft, ob dieser Wert positiv oder negativ ist. Die Ermittlung wird durchgeführt, da so auch ein positives Vorzeichen dargestellt werden kann.

005bb20140219		Zurücksetzen
Allgemeine Angaben		
Gemeinde:		
Straße:		
Kaufpreis:	250000 €	+23446 €
Kaufpreis / Wohnfläche:	2100.84	+482.59 €/m ²
Angaben zum Objekt		
Baujahr	2014	+24
Bodenrichtwert	115.00 €/m ²	-45.00 €/m ²
Modernisierungsgrad	0.0	-6.0
Restnutzungsdauer	70.00 Jahre	+25.89 Jahre
Standardstufe	4.0	+2.0
Statistische Kenngrößen		
Residuum:	-135.31 €/m ²	
Cook-Distanz:	0.00	

Abbildung 4.11: Inhalt der Selektionsausgabe im Overlay (EIGENE)

In der Abbildung 4.11 wird der gesamte Inhalt der Selektionsausgabe dargestellt. Die Beschreibung des Buttons „Zurücksetzen“ folgt in Kapitel 4.6.3.

Ausgabe im Graphen

Für die graphische Darstellung des Werts des Selektionsobjekts wird die Funktion „addDataToChart()“ verwendet.

```

1 addDataToChart(object) {
2     this.selectedObj = object;
3     for (let key in this.chartDataset) {
4         this.chartDataset[key].chartOption.legend.data[3] = this.selectedObj
5             .properties.kfkz;
6         this.chartDataset[key].chartOption.series[3] = { ... };
7         this.chartDataset[key].instance.setOption(this.chartDataset[key].
8             chartOption);
9     };
10    ...
11 }

```

Code 4.29: Anpassung der Optionen der Charts (EIGENE)

Es wird über das Array „chartDataset“ iteriert, welches die Informationen zu den darzustellenden Graphen der unterschiedlichen Attribute beinhaltet. Die chartOption-Eigenschaft der Objekte wird verändert, indem an vierter Stelle (Index 3) ein weiterer Datensatz hinzugefügt wird. Dieser umfasst die Daten des selektierten Kauffalls. Über die echarts-Methode „setOption()“ wird die Veränderung durchgeführt und die Graphen angepasst. Eine Wiederholung für die Graphen der Residuen und Cook-Distanzen findet unter Verwendung des Arrays „chartDatasetReg“ statt.

Die Abbildung 4.12 zeigt den um die Daten des Selektionsobjekts erweiterten Graphen für das Attribut „Baujahr“. Die Farbe der Kurve entspricht der des Selektionsobjekts in der Karte. Dem Nutzer entsteht auf diese Weise der Eindruck der Zusammengehörigkeit.

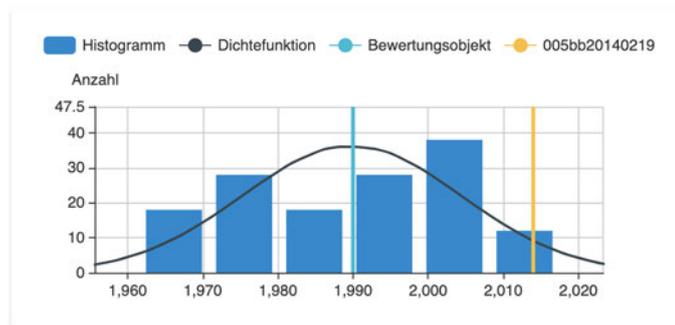


Abbildung 4.12: Darstellung des Graphen nach Einbinden des Selektionsobjekts (EIGENE)

4.6.3 Zurücksetzen der Selektion

Der Button „Zurücksetzen“ wird innerhalb der Ausgabe der Werte des Selektionsobjekts implementiert:

```
1 <button mat-stroked-button color="primary" (click)="reset()">Zuruecksetzen</button>
```

Code 4.30: Button zum Zurücksetzen der Selektion (EIGENE).

Die Ausführung der Funktion „reset()“ findet in der Komponentenklass „ToolStatisticsComponent“ statt:

```
1 reset() {
2   this.mapStylingService.statisticsReset.emit(this.selectedObj.id);
3   this.selectedObj = null;
4   for (let key in this.chartDataset) {
5     this.chartDataset[key].chartOption.series[3] = {type: 'line', data:
6       null};
7     this.chartDataset[key].chartOption.legend.data[3] = null;
8     this.chartDataset[key].instance.setOption(this.chartDataset[key].
9       chartOption);
10  }
```

Code 4.31: Funktion zum Zurücksetzen der Selektion (EIGENE).

Innerhalb des Funktionsrumpfes wird die Variable „selectedObj“ mit dem Wert „null“ belegt. Die chartOption-Eigenschaften der Elemente der Arrays „chartDataset“ und „chartDatasetReg“ werden mit leeren Werten überschrieben, sodass die Darstellungen in den Graphen nicht weiter durchgeführt werden. Eine Subscription an der Variable „statisticsReset“ setzt die „selected“-Eigenschaft des feature-states in der MapboxGlComponent auf „false“ und passt die Darstellung der Karte an.

Sämtliche in Kapitel 4 beschriebene Code-Ausschnitte sind innerhalb der vollständigen Quellcode-Dateien im Anhang C.2 abgelegt.

5 Ergebnisdarstellung und Bewertung

Dieses Kapitel fasst die Resultate der in der Arbeit umgesetzten Schritte zur Optimierung der Visualisierung der Ergebnisse der Regressionsanalyse zusammen und stellt diese in einem Vergleich den Darstellungen und Lösungsansätzen aus dem Tool „Automatisiertes Vergleichsverfahren“ gegenüber.

5.1 Aufbau der Webanwendung

Das Ergebnis der in Kapitel 4 beschriebenen Schritte zur Implementierung der Visualisierung der Kauffallobjekte und der Berechnung von statistischen Kenngrößen ist in Abbildung 5.1 dargestellt. Diese zeigt die Webanwendung mit dem im Overlay geöffneten Statistik-Tool. Außerhalb des Overlays ist die Kartendarstellung mit den Kauffallobjekten zu sehen. Diese unterscheiden sich je nach Größe des Residuums in ihrer Farbe. Ein Objekt ist dabei in gelb dargestellt. Dieses wurde selektiert und ist durch die gesonderte farbliche Visualisierung gekennzeichnet.

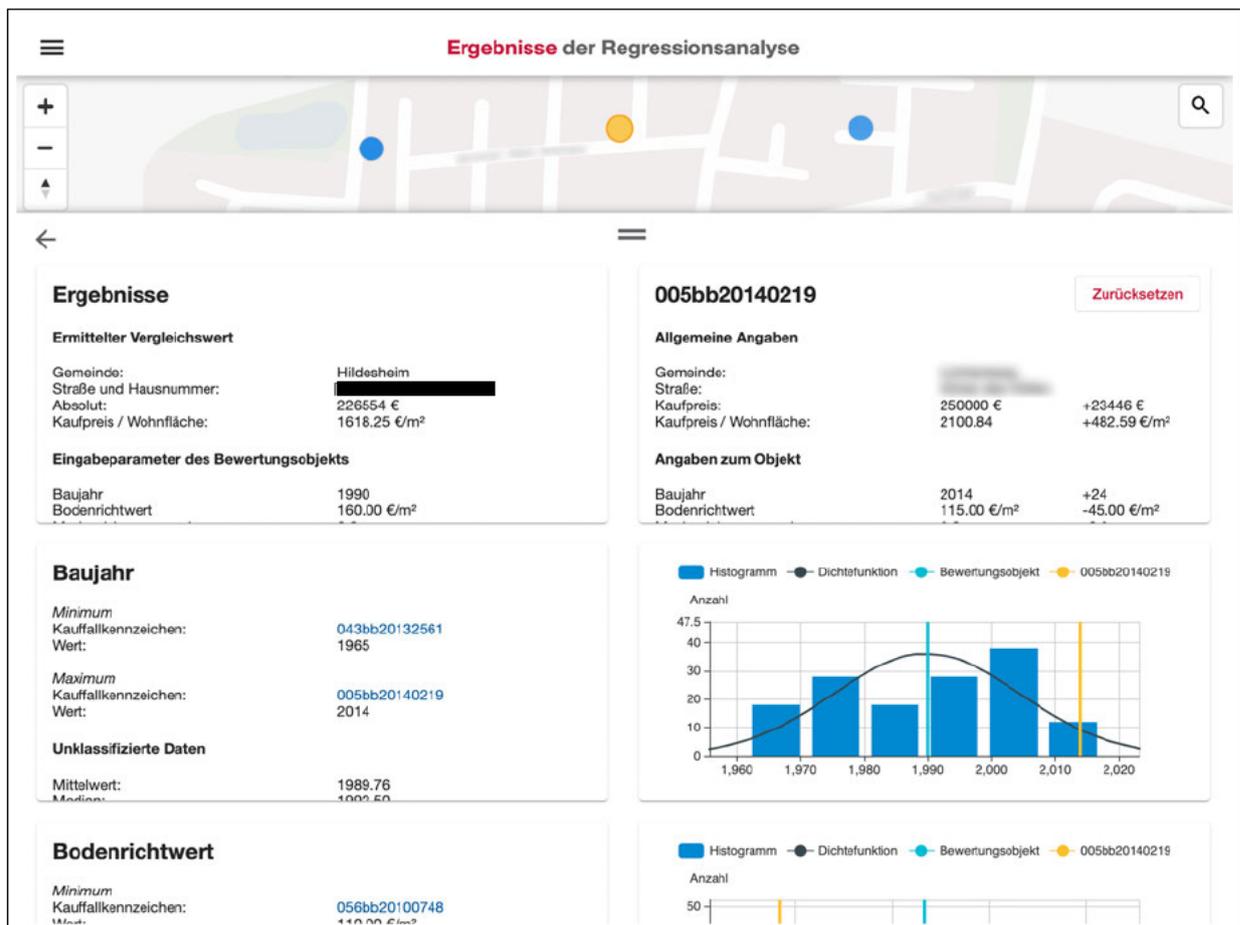


Abbildung 5.1: Gesamtansicht der Webanwendung mit geöffnetem Statistik-Tool (EIGENE)

Im Overlay selbst sind in den unterschiedlichen Cards die gegebenen bzw. bestimmten Parameter dargestellt. Die erste Zeile beinhaltet Informationen zum Bewertungsobjekt: die Eingangswerte, sowie die aus der Ausgleichung hervorgehenden Ergebnisse. Rechts daneben sind die Daten

des Selektionsobjekts mit den bestimmten Differenzen zum Bewertungsobjekt wiedergegeben. Es folgt die zeilenweise Auflistung der in der Ausgleichung verwendeten Attribute. Die linke Spalte gibt jeweils eine Übersicht zu den statistischen Kenngrößen. Rechts wird der zugehörige Graph dargestellt. Dieser beinhaltet, neben dem Histogramm und der Dichtefunktion der ermittelten Verteilung, eine Angabe zum Wert des Bewertungsobjekts und bei durchgeführter Selektion die Information dieses Objekts. In weiteren Cards aufgelistet sind die Informationen zu den aus der Ausgleichung hervorgehenden Daten: den Cook-Distanzen und den Residuen. Deren Graphen enthalten keine Information zum Bewertungsobjekt.

5.2 Vergleich zum Tool „Automatisiertes Vergleichswertverfahren“

Die folgenden Unterkapitel führen einen Vergleich der im Rahmen dieser Arbeit umgesetzten Funktionalitäten und Darstellungsformen der Webanwendung mit denen des bestehenden Tools „Automatisiertes Vergleichswertverfahren“ durch. In diesem wurden bereits Visualisierungen umgesetzt, sodass ein Vergleich positive und negative Erkenntnisse dieser Arbeit aufzeigen kann.

5.2.1 Kartendarstellung

Im Karteninhalt werden sämtliche in der Ausgleichung verwendeten Kaufobjekte dargestellt. Das Tool „Automatisiertes Vergleichswertverfahren“ stellt an dieser Stelle lediglich bis zu zehn Objekte, sowie das Bewertungsobjekt dar.

Dynamische Anpassung der Darstellung

Im Berechnungstool können die Objekte jedoch durch An- oder Abwahl in der zugehörigen Tabelle ein- bzw. ausgeblendet werden. Dies hat zur Folge, dass ein aus diesen Kauffällen dynamisch ermittelter Vergleichswert berechnet wird und simultan die verwendeten Kauffälle dargestellt werden. Eine Funktion, die im Rahmen dieser Arbeit nicht in die Webanwendung implementiert wurde.

Die Darstellung aller Kauffallobjekte ist als positiv zu betrachten. Das Fehlen der Funktionalität zur interaktiven Berechnung des Vergleichswerts ist nachteilig zu beurteilen. Es wird das Problem deutlich, dass es sich um zwei getrennte Anwendungen handelt. Eine Kombination der beiden Lösungen ist notwendig, sodass die Kartenfunktionen und die damit verbundenen Berechnungen gebündelt vorliegen. Diese kann durch die Migration der Visualisierungskomponente der hier entwickelten Lösung in das Tool „Automatisiertes Vergleichswertverfahren“ realisiert werden. Die Modularität der Komponenten einer Angular-Anwendung stellen die Grundlage dar. Eine weitere Option wird durch das Schaffen einer geeigneten Schnittstelle zwischen den Anwendungen geboten, die jedoch wiederum das Nachführen der beschriebenen Funktion in die Webanwendung erfordert.

Datengrundlage

Durch die Grundlage der in der Karte verwendeten Daten können aus der Kartendarstellung weitere Informationen erfasst werden. Da es sich unter anderem um ALKIS-Daten handelt, werden (maßstabsabhängig) Flurstücke und Gebäude angezeigt. Die Abbildung 5.2 zeigt die kombinierte Darstellung. Über die Lage des Kauffalls kann das zugehörige Flurstück ermittelt werden. Zu diesem können die auf ihm stehenden Gebäude erfasst werden. Es entsteht ein Eindruck über die Lage und Form des Flurstücks, der mit dem des Bewertungsobjekts verglichen werden kann.



Abbildung 5.2: Darstellung eines Kauffalls mit Flurstücken und Gebäuden (EIGENE)

Optimiert werden kann die Art der Darstellung durch eine flurstückstreue Abbildung der Kauf-fallobjekte in großen Maßstäben. Diese würde verhindern, dass die Kreise falschen Flurstücken zugeordnet werden. Auf diese Umsetzung wurde verzichtet, da die Information über die punktför-mige Lage für erste Visualisierungen hinreichend ist. Außerdem kann sich ein verkaufte Grund-stück aus mehreren Flurstücken zusammensetzen. Dies ist aus der hier gewählten Darstellung nicht ableitbar. In einer weiteren Arbeit an diesem Projekt könnte diese Thematik behandelt und implementiert werden.

5.2.2 Histogramm

Für die Darstellung der Histogramme wurden die in der Ausgleichung für die Ermittlung der Ko- effizienten verwendeten Kauffälle genutzt und nicht, wie im Berechnungstool „Automatisiertes Vergleichswertverfahren“, die Objekte der ersten Selektionsauswahl. Es sind lediglich die Häufig- keiten der Kauffalldaten von Interesse, die tatsächlich verwendet wurden. Durch das Nutzen einer bewährten Berechnungsvorschrift für die Bestimmung der Klassenbreite der einzelnen Attribute, lassen sich passende Verteilungen bestimmen und darstellen. Dies war bei den im Berechnungs- tool ermittelten Histogrammen nicht der Fall. Wird in diesem die Darstellung der Werte der Standardstufe betrachtet (Abbildung 5.3, links), fällt auf, dass sich die aufeinanderfolgenden Klassen deutlich in ihrer Häufigkeit unterscheiden. Dies liegt an der Wahl der Klassenbreite. Da dieser Wert kleiner ist als die Schrittweite der Werte der Standardstufe, können nicht allen Klas- sen Werte zugeordnet werden. Die Histogrammdarstellung in der hier entwickelten Anwendung ist im direkten Vergleich gleichmäßiger und konstanter (Abbildung 5.3, rechts).

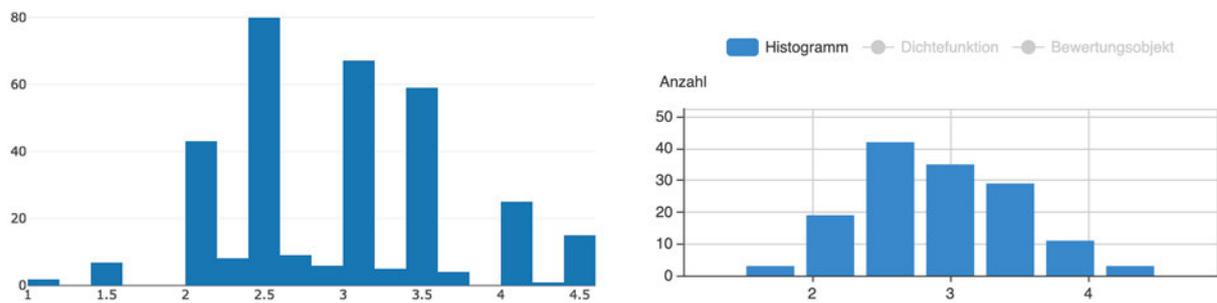


Abbildung 5.3: Vergleich der Histogrammdarstellungen (EIGENE)

Da in der Webanwendung zusätzlich Informationen zum Bewertungsobjekt und einem Selektionsobjekt angezeigt werden können, ist ein weiterer Informationsgewinn zu festzustellen. Dieser wird erweitert, da sich eine genaue Angabe zur Anzahl der Klasselemente in einem PopUp bei einem „mouseover“-Event anzeigen lässt.

5.2.3 Selektion

Es lassen sich, im Gegensatz zum Tool „Automatisiertes Vergleichsverfahren“, gezielt Attributwerte einzelner Kauffälle betrachten. Die durchzuführende Selektion lässt das konkrete Suchen nach Objekten zu. Die Möglichkeit der Selektion der Kauffälle mit minimalem bzw. maximalem Wert der jeweiligen Attribute gibt einen Eindruck über die Objekte, die sich im Bereich der größten Abweichungen befinden. Die Selektion der Kauffälle aus der Karte heraus bietet die Möglichkeit die Lage des Grundstücks und die Größe des Residuums zu berücksichtigen. Die nach diesem Attribut klassifizierten und unterschiedlich dargestellten Objekte geben direkt einen Eindruck über die Größe des Werts. Außerdem wird bei einem „mouseover“-Event ein PopUp mit dem jeweiligen Kaufpreis pro Quadratmeter Wohnfläche, sowie dem Residuum des Kauffalls dargestellt, sodass die Suche nach Objekten konkretisiert wird. Die kombinierte Darstellung des Selektionsobjekts und des Bewertungsobjekts im Graphen, sowie die Gegenüberstellung und Abbildung der Differenzen zwischen den Kaufobjekten, geben, neben der Einordnung in die Gesamtheit, einen Überblick der beiden Objekte zueinander. Es kann eine Interpretation bezüglich der Ähnlichkeit bzw. des Unterschied stattfinden.

Das gezielte Suchen nach Kauffällen kann durch die Einführung von Filtern weiter verbessert werden. Diese könnten, durch die Festsetzung von Schranken, Objekte in der Karte ausblenden, die die gegebenen Bedingungen nicht erfüllen.

Im digitalen Anhang C.3 ist ein Beispielvideo enthalten, das die Hauptfunktionalitäten der Anwendung zeigt.

6 Fazit

Das Ziel dieser Arbeit, die Visualisierungskomponente bei der Darstellung von Ergebnissen einer Regressionsanalyse zu optimieren und dadurch die Suche nach geeigneten Kaufobjekten für die Bestimmung des Vergleichswerts durchzuführen, wurde in vielen Teilen erfüllt.

Die Regressionsanalyse wurde nach dem Vergleichswertverfahren im Tool „Automatisiertes Vergleichswertverfahren“ durchgeführt. Dessen Ergebnisse konnten in JSON- und GeoJSON-Dateien ausgegeben werden, die anschließend in der Webanwendung statisch eingebunden wurden. Für die Darstellung in der Karte sind die tatsächlich in der Ausgleichung verwendeten Kauffallobjekte verwendet und diese in Abhängigkeit von der Größe des jeweiligen Residuums farblich abgebildet worden. Dieses Vorgehen zur Visualisierung sorgt für eine gute Lesbarkeit der Karte bei hohem Informationsgehalt. Die Darstellung der Daten in Graphen ist durch die Möglichkeit der dynamischen Veränderung bei einer durchführbaren Selektion ein Fortschritt zur reinen Ausgabe der Histogramme. Außerdem wird die Art der Verteilung, sowie der Wert des Bewertungsobjekts in den jeweiligen Graphen abgebildet. Durch dieses und die zusätzliche Berechnung und Ausgabe von statistischen Kenngrößen ist ein Informationsgewinn zu verzeichnen. Die statische Implementierung der Ergebnisse der Regressionsanalyse in der Webanwendung ist jedoch als Nachteil zu betrachten. Zwei der drei verfügbaren Export-Dateien wurden verwendet, von denen eine durch nachträgliche Anpassung verändert und erweitert wurde, sodass sie entsprechend des Verwendungszwecks genutzt werden konnte. Auch das Fehlen von Funktionen, die im Tool „Automatisiertes Vergleichswertverfahren“ vorhanden und notwendig für die Berechnung des Vergleichswerts sind, ist negativ zu beurteilen.

Die Weiterentwicklung des Prototyps zu einer Komponente für die Visualisierung der Ergebnisse einer Regressionsanalyse kann somit als Teilerfolg angesehen werden. Die umgesetzten Berechnungen und Darstellungen stellen gute Ansätze für die Suche nach geeigneten Kauffällen bei der Bestimmung des Vergleichswerts dar. Einige Funktionen und Visualisierungsmethoden müssen noch integriert werden. Eine Migration der Anwendung in das Tool „Automatisiertes Vergleichswertverfahren“ kann durch das modulare Konzept der Angular-Anwendung durchgeführt werden.

7 Ausblick

Für die Weiterentwicklung der Webanwendung ist die Verschneidung der Export-Daten aus dem Tool „Automatisiertes Vergleichsverfahren“, den Geobasisdaten aus ALKIS, sowie den Kaufalldaten der Automatisierten Kaufpreissammlung (AKS) durchzuführen. Eine Abbildung der darzustellenden Daten, den Kauffällen der AKS, wurde bereits in der Anwendung implementiert. Beim Hinzufügen der Ergebnisse der Regressionsanalyse wird eine GeoJSON-Datei als neuer Layer hinzugefügt. Dieser Schritt müsste übersprungen werden und die in der Datenbank vorliegenden Objekte müssten über die entsprechende ID, das Kauffallkennzeichen, gesondert dargestellt werden, sodass eine Unterscheidung möglich ist. Das Problem bei der Umsetzung ist die Verwendung von Vector-Tiles, da den Objekten in den Kacheln keine ID zugewiesen ist, auf die zugegriffen werden kann und unter dessen Verwendung ein Styling mit den entsprechenden Parametern aufgesetzt werden kann. Hier setzt eine Forschungsaufgabe zur Lösung dieses Problems an.

Eine weitere Möglichkeit zur Optimierung der Applikation wird durch die „User Experience“ geboten. Diese „hat zum Ziel, das Gesamterlebnis zu optimieren, das mit der Nutzung von Diensten oder Produkten zusammenhängt“ (RICHTER et al., 2013, S. 156). Der Fokus bei der Implementierung der Anwendung wird über die reine Funktionalität hinaus ausgeweitet, sodass die Aspekte des Designs und der Ästhetik berücksichtigt werden (RICHTER et al., 2013, S. 156). Eine Beachtung der User Experience ist relevant, wenn es sich um öffentlich verfügbare Anwendungen handelt. Die Nutzer sollen die Anwendung wiederholt besuchen und weiterempfehlen. Dies tun sie, wenn sie selbst eine positive Erfahrung bei der Nutzung erfahren haben. Wird die hier entwickelte Anwendung als Grundlage für eine offene Applikation zur Berechnung von Verkehrswerten verwendet, sind die Grundsätze der User Experience zu berücksichtigen und anzuwenden.

Literatur

- ADV (2018). *Tätigkeitsbericht 2017/2018*. Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland. Abgerufen am 18.12.2019 unter <http://www.adv-online.de/Veroeffentlichungen/Taetigkeitsberichte-der-AdV/>.
- BECKER, B. (1993). *Statistik*. R. Oldenbourg Verlag München Wien.
- BEHREND, E. (2013). *Elementare Stochastik*. Spektrum Verlag | Springer Fachmedien Wiesbaden.
- BISCHOFF, B. (2015). *Grundstückswertermittlung*. In: Schmoll genannt Eisenwerth, F. (Hg.): *Basiswissen Immobilienwirtschaft*. GRUNDEIGENTUM-VERLAG GmbH, Berlin und Verlag Franz Vahlen GmbH, München. 3. vollständig aktualisierte und erweiterte Auflage.
- BÖSELT, M. (1999). *Statistik*. R. Oldenbourg Verlag München Wien. 2., ergänzte Auflage.
- BUTLER, H., HOBUS INC., DALY, M., CADCORP, DOYLE, A., GILLIES, S., MAPBOX, HAGEN, S., SCHAUB, T., und PLANET LABS (2016). *The GeoJSON Format*. Internet Engineering Task Force (IETF). Standards Track. Abgerufen am 18.12.2019 unter <https://tools.ietf.org/html/rfc7946>.
- CAVAZZI, S. (2018). *OGC Testbed-13: Vector Tiles Engineering Report*. Open Geospatial Consortium. Public Engineering Report. Abgerufen am 18.12.2019 unter <http://www.opengis.net/doc/PER/t13-DS001>.
- DEMARY, M. (2009). *Hedonische Immobilienpreisindizes - Verfahren und Beispiele*. IW-Trends - Vierteljahresschrift zur empirischen Wirtschaftsforschung aus dem Institut der deutschen Wirtschaft Köln, 36. Jahrgang, Heft 3/2009. Deutscher Insitutiv-Verlag, Köln.
- ECKSTEIN, P. P. (2012). *Statistik für Wirtschaftswissenschaftler - Eine realdatenbasierte Einführung mit SPSS*. Gabler Verlag | Springer Fachmedien Wiesbaden. 3. überarbeitete und erweiterte Auflage.
- FAHRMEIR, L., KNEIB, T., und LANG, S. (2009). *Statistik und ihre Anwendungen - Regression - Modelle, Methoden und Anwendungen*. Springer-Verlage Berlin Heidelberg. Zweite Auflage.
- GAFFURI, J. (2012). *Toward Web Mapping with Vector Data*. In: *Geographic Information Science, 7th International Conference*, pp. 87-101. GIScience, Columbus, OH, USA.
- GAG (2016). *Grundstücksmarktbericht 2016 für die Bereiche der kreisfreien Städte Emden und Wilhelmshaven und der Landkreise Aurich, Friesland, Leer und Wittmund*. Gutachterausschuss für Grundstückswerte Aurich.

- GEOJSON.ORG (2019). *GeoJSON*. Abgerufen am 18.12.2019 unter <https://geojson.org/>.
- GOOGLE (2019a). *Card*. Abgerufen am 08.01.2020 unter <https://material.angular.io/components/card/overview>.
- GOOGLE (2019b). *EventEmitter*. Abgerufen am 20.12.2019 unter <https://angular.io/api/core/EventEmitter>.
- GOOGLE (2019c). *HttpClient*. Abgerufen am 23.12.2019 unter <https://angular.io/guide/http>.
- GOOGLE (2019d). *Introduction to modules*. Abgerufen am 06.01.2020 unter <https://angular.io/guide/architecture-modules>.
- GOOGLE (2019e). *Routing & Navigation*. Abgerufen am 06.01.2020 unter <https://angular.io/guide/router>.
- GOOGLE (2019f). *Template Syntax*. Abgerufen am 20.12.2019 unter <https://angular.io/guide/template-syntax>.
- GRUBER, F. J. und JOECKEL, R. (2009). *Formelsammlung für das Vermessungswesen*. Vieweg + Teubner | GWV Fachverlage GmbH, Wiesbaden. 14. Auflage.
- HÖLLER, C. (2017). *Angular - Das umfassende Handbuch*. Rheinwerk Verlag GmbH, Bonn.
- JSON.ORG (2019). *Einführung in JSON*. Abgerufen am 18.12.2019 unter <https://www.json.org/json-de.html>.
- KALBERER, P. (2019). *t-rex*. Abgerufen am 18.12.2019 unter <https://t-rex.tileservers.ch/>.
- LGLN (2019a). *Basisvisualisierung; hier: Einrichtung einer Folgeprojektgruppe „Basisvisualisierung II“*. Landesamt für Geoinformation und Landesvermessung Niedersachsen. Projektauftrag, Internes Dokument.
- LGLN (2019b). *Handbuch für die Führung und Auswertung der Automatisierten Kaufpreissammlung*. Landesamt für Geoinformation und Landesvermessung Niedersachsen. Internes Dokument.
- LI, C., LU, H., XIANG, Y., LIU, Z., YANG, W., und LIU, R. (2018). *Bringing Geospatial Data Closer to Mobile Users: A Caching Approach Based on Vector Tiles for Wireless Multihop Scenarios*. Mobile Information Systems Volume 2018, Article ID 5186495. Research Article.
- MAPBOX (2019a). *Mapbox GL JS*. Abgerufen am 18.12.2019 unter <https://docs.mapbox.com/mapbox-gl-js/api/>.
- MAPBOX (2019b). *Style Specification*. Abgerufen am 18.12.2019 unter <https://docs.mapbox.com/mapbox-gl-js/style-spec/>.

- MITTAG, H.-J. (2012). *Statistik - Eine interaktive Einführung*. Springer-Verlage Berlin, Heidelberg. 2., wesentlich überarbeitete Auflage.
- POSTGIS (2019). *PostGIS - Spatial and Geographic objects for PostgreSQL*. Abgerufen am 18.12.2019 unter <https://postgis.net/>.
- RICHTER, M., FLÜCKIGER, M. D., und ZÜHLKE ENGINEERING AG (2013). *Userbility Engineering kompakt - Benutzbare Produkte gezielt entwickeln*. Springer-Verlag Berlin Heidelberg. 3. Auflage.
- SACHS, L. und HEDDERICH, J. (2006). *Angewandte Statistik - Methodensammlung mit R*. Springer-Verlage Berlin, Heidelberg. Zwölfte, vollständig neu bearbeitete Auflage.
- SCOTT, D. W. (1979). *On optimal and data-based histograms*. In: *Biometrika*, Vol. 66, No. 3 (Dec. 1979), pp. 605-610. Oxford University Press on behalf of Biometrika Trust.
- SIMON, T. und GILICH, T. (2012). *Wertermittlung von Grundstücken - Aufgaben und Lösungen zur Verkehrswertermittlung*. Werner Verlag, Wolters Kluwer Deutschland GmbH, Köln. 6., überarbeitete und erweiterte Auflage.
- SOMMER, G. und KRÖLL, R. (2013). *Lehrbuch zur Immobilienbewertung - Unter Berücksichtigung der ImmoWertV und der Sachwert-Richtlinie*. Werner Verlag, Wolters Kluwer Deutschland GmbH, Köln. 4. Auflage.
- SOOT, M. und WEITKAMP, A. (2019). *Kurzanleitung zum automatisierten Vergleichswertverfahren*. Technische Universität Dresden, Fakultät Umweltwissenschaften. Internes Dokument.
- STRUNZ, M. (2012). *Instandhaltung - Grundlagen, Strategien, Werkstätten*. Springer-Verlag Berlin Heidelberg.
- THE APACHE SOFTWARE FOUNDATION (2019). *echarts - API*. Abgerufen am 10.01.2020 unter <https://echarts.apache.org/en/api.html>.
- URBAN, D. und MAYERL, J. (2011). *Regressionsanalyse: Theorie, Technik und Anwendung*. VS Verlag für Sozialwissenschaften | Springer Fachmedien Wiesbaden GmbH. 4., überarbeitete und erweiterte Auflage.
- WOIWODE, G., MALCHER, F., KOPPENHAGEN, D., und HOPPE, J. (2018). *Angular - Grundlage, fortgeschrittene Techniken und Best Practices mit TypeScript*. dpunkt.verlag GmbH, Heidelberg. 1., korrigierter Nachdruck.
- ZIEMS, M., Martin, F., und Staufienbiel, K.-J. (2019). *PG Basisvisualisierung - Team Wissensaufbau*. Präsentation, Internes Dokument.

A Report: Bewertung

Wertermittlung

Der Vergleichswert des Einfamilienhauses in Hildesheim, [REDACTED] beträgt rund

227.000 €*

und liegt mit einer Wahrscheinlichkeit von 95 % im Bereich von 151.617 € - 301.492 €.

Darin enthalten sind 0 € besondere objektspezifische Grundstücksmerkmale.

Vergleichskauffälle

Tabelle 1: Vergleichskauffälle mit der Nutzung Einfamilienhaus

ID	Gemeinde	Straße	Bodenrichtwert	Wohnfläche	Baujahr	Datum	Kaufpreis/Wohnfläche*
V	Hildesheim, [REDACTED]	Straße		140	1990		
1	Itzum	Elsa-Brändström-Weg	175	150	1985	2016-11-10	[REDACTED]
2	Itzum	Am Nottbohm	165	135	1985	2017-09-28	[REDACTED]
3	Ochtersum	Plötzenstraße	160	150	1978	2015-04-09	[REDACTED]
4	Ochtersum	Sohldfeld	150	138	1995	2014-04-09	[REDACTED]
5	Itzum	Hausbergring	145	154	1985	2012-12-10	[REDACTED]

Die Vergleichskauffälle liegen zwischen 1.079 €/qm und 2.542 €/qm und bestätigen somit den aus dem hedonischen Modell abgeleiteten Vergleichswert.

HINWEIS: Die dargestellte Kaufpreisliste entspricht nicht den vom Programm vorgeschlagenen Kaufpreisen.

Zeitliche Entwicklung

Preisliche Entwicklung



* Entspricht etwa 1.618 €/qm Wohnfläche

Lage

Später aus ALKIS

Entfernungen

Später aus Daseinsvorsorgeatlas

Auswirkung von Variationen in den Parametern

Parameter	Baujahr	Bodenrichtwert [€/qm]	Restnutzungsdauer [a]	Modernisierungsgrad	Standardstufe
Ausprägung	2000	176	49	7	2.2
Vergleichswert	281000	238000	199000	222000	237000
Ausprägung	1980	144	40	5	1.8
Vergleichswert	172000	215000	249000	231000	216000

Prognose**Abschließende Qualitätsbeurteilung**

-  Der Analyseprozess weist keine Auffälligkeiten auf.
- 

B Report: Modellbeschreibung

Modellbeschreibung

zur Bewertung

Hildesheim, [REDACTED]

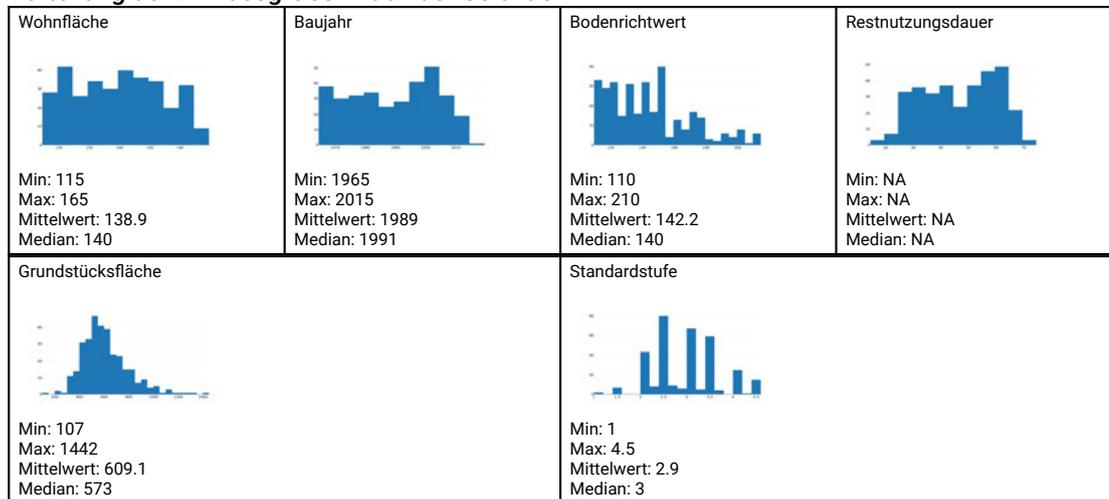
Vergleichskauffälle (Gesamte Auswahl)

Tabelle 1: Alle dem Nutzer zur Verfügung stehenden Kauffälle für den Teilmarkt Einfamilienhaus

ID	Gemeinde	Straße	Bodenrichtwert	Wohnfläche	Baujahr	Datum	Kaufpreis/Wohnfläche*
V	Hildesheim, [REDACTED]	Straße		140	1990		
1	Itzum	Sensburger Ring	175	140	1982	2016-12-28	[REDACTED]
2	Itzum	Elsa-Brändström-Weg	175	150	1985	2016-11-10	[REDACTED]
3	Itzum	Am Nottbohm	165	135	1985	2017-09-28	[REDACTED]
4	Ochtersum	Plötzenstraße	160	150	1978	2015-04-09	[REDACTED]
5	Ochtersum	Sohldfeld	150	138	1995	2014-04-09	[REDACTED]
6	Itzum	Martin-Boyken-Ring	165	160	1988	2015-03-05	[REDACTED]
7	Ochtersum	Am Wallhof	170	160	1980	2015-03-05	[REDACTED]
8	Ochtersum	Schenkenbrühl	150	135	2005	2015-03-23	[REDACTED]
9	Itzum	Hausbergring	145	154	1985	2012-12-10	[REDACTED]
10	Ochtersum	Plötzenstraße	160	160	1977	2014-03-24	[REDACTED]

* Die angegebenen Preise wurden durch die im hedonischen Modell abgeleitete konjunkturelle Entwicklung auf den Wertermittlungsstichtag fortgeschrieben.

Verteilung der Einflussgrößen nach der Selektion



Regressionsmodell

In der Regressionsanalyse wurden 142 Kauffälle genutzt. Das Regressionsmodell besteht aus folgenden Einflussgrößen:

Parameter	Intercept	Baujahr	Bodenrichtwert [€/qm]	Restnutzungsdauer [a]	Modernisierungsgrad	Standardstufe
Koeffizient	-75316.2306	38.8453	5.0081	-39.5718	-29.8874	378.0209
t-Wert	-5.5379	5.5323	5.5404	-5.0413	-4.7057	6.8017
p-Wert	1.5264e-7	1.5664e-7	1.5082e-7	0.0000014479	0.0000061483	2.9944e-10

Das Regressionsmodell ist linear und weist keine Transformationen von Größen auf. Das Regressionsmodell ist ausschließlich für das spezifizierte Wertermittlungsobjekt gültig.

Güte der Regression

- Das Bestimmtheitsmaß beträgt: 0.6884.
- Das adjustierte Bestimmtheitsmaß beträgt: 0.6769.
- Die Residuen sind mit einer Irrtumswahrscheinlichkeit von 5 % normalverteilt (Hypothese des Shapiro-Wilk-Test angenommen (p-Value = 0.3984 > 0.05)).
- Die Regression weist mit 5 % Irrtumswahrscheinlichkeit Homoskedastizität auf (Hypothese des Breusch-Pagan-Tests angenommen (p-Wert = 0.06 > 0.05)).
- Es besteht keine Multikollinearität zwischen den Größen (maximaler Variationsinflationsfaktor = 0 < 7).

Prognose

Es kann keine Prognose erstellt werden. Im aktuellen Selektionszeitraum ist der Kaufzeitpunkt nicht signifikant bestimmt worden.

C Digitaler Anhang

Die der Bachelorarbeit beigelegte CD enthält den digitalen Anhang dieser Arbeit. Dieser umfasst den implementierten Quellcode, nicht veröffentlichte, interne Dokumente, sowie sonstige zur Umsetzung erforderliche Daten.

Dieses Kapitel enthält eine Übersicht zum Aufbau der Ordnerstruktur der CD.

C.1 Export-Dateien

Im Ordner *1 Export-Dateien* befinden sich die im Tool „Automatisiertes Vergleichswertverfahren“ erstellten Dateien, die die Selektion (1-1 Selektion), die GeoJSON-Daten (1-2 GeoJSON), sowie die Ergebnisse der Ausgleichung (1-3 Ergebnis) beinhalten.

C.2 Quellcode

Die implementierten Programmcodes sind im Ordner *2 Quellcode* bereitgestellt.

Der Unterordner *2-1 Frontend* beinhaltet dabei, neben der neu erstellten Komponente, die bereits vorhandenen Dateien der Angular-Anwendung, die als Grundlage für die weitere Umsetzung im Rahmen dieser Arbeit verwendet wurden. Die folgende Übersicht gibt einen Überblick über die erweiterten und neu angelegten Dateien und an welcher Stelle sie zu innerhalb der Ordnerstruktur zu finden sind. Die Stellen im Code, an denen bestehende Dateien erweitert wurden, sind durch die Kommentare „Bearbeitung Anfang“ und „Bearbeitung Ende“ gekennzeichnet.

app → app.module.ts	erweitert
app → map → map.component.ts	erweitert
app → map → map-styling.service.ts	erweitert
app → map → mapbox-gl → mapbox-gl.component.ts	erweitert
app → map → tools → tool-statistics → tool-statistics.component.html	neu
app → map → tools → tool-statistics → tool-statistics.component.scss	neu
app → map → tools → tool-statistics → tool-statistics.component.ts	neu

Unter *2-2 Backend* ist die Datei des Python-Dienstes abgelegt.

C.3 Beispielvideo

Der Ordner *3 Beispielvideo* beinhaltet eine Aufzeichnung des Viewports bei der Verwendung der Applikation. Es werden die wesentlichen implementierten Funktionalitäten vorgestellt. Das Beispielvideo dient einer besseren Verständlichkeit der in der Arbeit erläuterten Visualisierungen.

C.4 Interne Dokumente

Die Dokumente zu den Einträgen im Literaturverzeichnis, die den Zusatz „Internes Dokument“ besitzen, sind im Ordner *4 Interne Dokumente* abgelegt.