Kapitel 2

Die Daten verstehen

In diesem Kapitel wollen wir einen Schritt besprechen, der bei allen Analysen und damit auch bei der Analyse von Immobiliendaten mithilfe des Modells der hedonischen Preise wichtig ist, nämlich dem, die vorhandenen Daten zu verstehen. Ein gutes Verständnis der Daten gibt uns einerseits eine solide Grundlage für die Analyse, weil wir Vertrauen in die Qualität und Validität unserer Daten haben, und hilft uns andererseits auch bei den einzelnen Schritten der Analyse, weil wir wissen, wie die Werte der einzelnen Variablen verteilt sind und welche Zusammenhänge möglicherweise zwischen ihnen bestehen.

Dabei werden wir uns auch mit einigen grundlegenden Konzepten der sogenannten deskriptiven Statistik beschäftigen. Diese Form der Statistik entwickelt Maßzahlen und Verfahren, die die in einzelnen Variablen enthaltene Information verdichten und uns sagen, wo die Werte der Variablen liegen, wie stark sie streuen, ob sie mit anderen Variablen zusammen hängen, etc. Einführende Darstellungen in die deskriptive Statistik finden sie etwa in Rumsey (2005), Bomsdorf (2013), Benesch (2012), Bourier (2013), Kuckartz (2013), Jarman (2013) oder Olbricht (2013). Formal etwas anspruchsvoller ist Steland (2013).

Bevor wir uns genauer mit den Daten auseinandersetzen können, müssen wir noch einen konzeptuellen Punkt klären. Das ist der Punkt, was wir unter den Daten verstehen sollen.

2.1 Daten kommen als Stichprobe aus einer Grundgesamtheit

Bei alle statistischen Verfahren und damit auch bei jeder Anwendung der Methode der hedonischen Preise gehen wir davon aus, dass die uns vorliegenden Daten eine *Stichprobe* sind, die aus einer dahinter liegenden *Grundgesamtheit* stammt. Die Frage, wie die Stichprobe aus der Grundgesamtheit extrahiert wird, bildet die Basis für einen eigenen Teilbereich der Statistik, der Stichprobentheorie (z.B. Thompson, 2002; Tillé, 2006; Dürr und Mayer, 2013), auf den wir hier nicht wirklich eingehen wollen. Etwas später werden wir uns kurz mit den praktischen Aspekten der Frage, wo die Daten her kommen, beschäftigen.

Wichtig in diesem Zusammenhang ist, dass mit jeder Stichprobe andere Beobachtungen aus der Grundgesamtheit gezogen werden. Damit stellen unsere Daten immer nur einen Ausschnitt aus der Wirklichkeit dar. Eine andere Stichprobe würde einen anderen Ausschnitt zeigen. Da sich alle Schätzergebnisse immer aus den Daten der Stichprobe errechnen, erhalten wir damit für jede neue Stichprobe auch etwas andere Schätzergebnisse.

Die Tatsache, dass die konkreten Werte unserer Daten von der Stichprobe abhängen, führt dazu, dass wir die einzelnen Variablen unseres Modells als sogenannte *Zufallsvariable* betrachten. Trotz ihres Namens ist eine Zufallsvariable eigentlich keine Variable, sondern eine Funktion (siehe etwa Blake, 1979; Holling und Gediga, 2013; Tappe, 2013). Etwas genauer werden wir auf Zufallsvariable in Kapitel 3 eingehen. Daher wollen wir hier keine präzise Definition, sondern nur eine intuitive Erklärung geben: In diesem Sinn ist eine Zufallsvariable eine Variable, deren konkrete Werte von

G. Maier, S. Herath, *Immobilienbewertung mit hedonischen Preismodellen,*
DOI 10.1007/978-3-658-02862-6_2, © Springer Fachmedien Wiesbaden 2015

einer Stichprobe abhängen. Dabei sind die Werte nicht völlig zufällig, sondern für einen bestimmten Wertebereich definiert. Beispiele für Wertebereiche sind etwa alle nicht-negativen ganzen Zahlen (wie oft passiert ein Unfall), die ganzen Zahlen zwischen eins und sechs (die Augenzahl beim Würfeln), „Kopf oder Zahl“ (beim Werfen einer Münze), alle nicht-negativen realen Zahlen (Kauf- oder Mietpreise von Immobilien), alle realen Zahlen (Jahresergebnis eines Unternehmens), usw. Werte außerhalb dieses Wertebereichs können auf keinen Fall eintreten, egal wie die Stichprobe aussieht. Für die Werte im Wertebereich gibt es eine bestimmte Chance dafür, dass dieser Wert in eine Stichprobe aufgenommen wird. Diese Chance wird durch eine *Wahrscheinlichkeit* ausgedrückt, über alle Werte des Wertebereichs spricht man von einer *Wahrscheinlichkeitsverteilung*.

Je nachdem, ob der Wertebereich diskrete Werte (z.B. Augenzahl beim Würfeln) oder stetige Werte (z.B. Preis, Miete, Unternehmenserfolg) enthält, spricht man von diskreten und stetigen Zufallsvariablen und diskreten und stetigen Verteilungen. In der Anwendung ist diese Unterscheidung sehr wichtig, weil für diese beiden Arten von Zufallsvariablen unterschiedliche Methoden existieren.

Ein Beispiel für eine Zufallsvariable ist die „Quadratmetermiete“. Es ist eine stetige Zufallsvariable, weil grundsätzlich zwischen zwei Werten des Wertebereichs weitere Werte liegen können. Da negative Werte für die Quadratmetermiete keinen Sinn machen, sind der Wertebereich die nicht-negativen realen Zahlen. Über die Wahrscheinlichkeitsverteilung der Quadratmetermiete können wir ohne weitere Analyse wenig aussagen. Weiter unten werden wir uns mit diesen Analysen beschäftigen. Die meisten Immobilien-Fachleute werden aber wahrscheinlich heute eine Quadratmetermiete von €10 für wahrscheinlicher halten als eine von €0,02 oder €5000. Das sagt zumindest aus, dass die Wahrscheinlichkeit für sehr niedrige und für sehr hohe Werte geringer ist als jene für Werte um €10.

Die konkreten Werte in den Daten nennt man übrigens *Realisierungen* der entsprechenden Zufallsvariablen. Die Tatsache, dass die Daten von der Stichprobe abhängen und die Analyseergebnisse von den Daten, führt zum Problem, dass wir uns bei den Analyseergebnissen nicht sicher sein können, dass sie bei anderen Stichproben so ausfallen würden. Ist etwa ein positiver Zusammenhang, den unsere Analysen liefern, nur auf die zufällige Datenauswahl unserer Stichprobe zurückzuführen, oder würden wir den auch bei Analysen mit anderen Stichproben beobachten? Das ist die Frage nach der *Signifikanz* unserer Ergebnisse. Sie wird uns an vielen Stellen des Buches beschäftigen. Grundsätzlich ist dazu zu sagen, dass wir deshalb, weil wir es mit Zufallsvariablen zu tun haben, nie eine hundertprozentige Sicherheit erlangen können. Wir können aber ein höheres oder niedrigeres Niveau an Signifikanz erreichen, also eine niedrigere oder höhere Gefahr, dass unser Ergebnis falsch ist.

2.2 Wie sind die Daten zustande gekommen?

Dass die Stichprobe Einfluss auf die Daten und damit auf die Ergebnisse unserer Analysen hat, haben wir oben schon besprochen. Das gilt auch für andere Aspekte der Datenerhebung. Unsere Analyseergebnisse können bestenfalls nur so gut sein, wie unsere Daten. Aus schlechten Daten können wir nie gute Analyseergebnisse ableiten.

Bevor wir mit Daten arbeiten, sollten wir uns daher noch etwas ausführlicher mit den Umständen der Datenerhebung beschäftigen. Hier können einige ernste Gefahren für die Qualität der Analyseergebnisse lauern.

Ein mögliches Problem ist jenes der *Selbst-Selektion*. Dabei betrachten wir ein Phänomen, das selbst einen Einfluss darauf hat, ob eine Beobachtung in die Stichprobe aufgenommen wird oder nicht. Ein klassisches Beispiel sind Untersuchungen über die Faktoren für den Erfolg von Unternehmensneugründungen. Wenn ich dabei von den zu einem bestimmten Zeitpunkt bestehenden Unternehmen ausgehe, so wird meine Stichprobe eher aus erfolgreichen Unternehmen bestehen, weil die erfolglosen bereits aus dem Markt ausgeschieden sind. Im Immobilienbereich sind etwa Aussagen über Einkommen auf der Grundlage einer Stichprobe von Wohnungseigentümern von dem Problem der Selbst-Selektion betroffen, weil Personen mit höheren Einkommen eher Wohnungseigentümer sind. Auch bei Untersuchungen auf der Basis von Mitgliederbefragungen tritt dieses

Problem regelmäßig auf und wird es meist ignoriert. Weil hinter der Mitgliedschaft üblicherweise eine bestimmte Motivation steht, selbst-selektieren sich die Akteure mit dieser Motivation eher in die Stichprobe als andere.

Ein eng damit verbundenes Problem ist jenes von *systematischer Antwortverweigerung*. Sie tritt besonders bei Phänomenen auf, die rechtswidrig, unmoralisch oder unangenehm sind. Wollen wir etwa Immobilienmakler über illegale Ablösezahlungen befragen, so werden jene, die illegale Ablösen verlangen, eher die Antwort verweigern als die anderen.

Probleme bei der Entstehung der Daten müssen sich aber nicht nur darin ausdrücken, dass die Stichprobe verzerrt ist. Sie können auch zu *systematisch verzerrten Daten* führen. Gerade bei der Immobilienbewertung haben wir immer mit diesem Problem zu kämpfen. Da sich Anbieter einer Immobilie am Markt einen Spielraum für Verhandlungen lassen wollen, liegen Angebotspreise am Immobilienmarkt üblicherweise über den bezahlten Preisen. Aber selbst wenn wir Zugriff auf die Kaufverträge bzw. die darin enthaltene Information haben, sollten wir uns nicht allzu sicher sein, dass unsere Daten nicht systematisch verzerrt sind. Da Steuern und Gebühren vom Kaufpreis abhängen, gibt es einen ökonomischen Anreiz dafür, einen Teil der Transaktion schwarz abzuwickeln. Damit sind die in den Kaufverträgen ausgewiesenen Preise wahrscheinlich niedriger als die tatsächlich bezahlten.

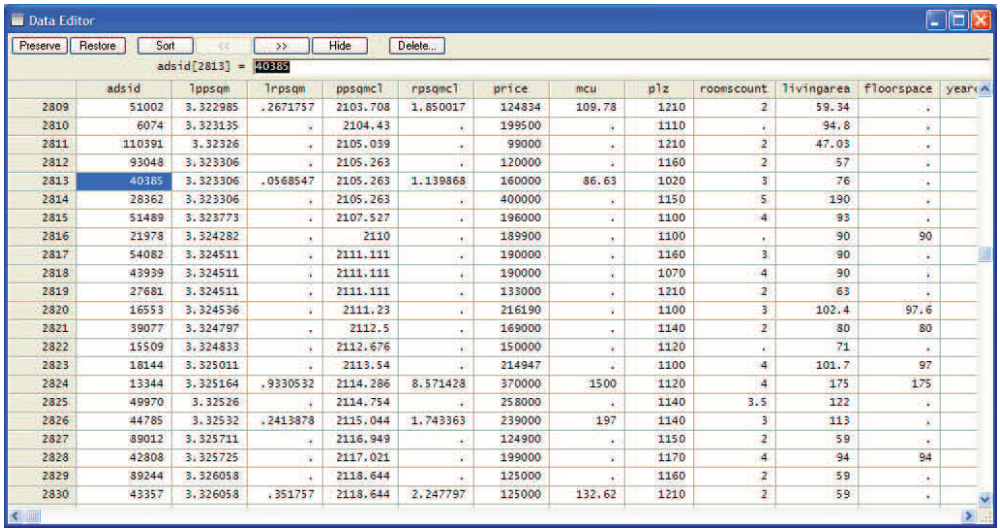
Für die meisten dieser potenziellen Probleme gibt es Möglichkeiten, in der Analyse darauf Rücksicht zu nehmen. Zum Teil können sie auch nur in der Interpretation der Ergebnisse berücksichtigt werden. Wichtig ist es, sich dieser potenziellen Probleme bewusst zu werden, und den eigenen Datenbestand daraufhin zu durchleuchten. „Augen zu und durch“, also die Gefahren zu ignorieren, ist sicher die schlechteste Variante. Sie führt zu unverlässlichen Analyseergebnissen.

2.3 Arten von Daten, Daten einlesen

Wir haben oben bereits erwähnt, dass es unterschiedliche Arten von Daten gibt. In Abschnitt 2.1 haben wir zwischen „diskreten“ und „stetigen“ Daten unterschieden. Die Ursache für diskrete Variable sind oft sogenannte *kategoriale* Daten. Diese bilden Kategorien von Beobachtungen ab. Bei Wohnungen etwa „Balkon“ und „kein Balkon“ oder „saniert“, „teilsaniert“, „nicht saniert“. Bei Einfamilienhäusern sind die Kategorien „unterkellert“ und „nicht unterkellert“ sicher wichtig.

Wenn Kategorien in dieser Form bezeichnet sind, so liegen sie IT-technisch als Zeichenketten (Strings) vor. Kategorien können aber auch nummerisch kodiert sein. Beispielsweise so, dass die Zahl 1 bedeutet, es gibt einen Balkon und die Zahl 0, es gibt keinen. Das ist aber nur eine Form der Kodierung. Die dahinter liegende Information, ob es einen Balkon gibt oder nicht, bleibt trotzdem kategorial. Die Kodierung mit 1 für „Balkon“ und 0 für „kein Balkon“ entspricht der Konvention. Wir könnten die beiden Kategorien aber auch mit 0 für „Balkon“ und 1 für „kein Balkon“ oder mit 1 für „Balkon“ und -1 für „kein Balkon“ kodieren. Wenn wir das bei der Interpretation der Schätzergebnisse berücksichtigen, so macht es keinen Unterschied, wie wir kategoriale Daten kodieren.

Die Kodierung ist notwendig, weil die statistischen Verfahren meistens nur mit Zahlen und nicht mit Zeichenketten arbeiten können. Es stellt sich aber die Frage, wann diese Kodierung in Zahlen vorgenommen werden soll. In früheren Zeiten wurden kategoriale Variable schon vor der Dateneingabe in sogenannten „Coding Sheets“ in Zahlen umkodiert und dann bereits als Zahlen erfasst. Das führte manchmal dazu, dass es schwierig war nachzuvollziehen, was eine 1 oder 0 nun bedeutet. Heute können alle Programme im Datenhandling sehr wohl mit Zeichenketten umgehen. Auch der Speicherplatz ist kein limitierender Faktor mehr. Daher ist es heute sicher eher zu empfehlen, kategoriale Variable als Zeichenketten zu erfassen und in die Statistiksoftware einzulesen und die Umwandlung in Zahlen – in neuen Variablen – erst dort vorzunehmen. Wertvolle Hilfen, die moderne Statistikprogramme anbieten, sind auch „Labels“. Damit können Beschreibungen dazu gespeichert werden, was einerseits eine bestimmte Variable bedeutet („Variable-Labels“) und was andererseits verschiedene Werte einer Variablen bedeuten („Value-Labels“). Gerade bei Daten, die öfter und von verschiedenen Personen genutzt werden sollen, sind das sehr wertvolle zusätzliche Beschreibungen.



	adsid	lppsqm	lrpsqm	ppsqmcl	rpsqmcl	price	mcu	plz	roomscount	livingarea	floorspace	year
2809	51002	3.322985	.2671757	2103.708	1.850017	124834	109.78	1210	2	59.34	.	
2810	6074	3.323135	.	2104.43	.	199500	.	1110	.	94.8	.	
2811	110391	3.32326	.	2105.039	.	99000	.	1210	2	47.03	.	
2812	93048	3.323306	.	2105.263	.	120000	.	1160	2	57	.	
2813	40385	3.323306	.0568547	2105.263	1.139868	160000	86.63	1020	3	76	.	
2814	28362	3.323306	.	2105.263	.	400000	.	1150	5	190	.	
2815	51489	3.323773	.	2107.527	.	196000	.	1100	4	93	.	
2816	21978	3.324282	.	2110	.	189900	.	1100	.	90	90	
2817	54082	3.324511	.	2111.111	.	190000	.	1160	3	90	.	
2818	43939	3.324511	.	2111.111	.	190000	.	1070	4	90	.	
2819	27681	3.324511	.	2111.111	.	133000	.	1210	2	63	.	
2820	16553	3.324536	.	2111.23	.	216190	.	1100	3	102.4	97.6	
2821	39077	3.324797	.	2112.5	.	169000	.	1140	2	80	80	
2822	15509	3.324833	.	2112.676	.	150000	.	1120	.	71	.	
2823	18144	3.325011	.	2113.54	.	214947	.	1100	4	101.7	97	
2824	13344	3.325164	.9330532	2114.286	8.571428	370000	1500	1120	4	175	175	
2825	49970	3.32526	.	2114.754	.	258000	.	1140	3.5	122	.	
2826	44785	3.32532	.2413878	2115.044	1.743363	239000	197	1140	3	113	.	
2827	89012	3.325711	.	2116.949	.	124900	.	1150	2	59	.	
2828	42808	3.325725	.	2117.021	.	199000	.	1170	4	94	94	
2829	89244	3.326058	.	2118.644	.	125000	.	1160	2	59	.	
2830	43357	3.326058	.351757	2118.644	2.247797	125000	132.62	1210	2	59	.	

Abbildung 2.1: Standard-Datensatz im **Stata** Data Editor

2.3.1 Daten einlesen

Damit wir in **Stata** mit unseren Daten arbeiten können, müssen sie zuerst im **Stata**-internen Format, dem DTA-Format vorliegen. Die entsprechenden DTA-Files sind auch jene, die wir in **Stata** über die Menüauswahl File-Open oder mit dem Befehl use öffnen können. Wenn wir in **Stata** Save oder Save as aus dem Menü auswählen, so speichert das Programm die Daten im Memory-Speicher im DTA-Format.

Um unsere in anderer Form vorliegenden Daten in das DTA-Format zu transferieren, gibt es mehrere Möglichkeiten. Wenn es die Daten schon im Format eines anderen Statistikprogramms gibt, so lohnt es sich zu prüfen, ob dieses Programm die Daten nicht im DTA-Format exportieren kann. **Stata** selbst ist in der aktuellen Version leider nicht gerade üppig mit Import-Möglichkeiten aus anderen Statistikprogrammen ausgestattet. Die einzige angebotene Möglichkeit ist das FDA-Format, ein Exportformat von SAS.

Etwas besser ausgestattet ist **Stata** mit Möglichkeiten, Daten direkt aus Datenbanken zu beziehen. Mit ODBC- und XML-Import stehen zwei recht flexible Datenbankzugänge zur Verfügung.

Das Standardformat für den Datenimport in **Stata** stellen aber ASCII-Daten in verschiedenen Formatierungen dar. Diese Formate haben den Vorteil, dass sie mit jedem Texteditor direkt eingesehen werden können und dass praktisch alle Programme in ein derartiges Format exportieren können. Liegen unsere Daten beispielsweise in Excel vor, so können wir sie aus Excel über „`Speichern unter`“ im Format „`Text (Tabstop getrennt)`“ exportieren, um sie dann über „`File - Import - ASCII Data generated by a spreadsheet`“ in **Stata** laden. Leider exportiert Excel – zumindest in der deutschsprachigen Version – die Daten standardmäßig in eine Datei mit der Endung „`txt`“ und erwartet **Stata** standardmäßig die Endung „`raw`“. Daher muss man bei diesem Weg immer den Dateinamen wechseln.

Enthält die erste Zeile dieser Datei nur Zeichenketten, so interpretiert **Stata** die erste Zeile als Variablennamen und verwendet sie dementsprechend. Treten beim Importieren irgendwelche Probleme auf, so gibt **Stata** Meldungen im Result-Fenster aus. Gehen Sie derartigen Meldungen auf den Grund, bevor Sie mit den Daten weiter arbeiten.

Eine andere Möglichkeit besteht noch darin, die Daten direkt in **Stata** einzugeben. Dazu müssen Sie **Stata** starten und bei einem leeren Datenspeicher den „Data Editor“ (über „`Data - Data Editor`“) aufrufen. Der Data Editor stellt eine Art Spreadsheet zur Verfügung, in das Sie die Daten eintragen können.

Wie praktisch alle anderen Standard-Statistikprogramme auch verlangt **Stata** die Daten in der Form einer rechteckigen Datentabelle. Dabei stellen die Spalten die Variablen dar und die Zeilen die Beobachtungen. Abbildung 2.1 zeigt einen Teil unseres Standard- Datensatzes im Data Editor von **Stata**. Wir sehen die ersten zehn Variablen mit den Namen „`adsid`“, „`lppsqm`“ usw., und die Beobachtungen Nr. 2809 – 2830.

Eine wichtige Funktion des Datenhandlings, die **Stata** und andere Statistikprogramme bereitstellen ist die Markierung von fehlenden Werten. Sie markieren jene Stellen der Datentabelle, für die keine Werte zur Verfügung stehen. Bei der Dateneingabe werden solche fehlenden Werte durch einen Punkt angegeben. Auch der Data Editor zeigt sie als Punkte an (siehe Abb. 2.1). Von **Stata** werden fehlende Werte speziell behandelt. Alle Datentransformationen, die fehlende Werte inkludieren, führen wiederum zu fehlenden Werten. Bei Schätzungen werden Beobachtungen, die bei den in der Schätzung verwendeten Variablen fehlende Werte enthalten, aus der Berechnung ausgeschlossen. Das kann dazu führen, dass bei der Hereinnahme oder beim Ausschluss von Variablen in einer Schätzung sich die Zahl der verwendeten Beobachtungen ändert, weil mehr oder weniger Beobachtungen wegen fehlender Werte aus der Berechnung ausgeschlossen werden.

2.4 Darstellungen von Daten

Wenn wir die Daten in das Statistikprogramm – in unserem Fall in **Stata** — eingelesen haben, sollten wir die Daten zuerst einmal auf verschiedene Arten darstellen. Damit verfolgen wir mehrere Zwecke:

1. zu sehen, ob die Daten auch korrekt in **Stata** angekommen sind;
2. zu prüfen, ob die Daten und alle Beobachtungen auch tatsächlich Sinn machen;
3. um ein „Gefühl“ für die Daten zu entwickeln.

Ein häufig gemachter Fehler besteht darin, die Überprüfung der Daten zu vernachlässigen oder gar zu überspringen. Oft herrscht ein gewisser Zeitdruck und es müssen erste Ergebnisse her. Da erscheint das Prüfen der Daten manchmal als Zeitverschwendung. Fehlerhafte oder fehlerhaft gelesene Daten entsprechen aber dem „zurück an den Start“ beim Spiel Mensch-ärgere-dich-nicht. Sie schlagen meistens dann zu, wenn man nach großem Aufwand endlich sinnvolle Ergebnisse zu haben glaubt. Nur ein Koeffizient zum Beispiel liefert einen unplausiblen Wert oder verhält sich eigenartig. Geht man dem nach, findet man dann oft einen Fehler in den Daten, nach dessen Korrektur man die gesamte Analyse von vorne beginnen muss. Das ist frustrierender als beim Mensch-ärgere-dich-nicht knapp vor dem Ziel hinausgeworfen zu werden. Mit einer ordentlichen Prüfung der Daten lassen sich solche Erlebnisse weitgehend verhindern.

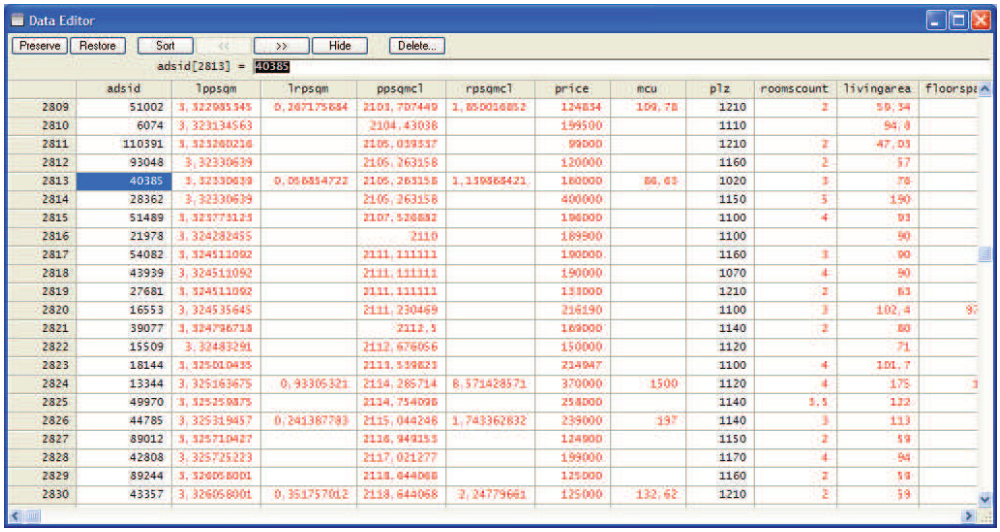
Ein wichtiger Ratschlag zur Datenüberprüfung ist auch der, wirklich *allen* Ungereimtheiten, die einem auffallen, auf den Grund zu gehen. Für die kleinen Ungereimtheiten gilt das gleiche, wie für die Datenüberprüfung insgesamt: Ignorieren rächt sich früher oder später.

Ein Nebeneffekt dieser Überprüfung ist auch der, dass man als Forscher die Daten besser kennen lernt. Dadurch kann man später gezielter Hypothesen aufstellen und man gelangt rascher zu einem guten Ergenis.

Hiermit sollte klar sein, dass ich sehr dafür bin, die Daten deskriptiv darzustellen und sie damit zu überprüfen. Aber wie sollten Sie das machen? Statistikprogramme im Allgemeinen und **Stata** im Besonderen bieten eine Reihe von Möglichkeiten dafür. Einige davon werden wir hier kurz besprechen:

Daten anzeigen

Rufen Sie in **Stata** den Data Editor oder den Data Browser auf und prüfen Sie stichprobenartig die Daten. Stimmen wirklich *alle* Werte mit den Ausgangsdaten überein? Treten irgendwo überraschend viele felhende Werte auf? Stellt **Stata** eine Variable, die eigentlich nummerisch sein



	adsid	lppsqm	lrpsqm	ppsqmcl	rpsqmcl	price	mcu	plz	roomscount	livingarea	floorspa
2809	51002	3,322985345	0,267175684	2101,707449	1,850016852	124834	109,78	1210	2	59,34	
2810	6074	3,323134563		2104,43038		199500		1110		94,8	
2811	110391	3,323260216		2105,039537		99000		1210	2	47,03	
2812	93048	3,32330639		2105,263158		120000		1160	2	57	
2813	40385	3,32330639	0,066884722	2105,263158	1,139868421	180000	86,63	1020	3	76	
2814	28362	3,32330639		2105,263158		400000		1150	5	190	
2815	51489	3,321771125		2107,526882		196000		1100	4	93	
2816	21978	3,324282455		2110		189900		1100		90	
2817	54082	3,324511092		2111,111111		190000		1160	3	90	
2818	43939	3,324511092		2111,111111		190000		1070	4	90	
2819	27681	3,324511092		2111,111111		133000		1210	2	63	
2820	16553	3,324535645		2111,230469		216190		1100	3	102,4	97
2821	39077	3,324796718		2112,5		169000		1140	2	80	
2822	15509	3,32483291		2112,676056		150000		1120		71	
2823	18144	3,325010435		2113,539823		214947		1100	4	101,7	
2824	13344	3,325163675	0,93305321	2114,285714	8,571428571	370000	1500	1120	4	175	3
2825	49970	3,325259875		2114,754098		258000		1140	3,5	122	
2826	44785	3,325319457	0,241387783	2115,044248	1,743362832	239000	197	1140	3	113	
2827	89012	3,325710427		2116,949153		124900		1150	2	59	
2828	42808	3,325725223		2117,021277		199000		1170	4	94	
2829	89244	3,326058001		2118,644068		125000		1160	2	59	
2830	43357	3,326058001	0,351757012	2118,644068	2,24779661	125000	132,62	1210	2	59	

Abbildung 2.2: Standard-Datensatz im **Stata** Data Editor – Strings statt Zahlen

sollte, als String-Variable dar? Stimmen die Anzahl der Variablen und die Anzahl der Beobachtungen mit Ihren Erwartungen exakt (nicht nur annähernd) überein? Tragen alle Variablen die erwarteten Namen?

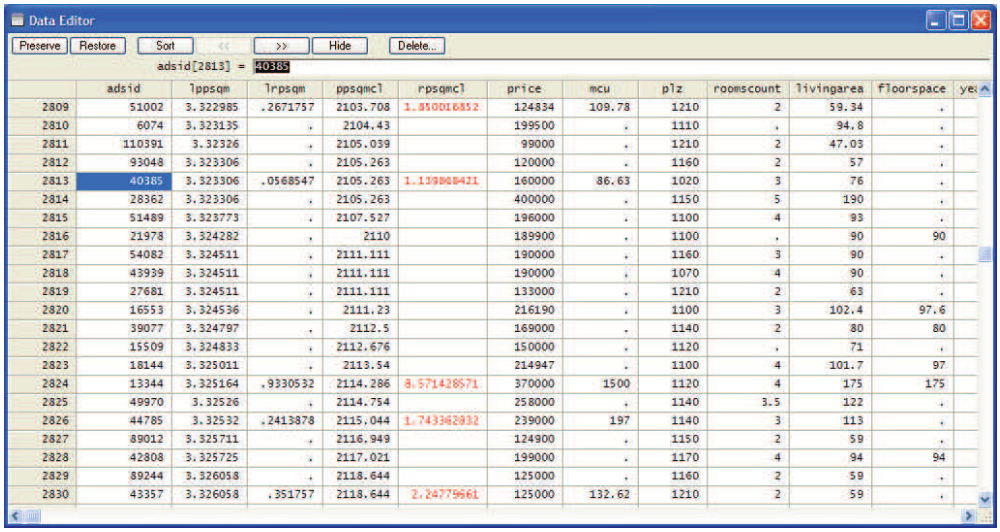
Das sind einige der Fragen, die Sie beim Betrachten der Daten beantworten sollten. An dieser Stelle sei auch noch einmal auf die Fehlermeldungen hingewiesen, die **Stata** im Fall von Problemen beim Datenimport im Result-Fenster ausgibt. Die Tatsache, dass es in ihrem **Stata**-Lauf Daten gibt, garantiert nicht, dass diese Daten auch brauchbar sind.

Führen wir die oben beschriebenen Schritte durch, um unsere Testdaten aus Excel zu exportieren und in **Stata** zu importieren, so sehen wir nicht das in Abb. 2.1 dargestellt Bild, sondern jenes in Abb. 2.2. Außer „`adsid`“ und „`plz`“ sind die Zahlen in allen Spalten rot[1] geschrieben, was in **Stata** anzeigt, dass diese Variable Zeichenketten und keine Zahlen enthält. Warum werden die Zahlen in allen diesen Spalten nicht als solche erkannt? Die Lösung des Rätsels liegt im Dezimalzeichen. Wir haben unseren Datensatz aus einer deutschsprachigen Excel-Version exportiert, die das Komma als Dezimalzeichen verwendet, und in eine englischsprachige Version von **Stata** importiert, die einen Punkt als Dezimalzeichen erwartet. Daher findet **Stata** in allen Variablen, die zumindest einmal Dezimalstellen enthalten, Zeichen, die in der englischsprachigen Version nicht zu einer nummerischen Darstellung gehören, und interpretiert diese Variablen daher als Zeichenketten.

Laden wir die ASCII-Datei in einen Texteditor und ersetzen wir alle Kommata durch Punkte – nicht ohne uns vorher vergewissert zu haben, dass die Datei sonst keine Punkte enthält –, so ist ein Großteil des Problems gelöst (siehe Abb. 2.3). Allerdings nicht das Ganze. Die Variable „`rpsqmcl`“ wird noch immer als Zeichenkette interpretiert. Der Grund für dieses Problem ist schwerer zu finden. Wenn wir in die Excel-Datei schauen, so erkennen wir dort, dass diese Variable bei allen – oder zumindest einigen – Beobachtungen, die leer erscheinen, Leerzeichen enthält. Diese, für uns am Bildschirm nicht sichtbaren Leerzeichen lassen **Stata** zum Schluss kommen, dass diese Variable eine Zeichenkette enthält. Diese Ursache ist besonders schwer zu erkennen, weil die störenden Leerzeichen in **Stata** gar nicht ankommen. Im Data Editor sind sie nicht zu sehen. Sie führen nur dazu, dass diese Variable als Zeichenkette klassifiziert wird, sodass wir sie in nummerischen Berechnungen nicht verwenden können.

Für die Lösung dieses Problems gibt es drei Möglichkeiten. Sie können die Ausgangsdaten in Excel so ändern, dass sie die störenden Leerzeichen entfernen, sie können die aus Excel exportierte

[1] In der Schwarzweißdarstellung dunkelgrau.



	adsid	lppsqm	lrpsqm	ppsqmcl	rpsqmcl	price	mcu	plz	roomscount	livingarea	floorspace	ye
2809	51002	3.322985	.2671757	2103.708	1.850016852	124834	109.78	1210	2	59.34	.	
2810	6074	3.323135	.	2104.43		199500	.	1110	.	94.8	.	
2811	110391	3.32326	.	2105.039		99000	.	1210	2	47.03	.	
2812	93048	3.323306	.	2105.263		120000	.	1160	2	57	.	
2813	40385	3.323306	.0568547	2105.263	1.139868421	160000	86.63	1020	3	76	.	
2814	28362	3.323306	.	2105.263		400000	.	1150	5	190	.	
2815	51489	3.323773	.	2107.527		196000	.	1100	4	93	.	
2816	21978	3.324282	.	2110		189900	.	1100	.	90	90	
2817	54082	3.324511	.	2111.111		190000	.	1160	3	90	.	
2818	43939	3.324511	.	2111.111		190000	.	1070	4	90	.	
2819	27681	3.324511	.	2111.111		133000	.	1210	2	63	.	
2820	16553	3.324536	.	2111.23		216190	.	1100	3	102.4	97.6	
2821	39077	3.324797	.	2112.5		169000	.	1140	2	80	80	
2822	15509	3.324833	.	2112.676		150000	.	1120	.	71	.	
2823	18144	3.325011	.	2113.54		214947	.	1100	4	101.7	97	
2824	13344	3.325164	.9330532	2114.286	8.571428571	370000	1500	1120	4	175	175	
2825	49970	3.32526	.	2114.754		258000	.	1140	3.5	122	.	
2826	44785	3.32532	.2413878	2115.044	1.743362832	239000	197	1140	3	113	.	
2827	89012	3.325711	.	2116.949		124900	.	1150	2	59	.	
2828	42808	3.325725	.	2117.021		199000	.	1170	4	94	94	
2829	89244	3.326058	.	2118.644		125000	.	1160	2	59	.	
2830	43357	3.326058	.351757	2118.644	2.24779661	125000	132.62	1210	2	59	.	

Abbildung 2.3: Standard-Datensatz im **Stata** Data Editor – versteckte Leerzeichen

Ascii-Datei in einen Texteditor laden und die störenden Leerzeichen mit Suchen und Ersetzen entfernen oder Sie wandeln in **Stata** die Variable mithilfe der Funktion „`real()`“ in eine nummerische Variable um. Welchen Weg Sie auch wählen, das Ergebnis entspricht dem in Abb. 2.1 dargestellten Datensatz.

Maxima und Minima errechnen

Über die **Stata**-Funktion „`summarize`“ lassen sich – gemeinsam mit anderen, noch zu besprechenden Indikatoren – die Maxima und Minima der Variablen errechnen.

Über die beiden Spalten „`max`“ und „`min`“ können wir leicht sehen, ob die Werte der Variablen im sinnvollen Wertebereich liegen. Negative Werte bei Preisvariablen würden hier zum Beispiel sofort auffallen. In unseren Testdaten sehen wir, dass der höchste Preis bei 22,9 Mio. Euro liegt. Dieser Wert ist zwar nicht unmöglich, aber schon ungewöhnlich hoch. Wir sollten die entsprechende Beobachtung in unserem Datensatz daher genauer anschauen.

Die Spalte „`Obs`“ zeigt übrigens die Anzahl der „validen“ Beobachtungen, also die Zahl der Beobachtungen, die bei dieser Variablen keinen fehlenden Wert aufweisen. In unserem Datensatz können wir da einige Variable erkennen, die relativ viele fehlende Werte aufweisen. Diese sind für die Modellschätzung mit Vorsicht zu verwenden, weil sie die Zahl der Beobachtungen in der Schätzung stark reduzieren.

Häufigkeiten

Für kategoriale und für diskrete Variable empfiehlt es sich, Häufigkeiten dastellen zu lassen. Dies geschieht mit dem **Stata**-Befehl „`table`“. Dabei zählt **Stata**, wie oft die verschiedenen Werte der darzustellenden Variablen im Datensatz vorkommen, und stellt die Ergebnisse tabellarisch dar.

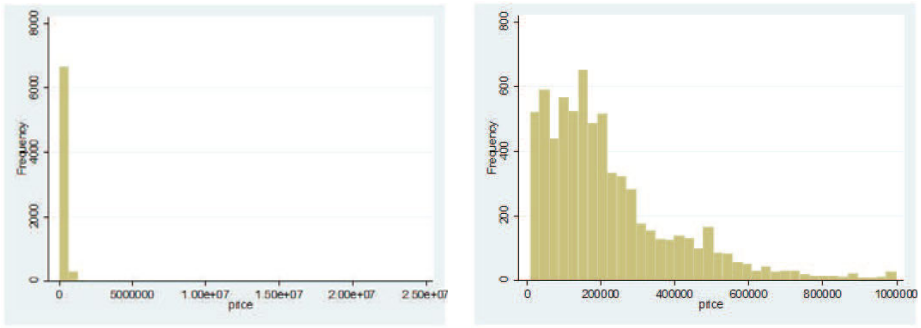
Die erste Spalte der Tabelle zeigt die verschiedenen Werte der Variablen (im konkreten Fall „`bathroomcount`“, also die Anzahl der Badezimmer). Die Spalte „`Freq.`“ zeigt dann, wie viele Beobachtungen diesen Wert aufweisen. Das nennt man *absolute Häufigkeit*. Die letzte Zeile, „`Total`“, zeigt dann in dieser Spalte die Summe aller Häufigkeiten, also die gesamte Anzahl der validen Werte. Die nächste Spalte, „`Percent`“, zeigt die *relative Häufigkeit*, also den Anteil der einzelnen Werte an der Gesamtzahl der validen Werte. In Summe macht das, wie in der letzten Zeile zu

```
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       adsid |      7078    66616.09    30525.27       4657     126028
      lppsqm |      7078    3.329797    .2934978   2.255952    5.26993
      lrpsqm |       621    .3826477    .1595573  -.0178517    1.223277
     ppsqmcl |      7078    2653.992    4079.854   180.2817    186178.9
     rpsqmcl |       621    2.631228    1.566708   .9597284    16.72158
-------------+--------------------------------------------------------
       price |      7078    263395.2    562375.9       8820    2.29e+07
         mcu |       621    217.8381    186.3499      76.91    1910.15
         plz |      7078     1114.46    68.00856       1010        1230
  roomscount |      6626     2.77377    1.212848          1          12
  livingarea |      7078    93.92644    115.1984         18        2469
-------------+--------------------------------------------------------
  floorspace |       881    136.3139    195.5168         28        2600
  yearconstd |      1338    1976.294    45.00127       1800        2011
      floorn |      4172    2.551774    1.670238          0           5
       condn |      4343    1.173152    .4788934          1           3
 toiletcount |      4102    1.215261    .5509684          0           8
-------------+--------------------------------------------------------
bathroomcount |      5552    .7060519    .6727413          0           6
```

Output 2.1: Ausgabe des **Stata**-Befehls „`summarize`“

```
bathroomcount |      Freq.     Percent        Cum.
--------------+-----------------------------------
            0 |      2,250       40.53       40.53
            1 |      2,732       49.21       89.73
            2 |        533        9.60       99.33
            3 |         28        0.50       99.84
            4 |          8        0.14       99.98
            6 |          1        0.02      100.00
--------------+-----------------------------------
        Total |      5,552      100.00
```

Output 2.2: Ausgabe des **Stata**-Befehls „`table`“



(a) gesamter Wertebereich (b) eingeschränkter Wertebereich

Abbildung 2.4: Histogramm in **Stata** – Variable „`price`“

sehen ist, 100 Prozent aus. Die letzte Spalte, „`Cum.`“, gibt die *kumulative relative Häufigkeit* wider. In dieser Spalte sieht man also, wie hoch der Anteil an Beobachtungen ist, die diesen Wert oder einen kleineren aufweisen. Wir sehen daran also, zum Beispiel, dass 99,33% der Beobachtungen über zwei oder weniger Badezimmer verfügen. Diese Spalte der Ausgabe macht nur dann Sinn, wenn die dargestellten Kategorien eine Reihung beinhalten. Bei mit Zeichenketten kodierten kategorialen Variablen ist diese Spalte daher oft nicht interpretierbar.

Verwendet man den Befehl „`summarize`“ für eine stetige Variable, so ist das Ergebnis zwar sehr umfangreich, aber wenig aussagekräftig. Da bei stetigen Variablen die Chancen gering sind, dass zwei Beobachtungen den gleichen Wert aufweisen, listet die Darstellung viele verschiedene Werte auf, wobei die allermeisten eine Häufigkeit von 1 aufweisen.

Histogramme

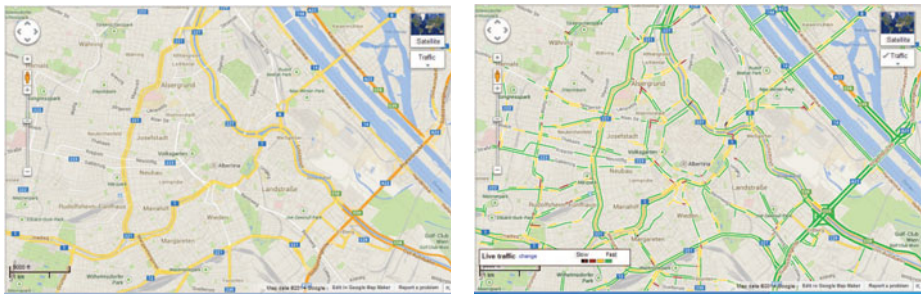
Da ein Bild oft mehr sagt als tausend Worte, ist es oft auch sinnvoll, die Variablen und ihre Werteverteilung auch graphisch darzustellen. Dazu gibt es in **Stata** verschiedene Arten der Darstellung. Die wichtigste ist wahrscheinlich das Histogramm. Es wird in **Stata** über den Befehl „`histogram`“ oder den Menüpunkt „`Graphics - Histogram`“ aufgerufen. Das Ergebnis ist eine Darstellung der Verteilung der Variablen mit Balken, deren Höhe von der Häufigkeit der entsprechenden Werte abhängt. Die Darstellung kann über eine große Menge an Parametern an die spezifischen Bedürfnisse angepasst werden.

Lassen wir **Stata** ein Histogramm der Variablen „`price`“ erstellen (Abb. 2.4(a)), so drückt sich im Ergebnis auch die Tatsache aus, dass der Maximalwert für diese Variable sehr hoch ist. Weil auf der horizontalen Achse der Darstellung die gesamte Spannweite der Werte zwischen Minimum und Maximum dargestellt und dabei in gleichmäßige Intervalle unterteilt wird, fallen fast alle beobachteten Werte in den ersten Balken. Wir sehen daher nur sehr wenig über die tatsächliche Verteilung der Preise.

Beschränken wir allerdings die Darstellung auf Preise bis zu einer Million Euro, so ziehen sich die Balken auseinander und die Darstellung wird wesentlich übersichtlicher (Abb. 2.4(b)). Wir sehen nun, dass die Mehrzahl der Beobachtungen bei den niedrigeren Preisen konzentriert ist, dann aber relativ kontinuierlich abfällt. Die Einschränkung auf Werte unter 1 Mio. Euro geschieht mithilfe einer „`if`“-Klausel im Aufruf des Histogramms. Um die Graphik in Abb. 2.4(b) zu generieren, verwendeten wir den Aufruf „`histogram price if price < 1000000, frequency`“.

Karten

Eine sehr wichtige Art der Darstellung räumlicher Daten sind Karten. Weil diese Option der Darstellung üblicherweise in Statistiksoftware nicht inkludiert ist, benötigen wir normalerweise



(a) nur geographische Information

(b) geographische und sachliche Information kombiniert

Abbildung 2.5: Geographische und sachliche Information in Google Maps

zusätzliche Software. Dieser Bereich, der oft mit dem Begriff „Geographisches Informationssystem“ oder kurz „GIS“ bezeichnet wird, ist zu umfangreich, um hier auch nur annähernd dargestellt zu werden. Daher wollen wir uns auf eine nur sehr knappe Darstellung beschränken. Für einen ausführlicheren Überblick siehe etwa Longley (2005), Longley et al. (2010), Sengupta (2007), Sengupta und Nag (2007), Bernhardsen (2002), Chun und Griffith (2013), Schuurman (2004), Steinberg und Steinberg (2006).

Ein Geographisches Informationssystem kombiniert zwei Arten von Informationen, nämlich geographische und sachliche. Beide Arten von Informationen stehen in einem GIS in elektronischer Form zur Verfügung. Geographische Informationen beschreiben die räumlichen Gegebenheiten also so Dinge wie den Verlauf von Grenzen, die Lage von Siedlungen oder bestimmten Gebäuden, die Verläufe von Straßen, Bahnlinien, Flüssen, Leitungen und ähnlichem. Im Gegensatz dazu beschreiben sachliche Informationen die Eigenschaften derartiger räumlicher Gegebenheiten. Beispiele dafür sind Bevölkerungszahl oder Bevölkerungsdichte, die Kapazität von verschiedenen Abschnitten der Infrastruktur, die Wasserqualität von Flussabschnitten, das Alter von Gebäuden oder deren Nutzungsart, etc.

Den Unterschied zwischen geographischen und sachlichen Informationen können wir anhand von Google Maps[2] darstellen. Abbildung 2.5 zeigt auf der linken Seite die Grundkarte von Google Maps für einen Teil von Wien, also nur die geographische Information. Im rechten Teil der Abbildung wird diese Grundkarte mit der sachlichen Information über die aktuelle Verkehrssituation kombiniert. Aus diesem Teil können wir also nicht nur die Straßenverläufe ablesen, sondern auch, wie schnell aufgrund der Verkehrssituation dort gerade gefahren werden kann. Für diese Darstellung wird die geographische Information über die Straßenverläufe mit der sachlichen Information über die Verkehrslage kombiniert.

Der Grund dafür, warum mit Statistikprogrammen typischerweise keine[3] Karten erstellt werden können, liegt in der Notwendigkeit, die sachliche Information mit geographischer Information zu verknüpfen. Weil statistische Auswertungen normalerweise geographische Gegebenheiten nicht berücksichtigen, können sie üblicherweise auch nicht mit derartigen Informationen umgehen. Zum Zeichnen elektronischer Karten sind daher spezielle Datenformate und spezielle Programme notwendig. Ein wichtiges Dateiformat für Geographische Informationssysteme ist das vom Unternehmen ESRI entwickelte Shapefile Format. Ein „Shapefile“ besteht eigentlich aus mehreren Dateien mit gleichem Namen und verschiedene Dateierweiterungen, wobei drei – `.shp`, `.shx` und `.dbf` – unbedingt notwendig und einige andere optional sind. Üblicherweise können Shapefiles von den verschiedenen GIS-Programmen gelesen werden und für viele räumliche Einheiten sind auch Shapefiles am Internet zu finden. Wichtige kommerzielle GIS Programme sind beispielsweise „ArcGIS“ und „MapInfo“. Open Source Programme sind „GRASS GIS“, „Quantum GIS“ und „GeoDa“, wobei letzteres auf explorative Analyse räumlicher Daten spezialisiert ist.

[2] `http://maps.google.com`

[3] Eine Ausnahme ist das Open Source Programm R.

http://www.springer.com/978-3-658-02861-9

Immobilienbewertung mit hedonischen Preismodellen
Theoretische Grundlagen und praktische Anwendung
Maier, G.; Herath, S.
2015, IX, 199 S. 30 Abb., Hardcover
ISBN: 978-3-658-02861-9