

# Räumliche Vorhersage des urbanen Radverkehrs mit Machine Learning

Maximilian Samuel Weinhold  
Economics, 7. Semester  
505314  
[mweinhol@uni-muenster.de](mailto:mweinhol@uni-muenster.de)

Masterarbeit  
Wintersemester 2022  
Institut für Verkehrswissenschaft  
Prof. Dr. Gernot Sieg

## INHALTSVERZEICHNIS

0.1 Abkürzungsverzeichnis . . . . .	6
1. <i>Einleitung</i> . . . . .	8
2. <i>Literaturüberblick</i> . . . . .	9
2.1 Erforschung des Radverkehrs . . . . .	9
2.1.1 Forschung mit Zählstationen . . . . .	10
2.1.2 Forschung mit Bike-Sharing-Diensten . . . . .	11
2.1.3 Forschung mit GPS- und Handy Daten . . . . .	14
2.2 Faktoren des Radverkehrs . . . . .	17
2.2.1 Faktor Wetter . . . . .	18
2.2.2 Faktor Feinstaubbelastung . . . . .	19
2.2.3 Faktor Corona-Maßnahmen . . . . .	20
2.2.4 Faktor der städtischen Unterschiede . . . . .	21
2.2.5 Faktor Radverkehrsunfälle . . . . .	21
2.2.6 Sonstige Faktoren . . . . .	25
2.3 Forschungslücken und Anknüpfungspunkte . . . . .	25
3. <i>Zusammensetzung des Datensatz</i> . . . . .	28
3.1 Fahrradzähler . . . . .	29
3.2 Wetterdaten . . . . .	31
3.3 Demographische und soziale Statistiken . . . . .	33
3.3.1 Statistisches Bundesamt . . . . .	33
3.3.2 ADFC Fahrradindex . . . . .	36
3.4 Corona-Daten . . . . .	37
3.5 Open-Street-Map-Daten . . . . .	40
3.5.1 Ausgewählte POIs . . . . .	41

3.5.2	Ausgestaltung des öffentlichen Verkehrs . . . . .	44
3.5.3	Straßentypen . . . . .	45
3.6	Sonstige Variablen . . . . .	50
4.	<i>Verwendete Methoden</i> . . . . .	52
4.1	Probleme zur Autokorrelation . . . . .	52
4.1.1	Lagged Variablen . . . . .	53
4.1.2	Erklärende Variablen . . . . .	54
4.1.3	Resampling . . . . .	55
4.2	OLS-Regression . . . . .	55
4.3	Support-Vector-Regression . . . . .	56
4.4	Random-Forests-Regression . . . . .	59
4.5	Neuronale Netze . . . . .	62
4.5.1	Aufbau eines neuronalen Netzes . . . . .	62
4.5.2	Berechnung eines neuronalen Netzes . . . . .	64
4.6	Validation . . . . .	66
4.6.1	Cross Validation . . . . .	67
4.6.2	Conditional-Validation-Set-Building . . . . .	67
4.6.3	Weighted-Subset-Building . . . . .	70
5.	<i>Ergebnisse</i> . . . . .	71
5.1	OLS-Regression . . . . .	71
5.2	Support-Vector-Regression . . . . .	74
5.3	Random-Forest-Regression . . . . .	77
5.4	Neuronales Netz . . . . .	81
5.5	Modellprojektion . . . . .	83
5.6	Modellprognosen . . . . .	92
5.7	Räumliche Korrelation zu Verkehrsunfällen . . . . .	93
6.	<i>Diskussion</i> . . . . .	98
6.1	Fazit . . . . .	103
7.	<i>Anhang</i> . . . . .	105

## ABBILDUNGSVERZEICHNIS

2.1	Anzahl der Leihräder	13
2.2	Fahrradaufkommen für Glasgow 2017	16
2.3	Berechnete Wahrscheinlichkeiten für Radunfälle in Philadelphia	23
2.4	Räumliche Verteilung aller Fahrradunfälle (2011 bis 2014) nach Zensus Blöcken in Florida.	24
3.1	Städte die im Datensatz vertreten sind.	30
3.2	Verteilung der Beobachtung nach Einwohnergröße der Städte	31
3.3	Verteilung der Beobachtung nach Jahren	32
3.4	Wetterdaten im Überblick	33
3.5	Verhältnis von Bevölkerung und Fahrradaufkommen	34
3.6	Verteilung des Fahrradaufkommens in Entfernung zum Stadt- zentrum	35
3.7	Verhältnis von Bevölkerung und Fahrradaufkommen	36
3.8	Verlauf der Corona Inzidenz und des Radverkehrs	38
3.9	Radverkehr im Lockdown	39
3.10	Verteilung von Straßentypen im Datensatz nach der Version	47
3.11	Beispiel der Straßentypanalyse	48
3.12	Radverkehr nach Straßenbeschaffenheit	51
4.1	Hyperebenen (Hyperplanes) in Support-Vector-Machines	57
4.2	Soft Margins in Support-Vector-Machines	57
4.3	Eindimensionale lineare Support-Vector-Regression	58
4.4	Ein Entscheidungsbaum basierend auf Daten von Fahrradzähl- stationen in Mannheim und Daten des DWD 2016 bis 2022.	60
4.5	Beispiel eines Random Forests	61
4.6	Beispiel eines neuronalen Netzes	63

4.7	Aktivierungsfunktionen . . . . .	64
4.8	Gradient Descent . . . . .	65
4.9	Beispiel einer Cross Validation . . . . .	68
5.1	Feature Selection Ergebnis der OLS-Regression in 6 Modellen	73
5.2	Support-Vector-Regression-Performance nach Anteil des Datensatzes . . . . .	75
5.3	Support-Vector-Regression Ergebnis in 6 Modellen . . . . .	76
5.4	Random-Forest-Performance nach Anteil des Datensatzes . . . . .	78
5.5	Random-Forest-Performance nach Anzahl der Zufallsbäume . . . . .	79
5.6	Feature Selection Ergebnis der RF Regression in 6 Modellen . . . . .	80
5.7	Räumliche Modellprojektion: Modellvergleich . . . . .	87
5.8	Feature-Selection-Ergebnis der RF Regression in 6 Modellen . . . . .	89
5.9	Modellprojektionen in Hamburg und Leipzig . . . . .	90
5.10	Modellprojektionen in Mannheim und Oberhausen . . . . .	91
5.11	Modellprojektionen in Dresden . . . . .	93
5.12	Stadtentwicklungsprognose . . . . .	94
5.13	Radverkehrsentwicklungsprognose für Hamburg . . . . .	94
5.14	Räumliche Korrelation von Unfällen und Radverkehr . . . . .	95
5.15	Karten zu Unfällen und Radverkehr . . . . .	97
6.1	Modellvorhersagen nach Wetter . . . . .	101
7.1	Verteilung des Fahrradaufkommens nach Alter . . . . .	105
7.2	Zusammenhang von Fahrradklima und Radverkehr . . . . .	106
7.3	Zusammenhang von Anzahl der Uni-Gebäude in einem 500 M Radius und Radverkehr . . . . .	107
7.4	Zusammenhang von Anzahl der Supermärkte in einem 1 km Radius und Radverkehr . . . . .	108
7.5	Zusammenhang von Anzahl der Kleidungsgeschäften in einem 2 km Radius und Radverkehr . . . . .	109
7.6	Zusammenhang von Anzahl der Busstationen in einem 1 km Radius und Radverkehr . . . . .	110

7.7	Zusammenhang von Anzahl der Ampeln in einem 1 km Radius und Radverkehr . . . . .	111
7.8	Zusammenhang von Anzahl der Straßenbahnstationen in einem 1 km Radius und Radverkehr . . . . .	112
7.9	Zusammenhang des nächsten Bahnhofes und dem Radverkehr	113
7.10	Entfernung zur nächsten Brücke und Radverkehr . . . . .	114
7.11	Kartendarstellung mit unterschiedlichen Details . . . . .	115
7.12	Kartendarstellung mit unterschiedlichem Wetter . . . . .	116

## TABELLENVERZEICHNIS

5.1	Performance des OLS-Modells . . . . .	74
5.2	Performance des SVR-Modells . . . . .	78
5.3	Performance des RF Modells . . . . .	81
5.4	Performance des neuronalen Netzes . . . . .	82
5.5	Performance des dritten RF Modells . . . . .	88

### 0.1 Abkürzungsverzeichnis

**ADFC** Allgemeiner-Deutsche-Fahrrad-Club

**bzw.** beziehungsweise

**ca.** circa

**CHAID** Chi-Squared-Automatic-Interaction-Detection

**Destatis** Bundesamt für Statistik

**Mio.** Million

**OLS** Ordinary Least Square

**PM<sub>2,5</sub>** Feinstaubpartikel kleiner als 2,5 Mikrometer

**POI** Point of Interest

**ReLU** Rectified-Linear-Unit

**RKI** Robert-Koch-Institut

**RMSE** Root-Mean-Squared-Error

**RSS** Residuenquadratsumme

**SVR** Support-Vector-Regression

**Tsd.** Tausend

**u.a.** unter anderem

**z.B.** zum Beispiel

**ZIV** Zweirad-Industrie-Verband

## 1. EINLEITUNG

Vielerorts erlebt der Individualverkehr eine Renaissance des Fahrradfahrens. So ist laut Eisenberger (2015) und dem Verband der Zweirad Industrie ZIV (2022) der Bestand an Fahrrädern in Deutschland von 72 Mio. in 2015 auf 81 Mio. in 2021 angestiegen. Der anhaltende Boom hat viele Gründe. Im Vergleich zum Auto ist die bewegungsintensivere Fortbewegung auf dem Fahrrad gesünder, schont die Umwelt und das Portemonnaie.

Viele Kommunen entscheiden sich u.a. aus diesen Gründen dafür, ihre Stadt fahrradfreundlicher zu gestalten. So unterstützt das Bundesministerium für Verkehr und digitale Infrastruktur BMDV (2020) die Länder und Gemeinden beim Ausbau von Radwegen durch eine direkte Hilfe in Höhe von 660 Mio. Euro bis 2023.

Dies, sowie der genaue lokale Bedarf und die Auslastung von Fahrradwegen, muss bei der Planung der Infrastruktur beachtet werden. Motiviert ist diese Arbeit durch den Wunsch, Antworten auf Fragen der Infrastrukturplanung bieten zu können. Die Forschungsfrage, die sich daraus ableitet, lautet: Ist es möglich, ein räumliches Modell zu entwickeln, das für ein vollständiges Straßennetz oder für ausgewählte flexible Knotenpunkte zu bestimmten Zeiten Vorhersagen zum Fahrradverkehr treffen kann und lassen diese auch zusätzlich die gefährlichsten Stellen für Fahrradunfälle erkennen? Ziel dabei ist es, eine Anleitung zur Erstellung solcher Modelle zu geben.

Dazu beginnt diese Masterarbeit mit einem Einblick in die bestehende Literatur, welcher kategorisiert ist nach Daten und Faktoren. Häufig verwendete Methoden werden im Nachgang näher beschrieben. Das erworbene Wissen dient dazu, ein Modell zu entwickeln, das im Rahmen dieser Abschlussarbeit auch evaluiert wird. Im Anschluss wird analysiert, wie sehr das Radverkehrsaufkommen mit Radverkehrsunfällen räumlich korreliert.

## 2. LITERATURÜBERBLICK

Die Auslastung von Fahrradwegen, bzw. das Aufkommen von Fahrrädern, beruht im Wesentlichen auf der individuellen Entscheidung eines jeden Fahrers, das Fahrrad einer anderen Transportalternative vorzuziehen. Versteht man die Faktoren, aus denen sich diese Entscheidung zusammensetzt, dann kann man leichter eine solche Entscheidung vorhersagen. Mit einer Zusammenfassung der bisherigen Literatur wollen Heinen et al. (2010) diese Faktoren aufzeigen, wobei sie sich hier nur auf Pendler beschränken. Zum einen ist einer dieser Faktoren die bauliche Substanz, die nicht nur die Radwege sondern auch Abstellmöglichkeiten und Ampeln wie auch Verkehrsschilder beinhaltet, zum anderem spielt auch das Wetter eine große Rolle. Einen negativen Effekt hat die Rate der Autobesitzer und auch die Verfügbarkeit anderer Transportmöglichkeiten.

### 2.1 *Erforschung des Radverkehrs*

Seitdem erschienen zahlreiche weitere empirische Studien, die das Bild über den Radverkehr konkretisieren. Diese verschiedenen Studien lassen sich nach Datenquellen kategorisieren. Die meisten nutzen drei verschiedene Datengrundlagen zur Modellierung des Fahrradverkehr in Verbindung zu Daten mit anderen Faktoren. Die drei Datenquellen zum Fahrradverkehr stammen von Fahrradzählstation, einer Induktionsschleife die darüber fahrende Räder verifiziert, Daten von Bike-Sharing-Diensten und Daten verschiedener GPS- und Handy-Applikationen.

### 2.1.1 Forschung mit Zählstationen

In Deutschland findet sich eine weite Verbreitung von Fahrradzählstationen in verschiedenen Städten, deren Daten oft öffentlich einsehbar sind. Deswegen wäre eine Verwendung dieser Datengrundlage überaus naheliegend. Studien, die ähnliche Daten verwenden, stammen z.B. von Holmgren et al. (2017), Broucke et al. (2019), Wessel (2020) und Goldmann und Wessel (2021). Die Arbeit von Wessel (2020) und die von Goldmann und Wessel (2021) werden in einem späteren Abschnitt behandelt, in dem es mehr um den kausalen Zusammenhang von Fahrradverkehr und Wetter geht, da sich beide Studien verstärkt diesem Thema annehmen. Holmgren et al. (2017) verwenden tägliche Daten von 2006 bis 2014 aus Malmö und verbinden diese ebenfalls wie später bei Wessel (2020) mit Wetterdaten, Feiertagen und Schulferien. Als Methode zur Auswertung dieser Daten und zur Schätzung des Fahrradaufkommens verwenden sie Random-Forest-Regressionssysteme, Support-Vector-Regression, eine lineare Regression und einen Multy-Layer-Perceptron, also ein neuronales Netz. Im Vergleich dieser Methoden erzielten sie die treffsichersten Resultate mit einem Regressionsbaum und der Support-Vector-Regression, welche auf quadratische und kubische Kernels zurückgreift. Mithilfe des Support-Vector-Regressionssystems kommen sie auf ein Bestimmtheitsmaß  $R^2$  von 86,9 %.

Um diese Methoden zu vergleichen nutzen Holmgren et al. (2017) die Cross-Validation. Ähnlich gehen Broucke et al. (2019) vor und verwenden dabei Daten von 12 Zählstationen aus Brüssel. Sie verbinden diese Daten mit temporalen, geographischen und meteorologischen Daten. Ebenfalls verwenden sie eine Support Vector Regression, die auf einer Radial-Basis-Function aufbaut, um eines ihrer vier Modelle zu berechnen. Für die anderen drei verwenden sie eine Random-Forest-Regression, welche auf Zufallsbäume aufbaut, ein Gradient-Boosting-Modell und ein voll verknüpftes neuronales Netzwerk. Letzteres beinhaltet 2 Schichten mit einmal 28 und einmal 14 Knoten, die die ReLU (Rectified-Linear-Unit) Funktion als Aktivierung verwenden. Von allen Modellen mit Wetterdaten erzielt das neuronale Netzwerk hier den niedrigsten RMSE (Root Mean Squared Error) und schneidet am besten ab.

### 2.1.2 Forschung mit Bike-Sharing-Diensten

Die überragende Mehrheit an Studien zur Schätzung und Vorhersage des Fahrradverkehrs verwendet Daten von Bike-Sharing-Diensten, dazu zählen Kaltenbrunner et al. (2010), Xu et al. (2013), Li et al. (2015), Mitchell (2018), Colace et al. (2020), Gao und Chen (2022) und Li et al. (2022). Dies ist natürlich immer noch nur eine Auswahl an Studien. Die Literatur zu Bike-Sharing-Systemen ist sehr ausführlich. Eine weitere Übersicht hierzu findet sich bei Mitchell (2018).

Kaltenbrunner et al. (2010) nutzen z.B. Daten von 400 öffentlichen Radausleihstationen in Barcelona. Ihre Studie hegt die Absicht, die Effizienz des bestehenden Verleihsystems in Barcelona, das zu dem Zeitpunkt um die 180.000 Abonnenten hatte, zu verbessern. Hier geht es also eher darum, vorherzusagen, wie viele Fahrräder sich in welcher Ausleihstation zu welchem Zeitpunkt befinden, um den Nutzern detaillierte Informationen zu geben.

Eine ähnliche Motivation haben Xu et al. (2013). Diese verwenden Daten des größten öffentlichen Bike-Sharing-Systems der Welt in Hangzhou in China. Ihr Datensatz beinhaltet die aufgezeichnete Auslastung der Stationen vorangegangener Tage und Wochen, sowie Daten zum aktuellen und vergangenem Wetter und Informationen über Feiertage. Diesen Datensatz normalisieren und clustern sie mit der k-Means-Cluster-Methode. Darauf wenden sie Support-Vector-Machines an, um die Gewichte der beschriebenen Estimatoren zu finden. Dieses hybride Modell hat nur noch eine Fehlerrate von 3,57 %, vergleicht man dessen Vorhersagen mit den tatsächlich eingetretenen Auslastungen.

Li et al. (2015) stellen hier ein Paper zur Verfügung das ähnlich funktioniert. Auch sie wollen eine Ausbalancierung der Fahrradbestände an allen Stationen in New York und Washington DC erleichtern. Und genau wie Xu et al. (2013) unterteilen sie die Stationen in Cluster. Für diese Cluster wollen sie Vorhersagen machen, was zu robusteren Ergebnissen führt, als wenn man für jede einzelne Station Vorhersagen bildet. Zusätzlich verwenden Li et al. (2015) Wetterbeobachtungen in ihrem Datensatz. Darauf wird ein Gradient-Boosting-Regression-Tree angewendet. Gradient-Boosting besteht aus kurz-

läufigen zufälligen Entscheidungsbäumen, die der Reihe nach auf dem Fehlerterm des vorherigen Entscheidungsbaumes beruhen und diesen versuchen zu verbessern.

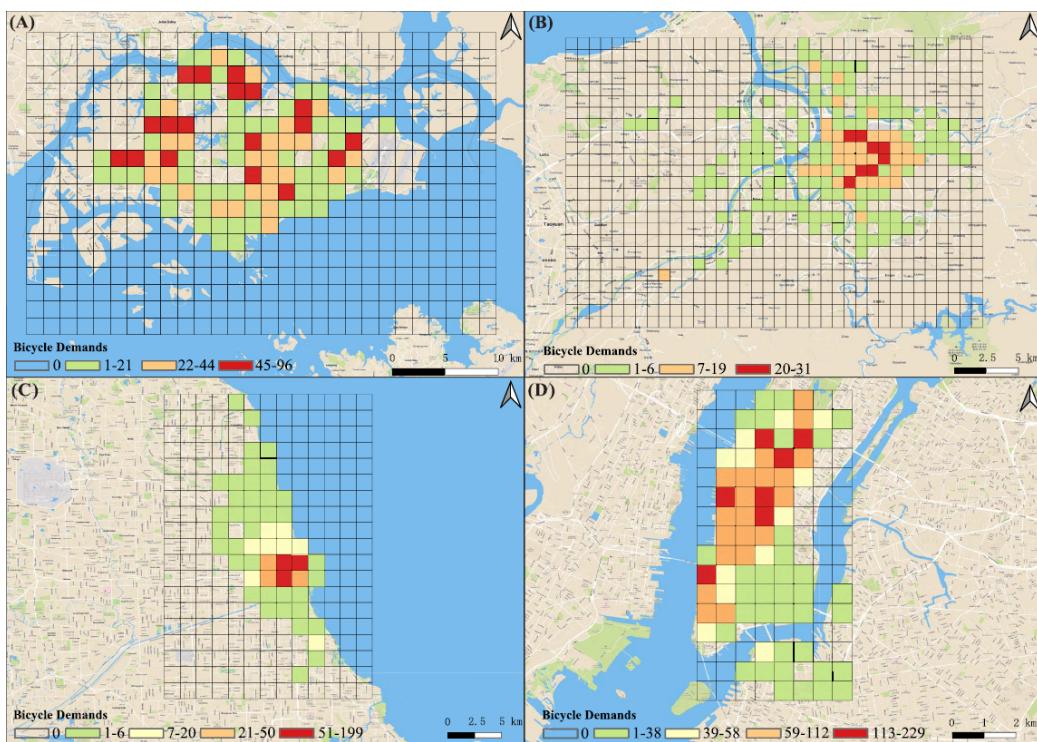
Auf Cluster-Level-Bike-Sharing geht auch Mitchell (2018) ein. Er evaluiert verschiedene Machine-Learning-Methoden miteinander: Random Forests, Fast-Feed-Forward-Neural-Networks, Deep-Residual-Networks und Recurrent-Neural-Networks. Dabei schlagen sich die Feed-Forward-Neural-Networks am besten. Einen besonderen Ansatz verfolgen Colace et al. (2020), die Aufnahmen von Überwachungskameras in ihrer Analyse mit aufnehmen.

Eine besonders aktuelle Analyse stammte von Gao und Chen (2022). Aufbauend auf Daten zu 2098 Fahrradstationen aus Seoul nehmen sie in ihr Modell z.B. Feinstaubbelastungen mit auf, was Hong J (2022) und Zhao et al. (2018) als relevanten Faktor gezeigt haben. Daneben inkludieren sie auch Verkehrsdaten, Corona Fälle und sozioökonomische Daten, Verkehrsunfälle und Saisonalität. Darauf wenden sie lineare Regressionen, k-Nearest-Neighbour (knn), Random Forests und Support-Vector-Machines an. All diese Methoden wurden in der Programmiersprache R angewendet. Diese Modelle vergleichen sie mit einem Validation-Set-Schnitt von 75 % für das Trainingsset und 25 % für das Testset. Damit betreiben Gao und Chen (2022) eine Feature Selection, die auf den Boruta Algorithmus zurückgreift. Das Ziel von Feature Selektion hier ist, am Ende ein Modell zu haben, dass nur auf statistisch relevante Variablen zurückgreift. Dazu erstellt der Boruta Algorithmus nach Kursa und Rudnicki (2010) verschiedene unabhängige Bagging Samples, zieht daraus Classification-Trees und beurteilt die Features nach Verlust an Akkurarität. Die relevantesten Variablen waren das Wetter und die Anzahl der Corona Fälle. Mit einer 10-fold Cross Validation fanden sie heraus, dass sich die Support-Vector-Machines und die Random Forests am besten geschlagen haben. So lässt sich mit Random Forests ein Test  $R^2$ -Wert von 93 % erreichen und mit Support-Vector-Machines ein Test  $R^2$ -Wert von 90 %, was beides sehr gute Werte sind. Dabei spielten auch die sozioökonomische Daten eine Rolle, die im Gesamten zwar wenig Relevanz hatten, aber Unterschiede zwischen den Docking Stationen erklären konnten.

Li et al. (2022) möchten nicht nur den Bestand von Rädern an fixen Fahr-

radausleihstationen vorhersagen, sondern die generelle Nutzung und das Verkehrsvolumen von Leihräder in einem Gebiet berechnen. Dieses Ziel geht schon eher in die Richtung der Fragesetzung dieser Masterarbeit, beschränkt sich jedoch auf Leihräder und nicht auf den gesamten Radverkehr. Dazu verwenden sie Daten von festen Fahrradstationen in Chicago und New York und Daten von Ausleihsystemen mit frei stehenden Fahrrädern, die GPS-Daten speichern, in Singapore und New Taipei City. Jede Stadt zerteilen sie in einem Raster und wenden darauf Convolutional-Neural-Networks an, die auch oft für die Bilderkennung verwendet werden. Das Ergebnis dessen ist in der Abbildung 2.1 zu sehen.

Eine Besonderheit dieser Arbeiten, die alle auf Daten von Bike-Sharing-



*Fig. 2.1:* Anzahl der Leihräder (A) Singapore zwischen 8-9 Uhr am 25. Juni 2017; (B) New Taipei City zwischen 18-19 Uhr am 1. August 2018; (C) Chicago zwischen 16-18 Uhr am 26. August 2019; (D) New York zwischen 17-18 Uhr am 9. Juni 2014. **Quelle:** Li et al. (2022)

Diensten beruhen, ist, dass sie mehr Datenpunkte in einer Stadt zur Verfügung haben als die Studien, die nur Daten von Zählstationen verwenden,

da ein Netz von Leihstation eine höhere Dichte haben muss. So können Gao und Chen (2022) z.B. 2098 Stationen in einer Stadt beobachten, während Broucke et al. (2019) z.B. nur 12 Zählstationen in einer Stadt zur Verfügung haben. Deshalb ergibt das Clustering von Bike-Sharing-Daten Sinn, wie es z.B. auch Xu et al. (2013) und Li et al. (2015) vornehmen.

Modelle die auf Zählstationen beruhen, können den gesamten Radverkehr an bestimmten Punkten messen und schätzen. Die hier kennen gelernten Modelle, die auf Bike-Sharing-Daten zurückgreifen, können das nicht, sondern ermitteln nur das Verkehrsvolumen, das von diesen Mieträdern ausgeht. Dafür können diese Modelle aber immer für den vollständige Verkehr dieser Mieträder schätzen. Interessant wäre die Frage, wie Mietradverkehr und Verkehr von Rädern in Privatbesitz miteinander korrelieren. Wer mit dem Fahrrad regulär pendelt, für den ist der Besitz eines Fahrrads langfristig kostengünstiger. So kann es sein, dass Spitzen in beiden Varianten von einander abweichen, weil Mieträder möglicherweise eher touristischen statt utilitaristischen Zwecken dienen. Mietraddaten allein sind also kein ausreichendes Mittel, um den gesamten Fahrrad Verkehr zu modellieren und sind gleichzeitig auch weniger zugänglich als Daten öffentlicher Zählstationen. Möglicherweise gäben aber Handy Daten mehr Aufschluss.

### 2.1.3 Forschung mit GPS- und Handy Daten

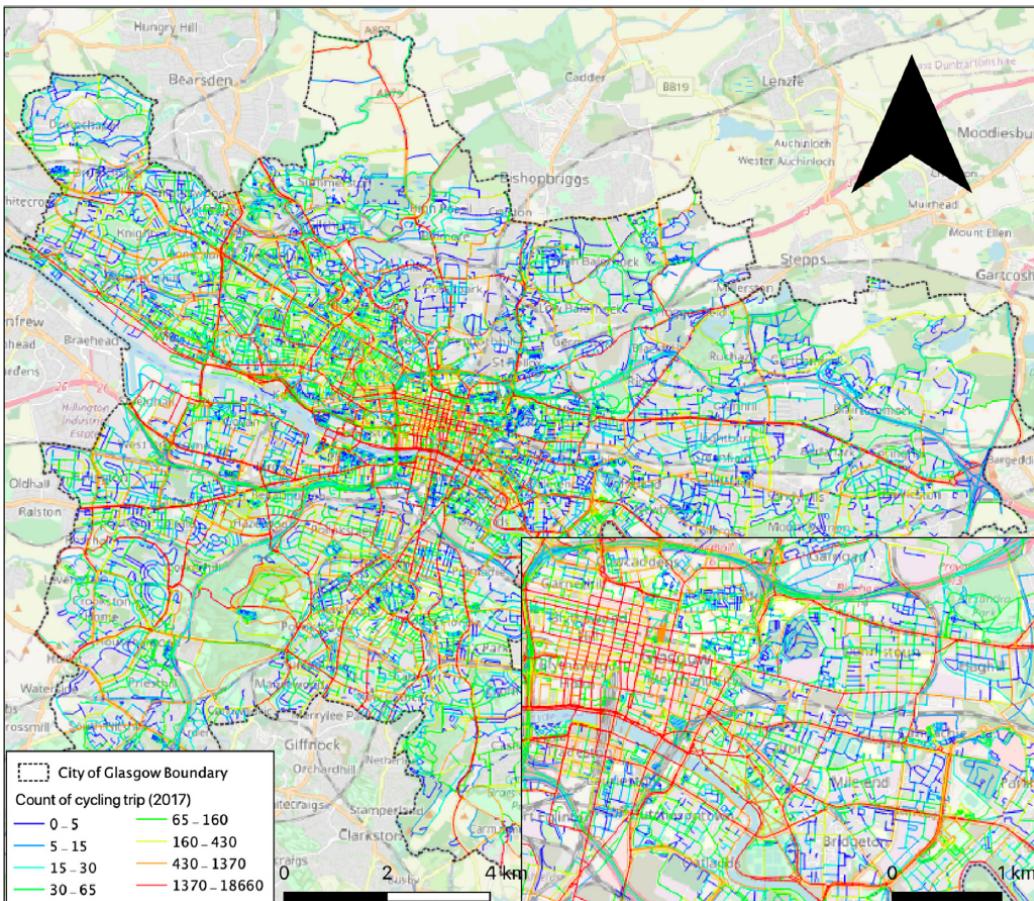
Schon im vorangegangenen Abschnitt zu den Daten von Bike-Sharing-Systemen haben Li et al. (2022) GPS-Daten von Leihfahrrädern für ihre Analyse genutzt. Doch das ist nicht die einzige Möglichkeit, um an GPS-Daten zu kommen. Ein weiterer Weg sind Handy Applikationen, die beständig GPS-Koordinaten aufzeichnen und damit Bewegungsprofile erstellen. Romanillos et al. (2016) stellen solche Studien vor, die auf GPS-Daten von Fitness- und Leisure-Applikationen zurückgreifen.

Eine der ersten Arbeiten wurde von Harvey und Krizek (2007) mit der Absicht erstellt, durch ihr Modell eine Priorisierung von Fahrradinfrastruktur zu ermöglichen. Dazu verwendeten sie noch GPS-Logging-Ausrüstung bei 51 Teilnehmern in einem Zeitraum von 3 Wochen in South Minneapolis, um fa-

vorisierte Radstrecken aufzuzeichnen. Neben der niedrigen Stichprobenmenge war diese Studie noch mit dem Problem des GPS-Cleanings konfrontiert, das zu Positionsabweichungen führen kann. Diese Probleme wurden von Folgestudien wie von Menghini et al. (2010) erstmals behoben, die in Zürich Daten von 2400 Teilnehmern hatten und mit einem verbessertem Detection Algorithmus auch Probleme des GPS-Post-Processing behoben haben.

Das Prinzip der GPS-Aufzeichnung entwickelte sich weiter und Reddy et al. (2010) verwendeten erstmals Handys als GPS-Logging-Geräte. Dazu verwenden sie Daten der App Biketastic. Das Problem bei Bike-Logging-Apps ist, dass hier eher Fahrradrouten aufgezeichnet werden, die der Erholung dienen und nicht unbedingt dem alltäglichen utilitaristischen Stadtverkehr, da es sich oftmals um Apps handelt, die der sportlichen Aktivität dienen. Die Nutzung dieser Daten gibt also nicht Aufschluss über den gesamten Fahrradverkehr. Bei der hier genutzten Applikation Biketastic ist das jedoch anders, denn Biketastic wird speziell von Pendlern genutzt, um nicht nur GPS-Daten aufzuzeichnen, sondern auch Fotos und Audioaufnahmen der Handys, um Straßen mit signifikant hohem Lärm ausfindig zu machen und so lauten Straßenverkehr in das Modell aufzunehmen. Bei der Evaluation stellten sie schnell fest, dass Nutzer der App dazu tendierten, diese nur für lange Strecken zu nutzen, und sich nicht die Mühe machen, die App auf dem Handy zu betätigen, wenn man eine Strecke von unter einer Meile zurücklegen wollte. Eine weitere Studie stammte von Broach et al. (2012), aber auch hier ist die Anzahl der Studienteilnehmer mit 164 Radfahrern gering. Weitere Studien in dem Bereich stammen von Musakwa und Selala (2016), Pritchard (2018), Lee und Sener (2020) und Alattar et al. (2021). Eine der aktuelleren Studien ist von Alattar et al. (2021). Sie verwenden Daten, die durch die Fitness App Strava in Glasgow 2017 bis 2018 gewonnen wurden, um den Einfluss des Straßenlayouts auf die Routenwahl der Fahrer zu untersuchen. Dabei bauen sie auf ein räumliches Modell, das an bestimmten Straßenknotenpunkten stündliche durchgehende Fahrradfahrten nach Straße, Abfahrtsort und Zielort betrachtet. Um zu testen, wie gut die Strava Daten den tatsächlichen Radverkehr abbilden, wurde mit insgesamt 36 temporären Zählstation das Radverkehrsvolumen über zwei Tage stichprobenartig verglichen. Neben den

Daten von Strava nutzten die Autoren das Python Paket OSMnx, dass direkten Zugang zu Open-Street-Map bietet. Open-Street-Map bietet umfassende Information über Glasgows Straßennetz und so können die Knotenpunkte mit dem Straßennetz verglichen werden. Das daraus resultierende Ergebnis ist in Abbildung 2.2 zu sehen. Auf diese Daten kann nun eine Regression



*Fig. 2.2: Fahrradaufkommen für Glasgow 2017*

**Quelle:** Alattar et al. (2021)

angewendet werden, deren erklärende Variablen logarithmerte Indikatoren der Zentralität innerhalb des Knotennetzwerkes sind, die die Anzahl der Fahrradtrips erklären sollen. Bei diesem Regressionsmodell kommt ein Erklärungswert  $R^2$  von 42 % zustande, was im Vergleich zu vorherigen Werten gering ist. Ein interessanter Forschungspunkt den Alattar et al. (2021) hätten hier noch verfolgen können, wäre wie sich die Vorhersagekraft ändert, wenn

man Wetter, Feier und Ferientage in das Modell mit aufnimmt.

Die Studie von Alattar et al. (2021) geht mit der Räumlichkeit ihres Modells schon sehr in die Richtung der Forschungsfrage dieser Masterarbeit. Open-Street-Map kann eine wesentliche Datenquelle sein. Das Ergebnis des Erklärungswertes von 42 % im Modell von Alattar et al. (2021) sollte mindestens auch erzielt werden und kann als Referenz dienen.

Im Überblick zeigen sich häufige Schwächen von Studien, die Handy Applikationsdaten nutzen wie den Strava-Datensatz. Zunächst unterliegen solche Apps häufig einer Selbstauswahl der Probanden. So sind z.B. in dem Datensatz von Alattar et al. (2021) Frauen unterrepräsentiert. Häufig dienen diese Apps der sportlichen Betätigung und nicht der Anfahrt zum Arbeitsplatz. Zudem sind ihre Daten schwer zugänglich.

Eine neuere Quelle für solche GPS-Daten ist die DB Rad+ App. Dies ist eine Tracking App der Deutschen Bahn, die mit dem Fahrrad gefahrene Routen aufzeichnet, und für jeden Kilometer Rabatte bei beteiligten Partnern verspricht. Vorteil zu den vorherigen Studien, die ebenfalls App-Daten verwendeten ist hier, dass diese App nicht allein auf Sportler ausgelegt ist und dass durch die Rabatte auch ein zusätzlicher Anreiz entsteht, auch kurze Radstrecken zu tracken. Bisher machen nur einige wenige Städte bei dieser Aktion mit, darunter ist auch Hamburg. Leider wurden diese Daten zu spät veröffentlicht und finden deswegen keine Beachtung mehr in dieser Masterarbeit. Zukünftige Studien könnten sich diese Daten jedoch zu Nutze machen.

## 2.2 Faktoren des Radverkehrs

Die Daten, die bisher am zugänglichsten waren, sind die Daten von Leihstationen. Mit ihnen lässt sich der Fahrradverkehr gut messen. Doch zielt die Forschungsfrage nicht darauf ab, den Radverkehr allein zu messen, sondern ihn vorherzusagen. Dazu benötigt es Daten, die im Zusammenhang mit dem Radverkehr stehen und einen wesentlichen Faktor bilden. Solche Faktoren können als Prädikator genutzt werden. Die Prädikatoren eines Modells spielen eine große Rolle, denn mit der Auswahl der richtigen Prädikatoren steht und fällt die Validität des Modells. Deswegen klärt ein gesonderter

Literaturüberblick die Tragweite einzelner Variablen, die möglicherweise das Aufkommen im urbanen Fahrradverkehr erklären können. Wichtiges Attribut eines Prädikators ist zudem die Datenverfügbarkeit.

### 2.2.1 Faktor Wetter

Der vorausgehende Abschnitt, der einen Blick auf die bestehende Literatur nahm, erwähnte oftmals Wetterdaten, die in verschiedenen Modellen genutzt worden, so z.B. bei Holmgren et al. (2017), Broucke et al. (2019) oder Li et al. (2015). Literatur, die sich speziell mit diesem Zusammenhang beschäftigt, findet sich bei Wessel (2020). Dabei hat er sich z.B. im Gegensatz zu Nankervis (1999), der sich auch mit dem Zusammenhang von Wetter und Fahrradaufkommen beschäftigt, gezielt mit dem Zusammenhang von Wettervorhersagen beschäftigt. Auch Meng et al. (2016) untersucht diesen Zusammenhang.

Jedoch während Meng et al. (2016) Daten einer Umfrage von 553 Fahrradfahrern in Singapore verwendet, stützt sich Wessel (2020) auf umfangreichere Daten von 188 Fahrradzählstationen in 37 deutschen Städten, die stündlich zählen. Daten zum aktuellen Wetter stammen vom Deutschen Wetterdienst. Daten über Wettervorhersagen stammen aus der ARD-Mediathek, der abendlichen Tagesschau um acht Uhr. Die Aufzeichnungen der Wettervorhersagen werden dabei manuell ausgewertet, wobei der deutsche Raum in sechs Hemisphäre Nordwest, Nordost, mittlerer Westen und mittlerer Osten so wie Südwest und Südost unterteilt wurden und die Städte in die korrespondierende Hemisphäre eingeteilt worden sind nach folgenden Wetterklassifikationen: klarer Himmel, leichte Bewölkung, schwere Bewölkung, Regen, Schneefall, Gewitter und zusätzlich Wind, Rutschgefahr, Vereisungen, Überflutung und generelle Warnungen. Neben dieser manuellen Einschätzung der Bewölkung wurde zusätzlich eine digitale automatisierte Einschätzung verwendet, die sich auf die Dunkelheit der Pixel des Kartenmaterials bezieht.

Aufbauend auf diesen Daten nutzt Wessel (2020) ein log-lineares Regressionsmodell und ein negativ binomiales Regressions Modell. Das letztere Modell geht auf Hausman et al. (1984) zurück und ist im Besonderen für ganzzahlige Regressoren nützlich, wie es hier der Fall ist. Der Bestimmtheitswert

$R^2$  dieser verschiedenen Modelle, die teils abweichende Variablen verwenden, liegt zwischen 75,9 % und 78,5 %. Gerade der Schritt mehrere Städte in ein Modell zu bringen ist interessant. Saha et al. (2018) hat zwar eine Betrachtung für einen ganzen Bundesstaat angefertigt, hat Vorhersagen jedoch nur auf Makroebene getroffen. Das Modell von Wessel (2020) findet hingegen auf der Mikroebene der einzelnen Zählstationen statt, und zeigt, dass trotz der Unterschiede in der städtischen Infrastruktur und Fahrkultur präzise Vorhersagen machbar sind, denn der Einfluss des Wetters ist einer der wesentlichen Prädikatoren und gehört auf jeden Fall in ein Modell, dass das Fahrradverkehrsvolumen vorhersagen möchte. Außerdem verwendet sein Modell auch Schul- und Semesterferien, sowie Feiertage als Prädikator, die ebenfalls stark ins Gewicht fallen.

Auch wenn das Modell von Wessel (2020) zu guten Vorhersagen des Radverkehrs führt, muss diese Masterarbeit darauf verzichten, Information der Wettervorhersagen händisch in Variablen zu übersetzen, da dies mit einem enormen Arbeitsaufwand verbunden wäre. Es ist bedeutend einfacher Daten des Deutschen Wetterdienstes zu nutzen, die sicherlich ähnlich gute Vorhersagen liefern, denn der Unterschied zwischen beiden Varianten ist auch bei Wessel (2020) nicht allzu groß.

Zusätzlich beschäftigen sich Goldmann und Wessel (2021) mit der sogenannten Wetterelastizität des Radverkehrs. Diese beschreibt die Reaktion von Radfahrern auf Wetterereignisse und unterscheidet sich in verschiedenen Städten. Ist der Radverkehr in einer Stadt inelastisch, so bedeutet das, dass selbst bei Regen noch relativ viele Radfahrer auf den Straßen unterwegs sind, wo in elastischen Städten unter selben Bedingungen bedeutend weniger Radfahrer unterwegs wären. Ein Beispiel für eine wetterinelastische Stadt ist z.B. Münster.

### 2.2.2 Faktor Feinstaubbelastung

Die Feinstaubbelastung ist ein weiterer möglicher Prädikator, das zeigen Zhao et al. (2018), Gao und Chen (2022) und Hong J (2022). Im Speziellen setzt sich Hong J (2022) mit der Nutzung von Bike-Sharing-Systemen in Seoul

unter der Aussetzung von Feinstaubbelastung auseinander. Er argumentiert, dass durch die Corona-Krise, mehr Menschen vom öffentlichen Verkehr auf Fahrräder umgestiegen sind, um Menschenmassen in U-Bahnen aus dem Weg zu gehen, dabei aber einer höheren Feinstaubbelastung ausgesetzt waren. Ob Fahrradfahrer der Feinstaubbelastung aus dem Weg gehen, untersucht er mit einer linearen Regression, in der das PM<sub>2.5</sub> Level als Maßstab der Feinstaubbelastung herangezogen wird. Daneben berücksichtigt er Wochentage, Jahreszeiten und Wetterdaten wie Wind, Bewölkung und Temperatur. Demnach hat das PM<sub>2.5</sub> Level einen negativen Effekt auf die gesamte Dauer aller Fahrradtouren auf einem Signifikanz Level von 0.05.

Grundsätzlich sind Daten zur Feinstaubbelastung in Deutschland an vielen Stellen<sup>1</sup> erhältlich, doch ist zu bezweifeln, dass hier der Aufwand im Verhältnis zum Nutzen stünde. Denn in den allermeisten deutschen Städten dürften grundsätzlich andere Verhältnisse vorherrschen als in Seoul. So lag nach OECD Daten<sup>2</sup> die durchschnittliche Feinstaubbelastung PM<sub>2.5</sub> 2019 in Südkorea bei 45.2 und in Deutschland bei 11.9 Mikrogramm per m<sup>3</sup>. Zusätzlich ist zu bedenken, dass ohne öffentliche Smogwarnungen Fahrradfahrer keine Möglichkeit haben, auf gestiegene Feinstaubbelastungen zu reagieren, oder diese bei Auswahl ihrer Fahrradrouten zu berücksichtigen. Demnach dürfte die Feinstaubbelastung als Faktor zur Vorhersage des Fahrradverkehrsaufkommens in Deutschland irrelevant sein.

### 2.2.3 Faktor Corona-Maßnahmen

Es ist vorstellbar, dass Corona-Lockdowns zu einer Verzerrung des allgemeinen Verkehrs geführt haben, was ebenso für Fahrräder gelten würde. Diesen Effekte untersuchen Möllers et al. (2021) mittels Daten von Zählstation in 10 deutschen Städten. Während der Lockdown zu einer Reduzierung von Fußgängern führte, ist der Effekt auf Fahrradfahrer uneindeutig. Bei dieser Betrachtung konzentrieren sich Möllers et al. (2021) auf die erste Corona-Welle. Für ihr Modell verwenden sie Daten des Robert-Koch-Institutes, dass

<sup>1</sup> <https://openseNSEmap.org> (letzter Zugriff: 13.2.2023)

<sup>2</sup> <https://data.oecd.org/air/air-pollution-exposure.htm> (letzter Zugriff 13.2.2023)

tägliche Daten über neue Fälle zur Verfügung stellt je Region. Zusätzlich kontrollieren sie auch die örtlichen Maßnahmen durch die Öffnungszeiten örtlicher Geschäfte und Schulen. Ihre Analyse zeigt, dass die Anzahl der Fahrradfahrer unter der Woche abgenommen hat, aber an Wochenenden zu genommen hat.

#### *2.2.4 Faktor der städtischen Unterschiede*

Die Studien von Wessel (2020), Möllers et al. (2021) und Goldmann und Wessel (2021) zeigten bereits, dass die Verwendungen von Daten aus verschiedenen Städten immer noch zu guten Vorhersagen führt, selbst wenn Unterschiede in der Höhe des Fahrradverkehrs in diesen Städten existieren. Dieser Punkt spielt für diese Masterarbeit eine große Rolle, denn die Daten von verschiedenen Städten zu vereinen, wäre ein hilfreicher Weg, um mehr Stichproben für ein schlüssiges Modell zu erhalten.

Damit die Aussagekraft des Modells jedoch in den Unterschieden der Städte nicht untergeht, empfiehlt es sich, Variablen in das Modell mit aufzunehmen, die diese Unterschiede erklären können. Goldmann und Wessel (2021) machen dies vor. Sie versuchen Unterschiede der Städte in der Wetterelastizität des Radverkehrs zu begründen. Dazu nutzen sie nicht nur Wetterdaten und Daten zu Feier- und Ferientagen sondern auch Daten zur demographischen Bevölkerungsstruktur der jeweiligen Stadt, Daten zum Autobesitz, die Unfallrate, die Verkehrsdichte, Radwegdichte und die Dichte des öffentlichen Nahverkehrs. Diese Auswahl ergibt einen guten Anhaltspunkt dafür, welche Variablen der Datensatz dieser Masterarbeit mit aufnehmen sollte. Weitere Angaben und Ergebnisse hierzu finden sich im Kapitel 3.3.

#### *2.2.5 Faktor Radverkehrsunfälle*

Da diese Arbeit beabsichtigt, die Auslastung von Fahrradwegen und das allgemeine Verkehrsaufkommen von Fahrrädern hervorzusagen, haben sich die bisherigen Abschnitte darauf konzentriert, Datenquellen aus der Literatur vorzustellen, die zur Erstellung von Radverkehrsmodellen genutzt worden sind. Interessant wäre es, die Erkenntnisse aus diesen Daten mit dem Aufkom-

men von Fahrradunfällen zu verbinden, um z.B. kritische Stellen im Stadtbild als potentielle Verkehrsunfallorte ausfindig zu machen. Um in Erfahrung zu bringen, ob man durch statistische Methoden eine Antwort auf diese Nebenfrage finden kann, ist es wichtig, Literatur zu betrachten, die Unfallstatistiken als Datengrundlage verwendet.

Eine solche Studie stammte z.B. von Vandenbulcke et al. (2014), die der Frage nachgehen, wie Infrastruktur Unfälle hervorruft. Dazu verfolgen sie einen räumlichen bayesianischen Modellierungsentwurf mit Daten aus Brüssel. Ergebnisse dieser Studie sind z.B., dass das Gefahrenpotential für Fahrradfahrer steigt, wenn sich z.B. Straßenbahnschienen auf der Fahrspur befinden. Weitere Einflussfaktoren sind Brücken ohne separate Fahrradwege, komplizierte Kreuzungen, die Nähe zu Einkaufszentren oder Garagen und ein erhöhtes Busaufkommen.

Prati et al. (2017) verwenden ebenfalls einen bayesianischen Ansatz. Zum einen verwenden sie eine bayesianische Netzwerk Analyse und einen CHAID-Decision-Tree, um die Schwere von Fahrradunfällen anhand von Charakteristiken wie Geschlecht und Alter der Fahrradfahrer, Art des Fahrzeuges des Unfallpartners, Art des Straßenabschnitts etc. vorherzusagen. Vorteil der CHAID-Analyse ist es, dass die Verästelungen des Entscheidungsbaums Gabelungen mit mehr als zwei Pfaden zulässt. Als Datengrundlage verwenden Prati et al. (2017) italienische landesweite Unfallstatistiken von 2011 bis 2013. Das erlaubt den Vorteil einer großen Stichprobengröße mit 49621 Unfällen, bei denen mindestens ein Fahrradfahrer verletzt worden ist. Ein Validation-Set-Approach mit einem 70:30 Split vergleicht beide statistischen Modelle. Im Ergebnis zeigt sich, dass die wichtigsten Faktoren für die Schwere eines Unfalls der Straßentyp, der Unfalltyp, das Alter des Radfahrers, die Straßenbeschilderung, das Geschlecht des Radfahrers, der Typ des gegnerischen Fahrzeugs und der Monat sind. Wobei das CHAID Modell im Testset zu 98 % akkurat war.

Auch Kondo et al. (2018) verfolgen einen bayesianischen Ansatz, entwickeln aber ein räumliches Modell. Sie untersuchen wie Fahrradstreifen das Unfallrisiko senken können. So führen getrennte Fahrradstreifen zu einem 48 % geringeren Risiko an einer Viererkreuzung und zu einem 43 % geringeren Ri-

siko auf Straßen mit hohem Verkehr. Zu diesen Erkenntnissen kommen sie auf Grundlage eines Datensatzes aus Philadelphia mit über 37000 beobachteten Unfällen zwischen 2011 und 2014 mit Charakteristiken der Straßenbeschaffenheit. Dazu verwendeten sie ein Bayesian-Conditional-Autoregressive-Logit-Modell. Als unabhängige Variablen verwenden sie Straßencharakteristiken, Charakteristiken von Kreuzungen und einen Traffic-Indikator. Zu den Charakteristiken zählen z.B. die Anzahl der Keuzungszugängen, Stoppzeichen oder auch Fußgängerüberwege. Mithilfe dieser Daten ergibt sich ein Bild wie in Abbildung 2.3 zu finden: Wie viele der bisher genannten Studien

a) Excluding CAR random effect

b) Including CAR random effect

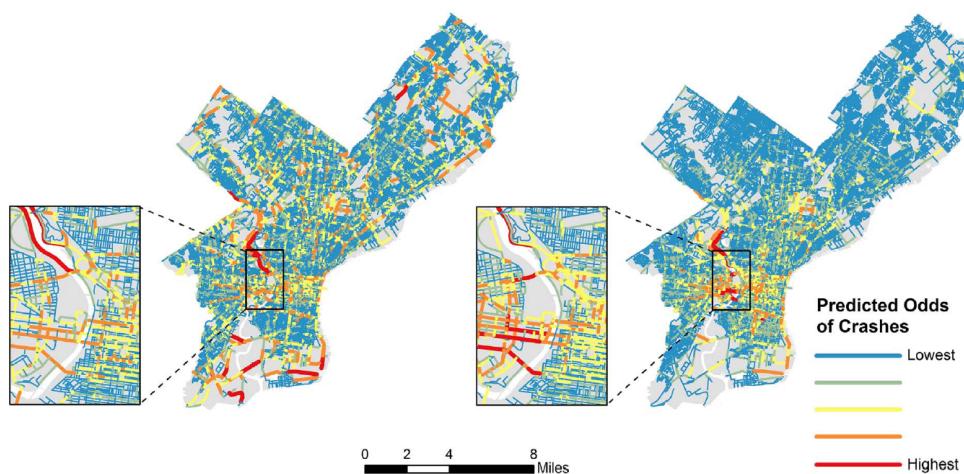


Fig. 2.3: Berechnete Wahrscheinlichkeiten für Radunfälle in Philadelphia

**Quelle:** Kondo et al. (2018)

konzentrierte sich Kondo et al. (2018) auf eine Stadt, was ein nicht unwesentliches Problem darstellen kann, denn unterscheidet sich der Straßenverkehr von Stadt zu Stadt, wie auch Goldmann und Wessel (2021) zeigen. Gerade die Ergebnisse aus Philadelphia, einer der fahrradfreundlichsten Städte der USA, lassen sich eventuell nicht eins zu eins auf andere Städte der USA übertragen, da Autofahrer in Philadelphia durch den höheren Verkehr von Fahrrädern an die Rücksichtnahme gewöhnt sein könnten, die Autofahrer aus Erfahrung mit dem Radverkehr walten lassen. Deswegen wäre es ein interessanter Schritt, diese Beobachtungen auf einer Makro-Ebene zu tätigen, um

zu sehen, welche Faktoren Städte übergreifend für Fahrradsicherheit sorgen. Genau dies machen Saha et al. (2018), in dem Sie eine Analyse Floridas auf Zensus Block Größe vornehmen. Bei dieser Analyse stellen sie eine räumliche Konzentration und keine Gleichverteilung von Unfällen fest. Um dies zu erklären nutzen die Autoren ein bedingtes autoregressives Modell mit bedingten Variablen der Demographie, sozioökonomischen Daten, Straßen Infrastruktur und Radverkehrcharakteristiken. Davon wurden 21 Variablen als einflussreich identifiziert z.B. Bevölkerung, Alterskohorten, Autobesitz von Haushalten, Straßennetzdichte oder Fahrradtourintensität. Für das Modell wurden Zensus Daten von 2011 bis 2014 verwendet, zu sehen auch in der Abbildung 2.4. Gerade die letzten zwei Studien zeigen, wie interessant die

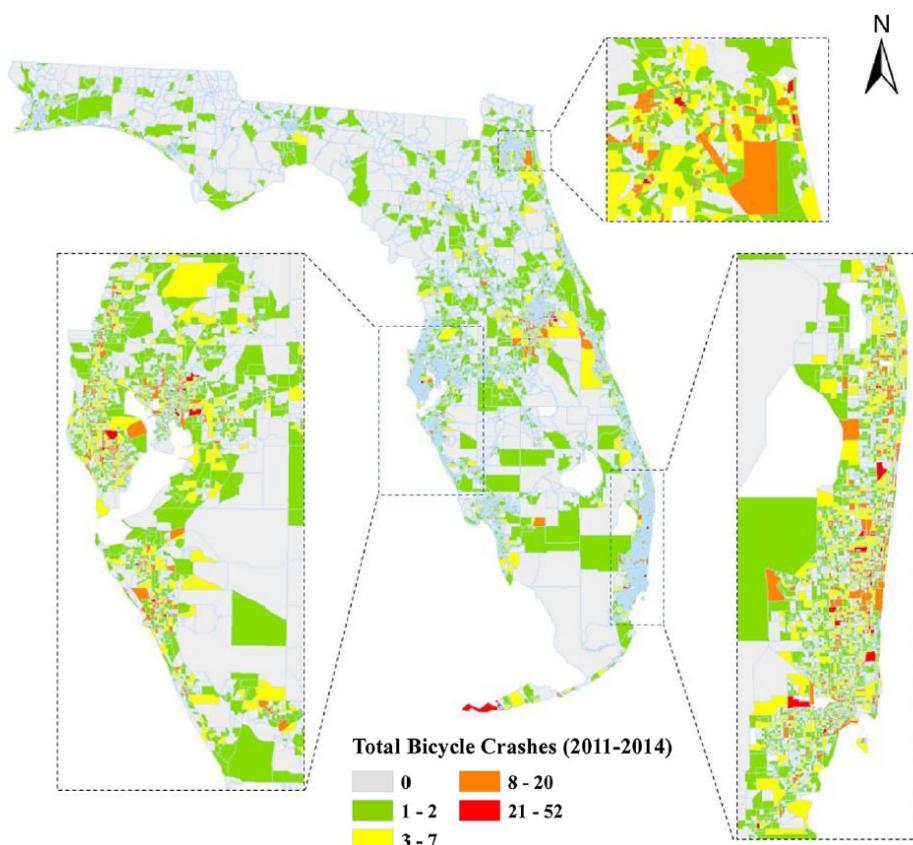


Fig. 2.4: Räumliche Verteilung aller Fahrradunfälle (2011 bis 2014) nach Zensus Blöcken in Florida.

Quelle: Saha et al. (2018)

räumliche Verteilung von Fahrradunfällen ist und Kondo et al. (2018) zeigen auch das Verkehrsvolumen als Faktor der Unfallwahrscheinlichkeit.

Neben all dem ist interessant, dass Kondo et al. (2018) darauf verweisen, dass Unfallsicherheit ein starker Prädikator sei für den Fahrradverkehr laut Pucher et al. (2010), Thomas und DeRobertis (2013) und Winters et al. (2010). Würden also ausreichend Daten über Unfälle je Stadt und Lage vorhanden sein, dann ließen sich Unfalldaten auch als weitere Variable im Modell dieser Masterarbeit nutzen lassen. Allerdings waren nur Daten aus Münster zu Unfällen bekannt. Diese werden in einem späteren Kapitel dazu genutzt, die räumliche Korrelation zum Fahrradverkehr zu messen.

### 2.2.6 Sonstige Faktoren

Eine weitere interessante Datenquelle, die wir z.B. schon bei Alattar et al. (2021) kennen gelernt haben, ist Open-Street-Map. Dies ist ein 2004 gegründetes gemeinnütziges Projekt, dass das Ziel verfolgt Kartenmaterial zu sammeln und online allen frei zur Verfügung zu stellen. Hier sind Daten über Straßen, Eisenbahnen, Flüsse, Wälder, Häuser und viele weitere Faktoren zu finden. Auch Carl und Dror (2015) nutzen die frei verfügbaren Daten zu Höhenmetern des Geländes von Open-Street-Map, um die Planung von mehrtägigen Fahrradtouren zu erleichtern, die der Erholung dienen. Deswegen ist dieses Paper für die Frage dieser Hausarbeit weniger interessant, die Idee, diese Art von Daten zu verwenden, aber könnte hilfreich sein. Dass aber die Topographie eine Rolle auch für den Pendler Verkehr spielt, zeigen Rietveld und Daniel (2004).

## 2.3 Forschungslücken und Anknüpfungspunkte

Was die meisten der hier gezeigten Studien gemein haben, ist, dass sie sich auf eine reine Zeitreihenanalyse beschränken, jedoch die Daten nicht dazu nutzen, die räumliche Verteilung der Daten zu untersuchen. Es gibt eine wenige Beispiele die das zwar tuen wie z.B. Alattar et al. (2021), jedoch nie für den ganzen Verkehr, sondern nur für den Teil des Verkehrs, der im Rahmen von Leihradssystemen geschieht. Auch die Kontrolle der Routenwahl durch

GPS-Tracking, stellt immer nur einen verzerrten Teil des gesamten Verkehrs dar, entweder durch zu geringe Stichprobengrößen oder durch die Selbstselektion der Studienteilnehmer.

Das lässt nur die Forschung übrig, die mit Daten von Zählstationen rechnet. Diese verfolgt aber meist das Ziel kausale Einflüsse auf den Radverkehr zu untersuchen wie z.B. bei Wessel (2020). Andere Studien treffen zwar Vorhersagen, aber eben nicht in eine räumliche Dimension hinein, wie z.B. bei Holmgren et al. (2017). Die Forschungsfrage dieser Masterarbeit erfordert aber eine räumliche Interpolation, um Vorhersagen für ein gesamtes Stadtgebiet zu treffen und nicht über einzelne Stationen, die vorher so beobachtet wurden. Eine kausale Interpretation ist hierfür nicht zwangsläufig notwendig, wenn auch interessant. Außerdem würde eine kausale Interpretation die Verwendung von neuronalen Netzwerken ausschließen.

Nur wenige Studien untersuchen die Daten von mehr als einer Stadt, mit Ausnahme von z.B. Wessel (2020) oder auch Li et al. (2022). Aber die Verwendung von mehreren Städten wäre zwingend notwendig, denn wie man im Vergleich sieht, bieten Studien mit Radzählstellen oft weniger Beobachtungspunkte pro Stadt, als z.B. Studien die GPS-Tracking-Daten verwenden oder Daten von Leihsystemen. Um diesen Nachteil auszugleichen, ist es ratsam, Daten mehrere Städte zu verwenden. Die Studie von Goldmann und Wessel (2021) zeigt, wie man die Unterschiede zwischen Städten hervor heben kann. So ist es ratsam demographische und soziale Daten über die Stadtstrukturen in einen Datensatz mit aufzunehmen. Bei Goldmann und Wessel (2021) zeigten diese Daten Unterschiede in der Wetterelastizität des Fahrradverkehrs auf. In dieser Studie würden die stadtbezogenen Daten Unterschiede im Radverkehr selbst kenntlich machen.

Zuletzt erfordert die Forschungsfrage auch stationsbasierte Daten, möchte man Unterschiede im Radverkehrsaufkommen innerhalb der selben Stadt erkennen und auch vorhersagen können. Nur daraus ergeben sich räumliche Unterschiede im Stadtbild. Eine Studie von Hankey et al. (2021) dient hier als Vorbild, in dem sie Daten von POIs (Points of Interest) in Google-Streetview verwendet. Hankey et al. (2021) Verwenden hierbei Bilderkennung. Soweit würde diese Masterarbeit nicht gehen. Allerdings könnte man Entfernung

von Zählstationen und einigen POIs im Stadtbild mit in das Modell aufnehmen. Auch Alattar et al. (2021) gab dafür ein Vorbild, die in ihrer Arbeit Open-Street-Map-Daten verwendeten. Genau dies ließe sich als Datenquelle für POIs nutzen. Dies allgemein gibt einen guten Ausblick auf das kommende Kapitel im Allgemeinen. Anhand der bestehenden Literatur ließ sich festmachen, welche Datenstruktur notwendig ist, um die Forschungsfrage zu beantworten. Das kommende Kapitel gibt einen Einblick in den endgültigen Datensatz.

### 3. ZUSAMMENSETZUNG DES DATENSATZ

Das vorherige Kapitel zeigte den aktuellen Stand der Forschung zum Aufkommen des Fahrradverkehrs und endete mit einer Einschätzung, an welchen Studien sich diese Masterarbeit orientieren muss, um die gestellte Forschungsfrage zu beantworten. Darauf baut auch die Beschaffung von Daten für das Modell dieser Arbeit auf, angefangen über die Daten der Fahrradzähler, Daten zum Wetter, demographischen Daten und Daten der vorhandenen Infrastruktur erhoben durch Open-Street-Map und dazu gehörigen POIs. Einen tieferen Einblick über die Datenbeschaffung und die Verteilung von den Daten mit zugehörigen Abbildung beinhaltet dieses Kapitel.

Die Datenbeschaffung an sich beschränkt sich allein auf Deutschland aus Gründen der Einfachheit. Bestimmte Daten, wie Daten zum Wetter und zur Demographie, lassen sich so allein von einer Datenquelle beschaffen, in diesem Fall dem Deutschen Wetter Dienst und dem statistischen Bundesamt (DESTATIS). Der Nachteil davon ist, dass wiederum alle Aussagen des Modells allein für Deutschland gültig sind, da dieser Rahmen die Evaluierung für andere Regionen der Welt nicht zu lässt.

Die Kombination dieser Daten in einem gemeinsamen Datensatz erfolgte in der Programmiersprache R. Auch die darauf aufbauenden Modelle verwenden R. Einen genauen Einblick in den dazu nötigen Quellcode gibt das Github-Repository<sup>1</sup>, welches zur Veranschaulichung dieser Masterarbeit angelegt wurde.

---

<sup>1</sup> [https://github.com/MaxWeinhold/Masterthesis\\_BikeTrafficForecast](https://github.com/MaxWeinhold/Masterthesis_BikeTrafficForecast), letzter Zugriff: 26.2.2023

### 3.1 Fahrradzähler

Notwendige Grundlage der Forschung sind Daten von Fahrradzählstationen. Dankbarerweise stellen viele Kommunen diese Daten öffentlich zur Verfügung oder teilen diese auf Nachfrage. In welchen Städten sich Fahrradzähler finden lassen, die in das Modell aufgenommen werden können, zeigt die Abbildung 3.1. Alle Fahrraddaten sind stündlich aufgelöst. D.h. die Anzahl der Fahrradfahrer, die eine Zählstation passierten, wurde innerhalb einer vollen Stunde aufsummiert.

Ein grundsätzliches Problem des Datensatzes, dass sich z.B. durch Resampling lösen ließe, ist, dass zum einem Großstädte mehr Fahrradzähler errichten, zum anderem ihre Daten oft auch leichter zugänglich machen. Die Abbildung 3.2 zeigt deutlich, dass Großstädte mit mehr als 300 Tsd. Einwohnern im Datensatz überrepräsentiert sind. Deshalb sollte man bei Vorhersagen für Städte mit weniger als 100 Tsd. Einwohnern vorsichtig in der Interpretation sein.

Die Datenquelle sind die Kommunen selbst. Eine Veröffentlichung aller Daten ist dabei nicht möglich, weil die Kommunen unterschiedliche Bedingungen für die Verwendung der Daten gestellt haben. Der Betrachtungszeitraum der Daten reicht von 2012 bis 2022 in einigen Fällen. Einige wenige Zählstellen in Hamburg und Siegen wurden erst 2022 aufgestellt. In allen anderen Städten reichen die Beobachtungen aber nur bis 2021. Die Abbildung 3.3 zeigt, wie sich die Daten über die Zeit verteilen. Dabei ist festzustellen, dass je weiter die Daten in der Vergangenheit liegen, desto weniger Daten finden sich, da nur wenige Zählstationen schon 2012 dauerhaft im Betrieb waren und alle weiteren Zählstationen erst im Verlauf der Zeit aufgestellt wurden. Auch sehr wenige Daten finden sich im Jahr 2022. Diese Daten stammen zum größten Teil aus Hamburg. Hamburg selbst verfügt nur über eine Dauerradzählstation, die seit 2014 im Betrieb ist, hat aber seit 2022 viele weitere Infrarotdetektoren für Fahrräder aufgestellt, die etwas anders funktionieren als die Zählstation, die mittels einer Induktionsschleife zählen. Weil diese Daten relativ flächendeckend zur Verfügung stehen, sind diese in den Datensatz mit eingeflossen.

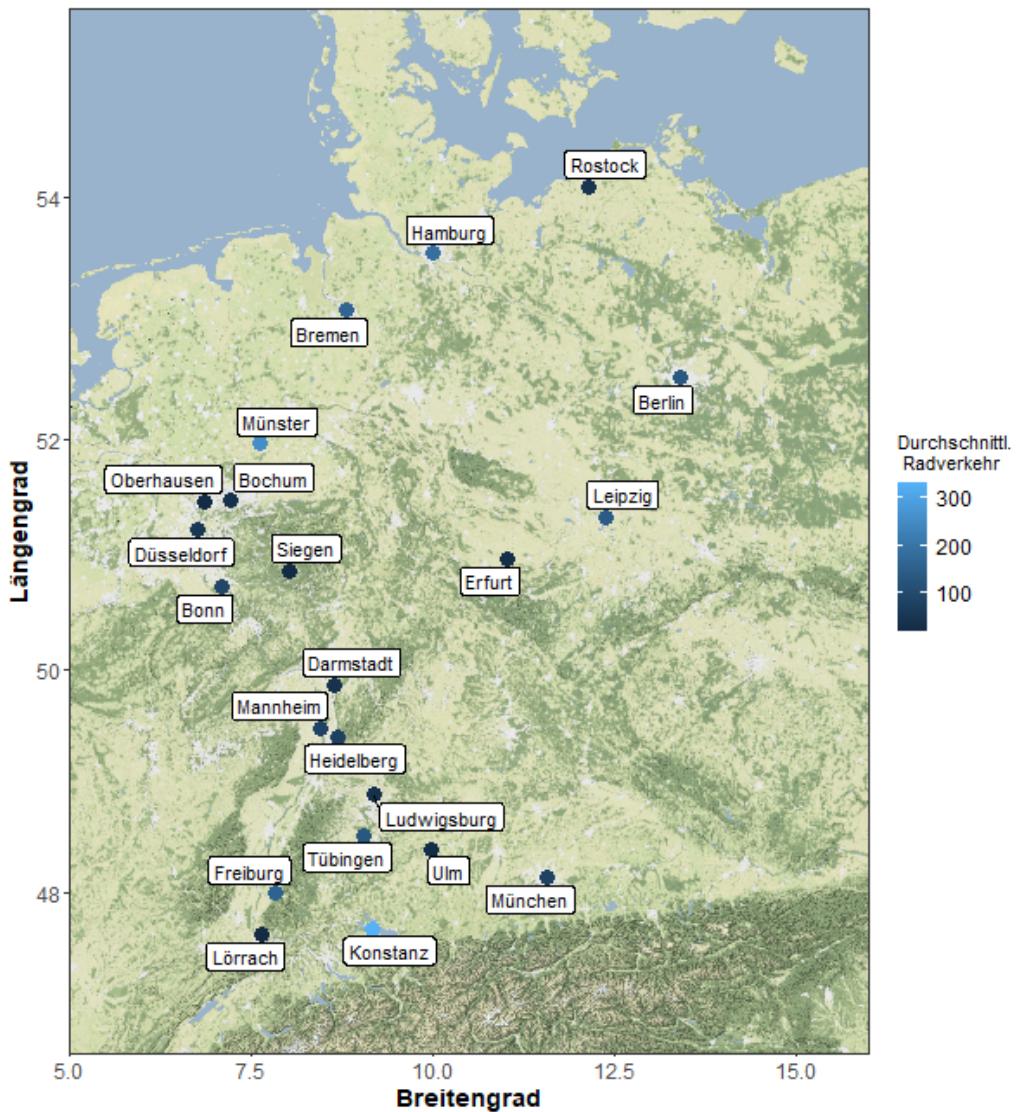


Fig. 3.1: Städte die im Datensatz vertreten sind.

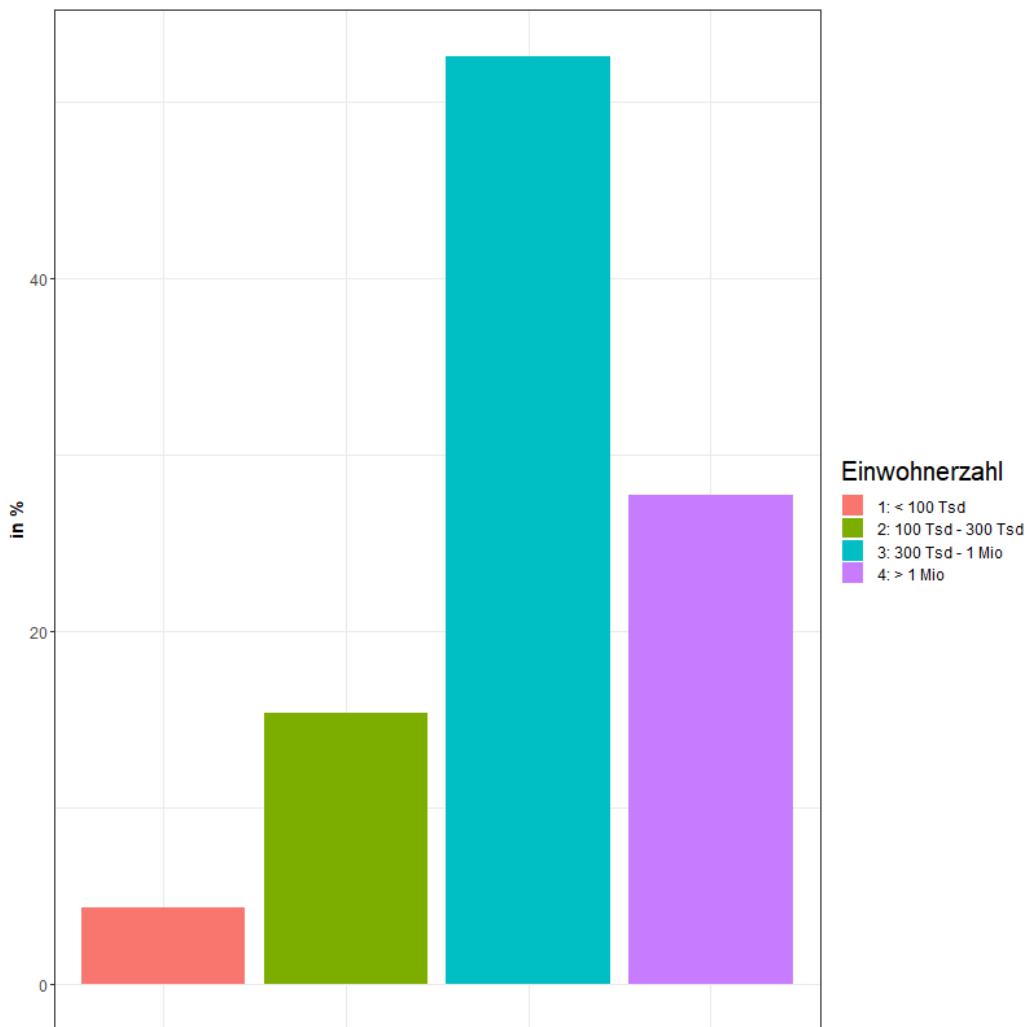


Fig. 3.2: Verteilung der Beobachtung nach Einwohnergröße der Städte

### 3.2 Wetterdaten

Daten zum Wetter stammen einheitlich vom Deutschen Wetterdienst. Dabei wurden die Fahrraddaten einer Stadt immer mit den Wetterdaten der nächstgelegenen Wetterstation verwendet. Nur Mannheim und Heidelberg teilen sich die Daten einer Wetterstation, da beide Städte sehr nah beieinander liegen. Insgesamt wurden Daten in den Datensatz mit aufgenommen zum Niederschlag in mm, zur Lufttemperatur in 2 m Höhe in Grad Celsius, zur Wolkenbedeckung in Achteln, zur relativen Feuchte in % und zur

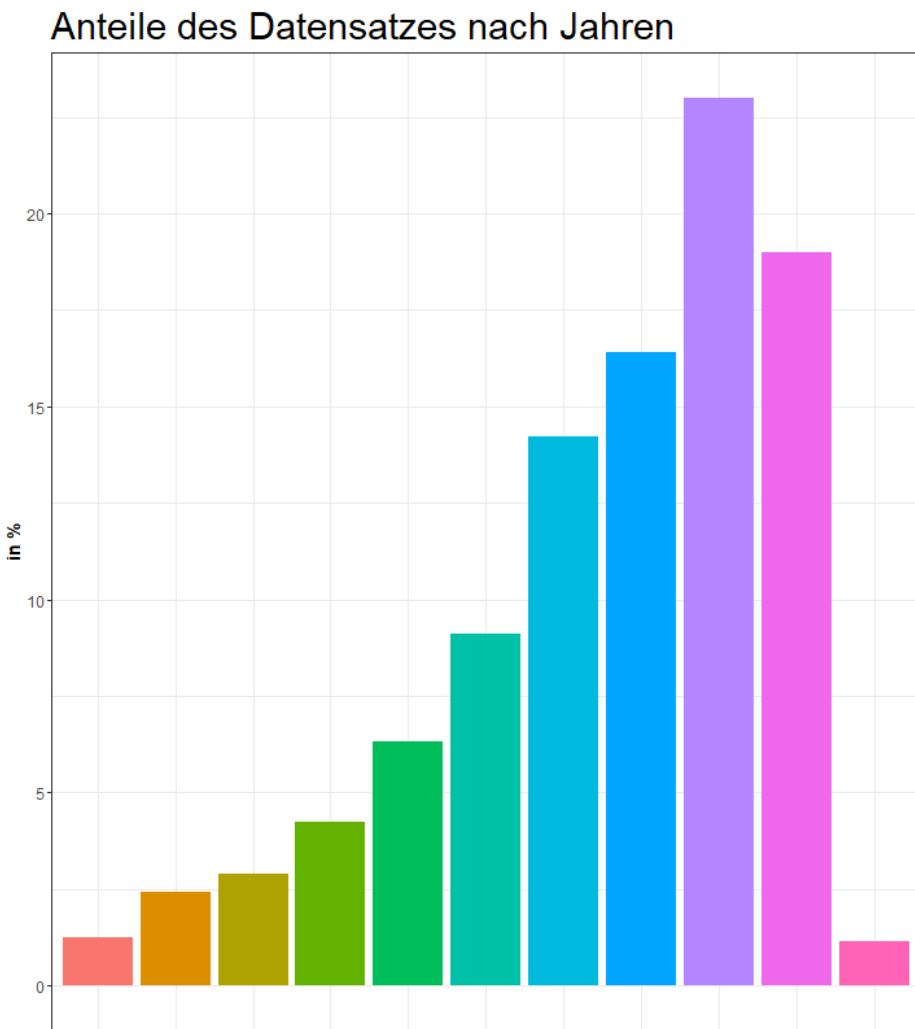


Fig. 3.3: Verteilung der Beobachtung nach Jahren

durchschnittlichen Windgeschwindigkeit. Genau wie die Fahrraddaten sind die Wetterdaten stündlich aufgelöst. Einen jährlichen Durchschnittsverlauf über alle Daten zeigt die Abbildung 3.4a. Weiterhin zeigt die Abbildung 3.4b den Zusammenhang zwischen der Anzahl der Radfahrer und der Monatstemperatur.

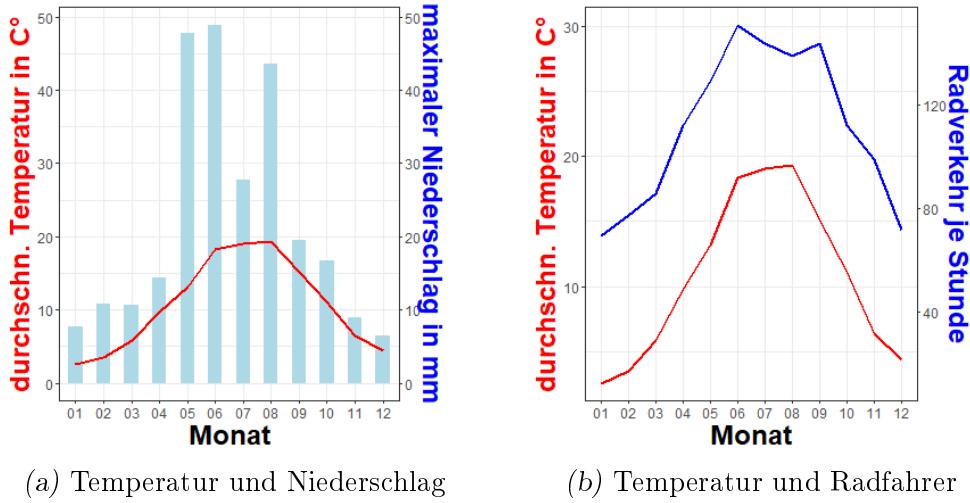


Fig. 3.4: Wetterdaten im Überblick

### 3.3 Demographische und soziale Statistiken

In diesem Abschnitt sammeln sich verschiedene Daten, die Unterschiede in den Städten in ihrer sozialen und demographischen Struktur, in ihrer Bevölkerung, im Autobesitz und im Fahrradklima hervorheben. Daten dafür stammen aus zwei Quellen. Die meisten Variablen nutzen als Datenquelle das statistische Bundesamt (Destatis). Eine Variable bezieht sich auf den ADFC-Fahrradklimaindex. Für beide Quellen gilt, dass Daten für das Jahr 2022 noch nicht vorhanden waren, da das Jahr zum Zeitpunkt der Recherche noch nicht abgeschlossen war. Für die wenigen Beobachtungen aus Hamburg und Siegen z.B., die aus dem Jahr 2022 in den Datensatz mit aufgenommen worden sind, gilt, dass für diese Beobachtungen angenommen wurde, dass Variablen, die nicht erneuert werden konnten, konstant blieben. Das betrifft insgesamt jedoch nur wenige Beobachtungen, wie auch Abbildung 3.3 zeigt.

#### 3.3.1 Statistisches Bundesamt

Das statistische Bundesamt stellt verschiedene Tabellen zur Verfügung, die wertvolle Einblicke in die Unterschiede der Gemeinden geben, in denen sich die verschiedenen Fahrradzählstationen befinden.

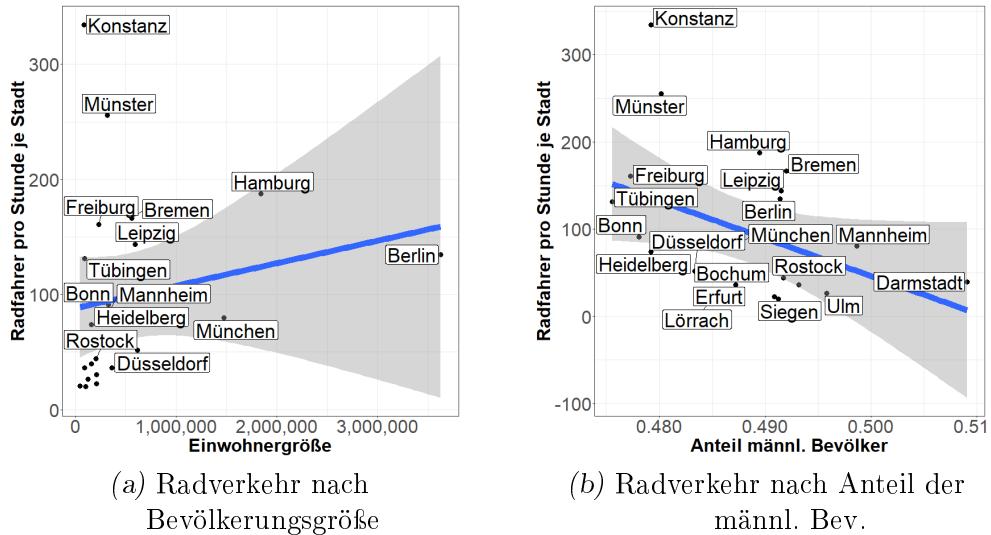


Fig. 3.5: Verhältnis von Bevölkerung und Fahrradaufkommen

Eine wichtige grundlegende Quelle ist das Gemeindeverzeichnis für alle politisch selbständigen Gemeinden (mit Gemeindeverband) in Deutschland, das jedes Jahr veröffentlicht wird. Dieses Verzeichnis beinhaltet nicht nur Landkreise und kreisfreie Städte, sondern hält auch Zahlen zu einzelnen Gemeinden bereit. Dabei sind Daten zur Fläche in km<sup>2</sup>, zur Einwohneranzahl, sowohl männlich, weiblich als auch insgesamt vorhanden. Diese Daten fanden auch ihren Weg in das Modell. Wie sich Einwohnergröße und der Anteil an Männern in der Bevölkerung zum Aufkommen an Fahrradfahrern verhalten, kann man in der Abbildung 3.5 sehen. Außerdem verwendet das Modell die angegebenen Längen- und Breitengrade der Gemeinden, aus denen der jeweilige Abstand der Fahrradzählstation zum Stadtzentrum berechnet wird, auch dargestellt in Abbildung 3.6. Das Modell greift dabei auf das Gemeindeverzeichnis von 2012 bis 2021 zurück.

Weitere Daten zur demographischen Verteilung der Bevölkerung in den jeweils betroffenen Landkreisen und kreisfreien Städten stammen aus der Tabelle 12411-0017 zur Bevölkerung nach Kreisen, Stichtag, und Altersgruppen von Destatis (2022b). Mit diesen Daten ließ sich der Anteil der Bevölkerung

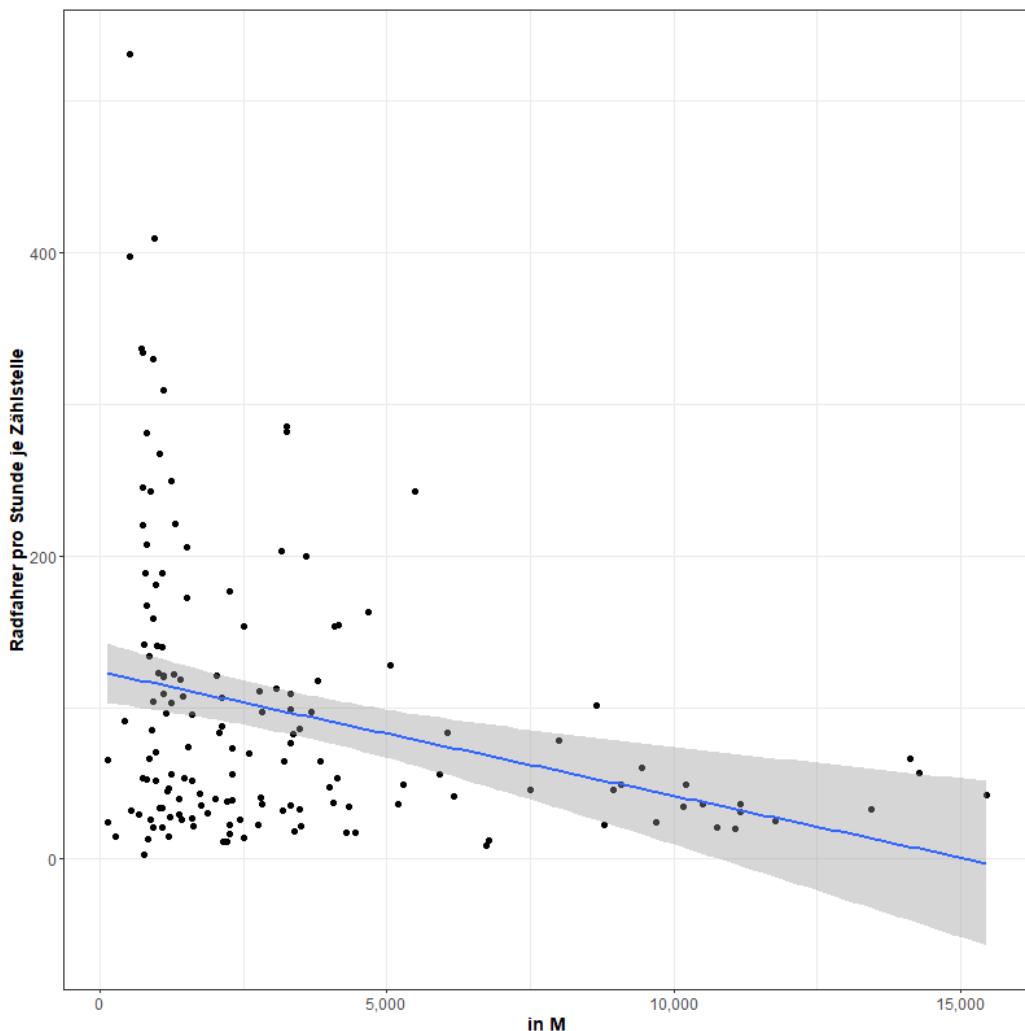


Fig. 3.6: Verteilung des Fahrradaufkommens in Entfernung zum Stadtzentrum

berechnen, der jünger als 18, 25 und 30 und älter als 40 und 60 ist. Wie sich das Radverkehrsaufkommen zum Anteil der Bevölkerung unter 30 verhält nach Städten, zeigt auch die Abbildung 7.1 im Anhang.

Auch eine wertvolle Statistik bietet die Tabelle 46251-0020 über den Kraftfahrzeugbestand nach Kreisen, Stichtag und Kraftfahrzeugarten von Destatis (2022c). Mithilfe der Einwohneranzahl je Kreis ließ sich die Quote an Kraftfahrzeugen je Person berechnen. Außerdem zeigt die Tabelle 12521-0040 über die Anzahl der Ausländer nach Kreisen, Stichtag und Geschlecht

von Destatis (2022a) ebenfalls wichtige Einblicke, zu sehen in Abbildung 3.7.

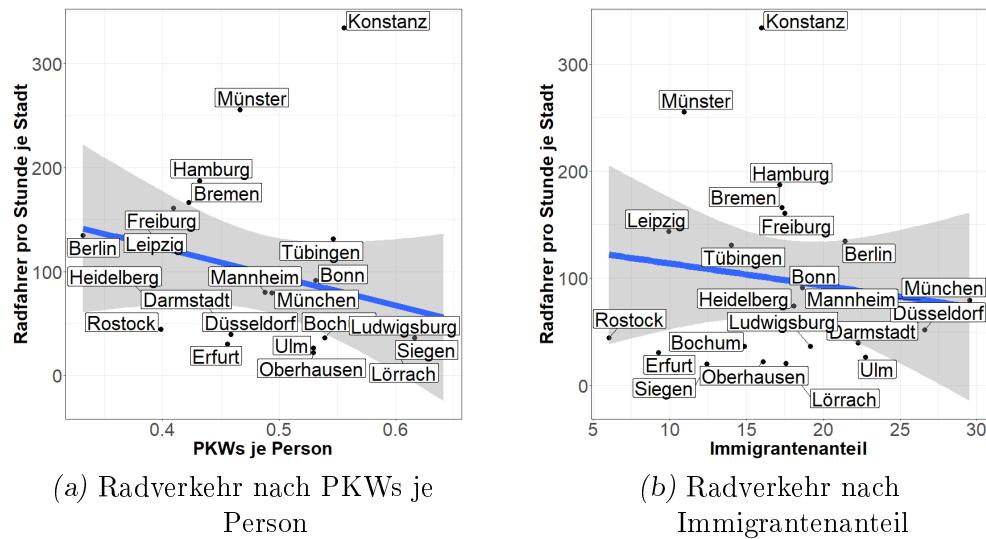


Fig. 3.7: Verhältnis von Bevölkerung und Fahrradaufkommen

### 3.3.2 ADFC Fahrradindex

Neben den stadspezifischen Daten, die von den Quellen des statistischen Bundesamtes (Destatis) stammen, nutzt der Datensatz auch die Daten zum ADFC-Fahrradklimaindex. Der Allgemeine-Deutsche-Fahrrad-Club ist eigener Angaben<sup>2</sup> zur Folge ein eingetragener Verein mit Sitz in Berlin und Bremen, der sich für die Interessen von Fahrradfahrern einsetzt und Stand 2022 220.000 Mitglieder in Deutschland zählt. Im 2-Jahresrhythmus veröffentlicht der ADFC den per Umfrage ermittelten Fahrradklimaindex. 2022 wurden z.B. 238 Tsd. Menschen befragt aus verschiedenen Städten. Neben Fragen zum persönlichen Fahrradfahren enthält der Fragebogen zum Fahrradklimaindex auch Fragen betreffend des Fahrradklimas, dem Stellenwert des Radverkehrs in der jeweiligen Stadt, der Sicherheit, dem Komfort und der Infrastruktur sowie der Radverkehrsnetze. All diese Antworten fließen in eine Endnote, die sich zwischen 1 und 5 befindet, wobei 1 am besten und 5 am

<sup>2</sup> <https://www.adfc.de>, letzter Zugriff 27.2.2023

schlechtesten ist. Wie sich das Fahrradklima ins Verhältnis zum aufgezeichneten Radverkehr nach Städten setzt, zeigt die Abbildung 7.2 im Anhang.

### 3.4 Corona-Daten

Wie Möllers et al. (2021) bereits zeigten, hatten die Corona-Maßnahmen unterschiedliche und nicht ganz eindeutige Auswirkungen auf den urbanen Radverkehr. Dennoch ist es ratsam, Variablen in das Modell mitaufzunehmen, die Ausprägung von Corona-Maßnahmen berücksichtigen. Wie bei Möllers et al. (2021) ließen sich dafür die Corona-Inzidenzzahlen verwenden, oder aber eine Dummy-Variable für Lockdowns.

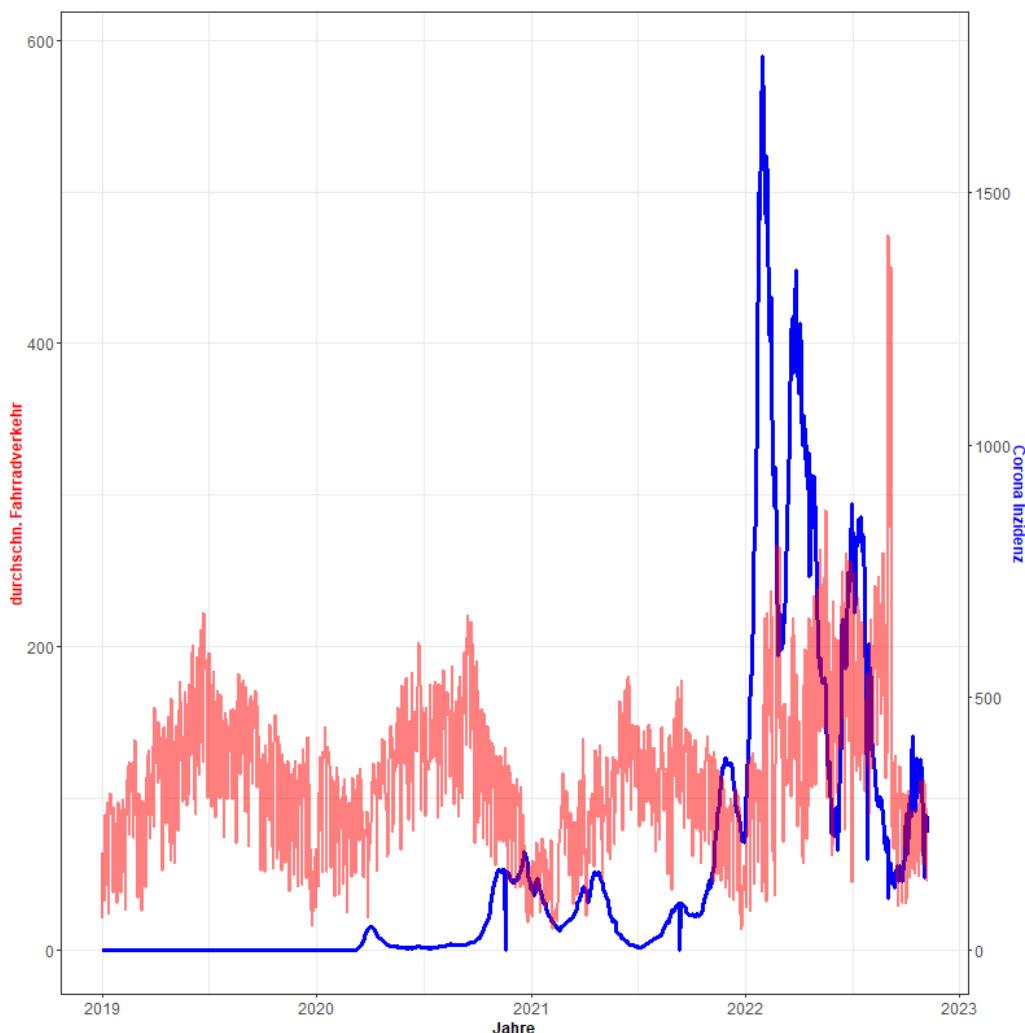
Bei beiden Ansätzen stellen sich Probleme ein. War die Corona-Inzidenz zu Beginn recht klein, waren die Auswirkungen für den Verkehr dennoch drastisch. Die Abbildung 3.8 gibt darüber einen Überblick. Deswegen beinhaltet der Datensatz zusätzlich eine Dummy Variable für die zwei bundesweiten Corona-Lockdowns vom 22.3.2020 bis zum 4.5.2020 nach Kodzo (2022) und vom 2.11.2020 bis zum 14.2.2020 nach Kodzo und Imöhl (2022). In diese Zeitphasen fallen verschiedene Maßnahmen zur Kontaktbeschränkung, die jeweils bundesweit stattgefunden haben. Dabei kann es regional zu Abweichungen kommen, die sich im Nachhinein nicht ohne immensen Aufwand nachvollziehen lassen.

Diese Schwäche sollen die Daten zur Corona-Inzidenz ausgleichen, die lokal mit Maßnahmen zur Kontaktbeschränkung korrelieren. Die Daten dazu stammen vom Robert Koch Institut<sup>3</sup>. Leider reichen die Daten des RKIs nicht für alle einzelnen Kommunen bis zum Beginn der Pandemie zurück. Im Zeitraum des 10.3.2020 bis zum 6.5.2020 sind die im Datensatz enthaltenen Daten zur Corona-Inzidenz bundesweit. Danach und bis zum 18.11.2020 verwendetet der Datensatz die Inzidenzen der jeweiligen Bundesländer. Erst danach sind die Inzidenzen der jeweiligen Landkreise und die der kreisfreien Städte verfügbar.

Bei der ersten Erstellung des Datensatz dieser Masterarbeit, waren die Corona-

---

<sup>3</sup> [https://www.rki.de/DE/Content/InfAZ/N/Neuartiges\\_Coronavirus/Daten/Inzidenz-Tabellen.html?nn=2386228](https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Daten/Inzidenz-Tabellen.html?nn=2386228) (letzter Aufruf: 14.2.2023)



*Fig. 3.8: Verlauf der Corona Inzidenz und des Radverkehrs*

Daten leider noch nicht mit inbegriffen und erste Modelle wurden ohne diese Daten berechnet. Daraufhin wurde ein zweiter Datensatz erstellt, der nicht nur die Corona-Daten zusätzlich berücksichtigte, sondern auch neuere Daten zu Straßen anlegte, dazu mehr im Detail im folgenden Abschnitt. Im weiteren Verlauf des Textes wird kenntlich gemacht, mit welcher Version des Datensatzes Berechnungen gemacht wurden, ob mit der alten Version ohne Corona-Daten oder der neuen Version mit Corona-Daten.

Anders als bei Möllers et al. (2021) gibt es in dem neuen Datensatz keinen

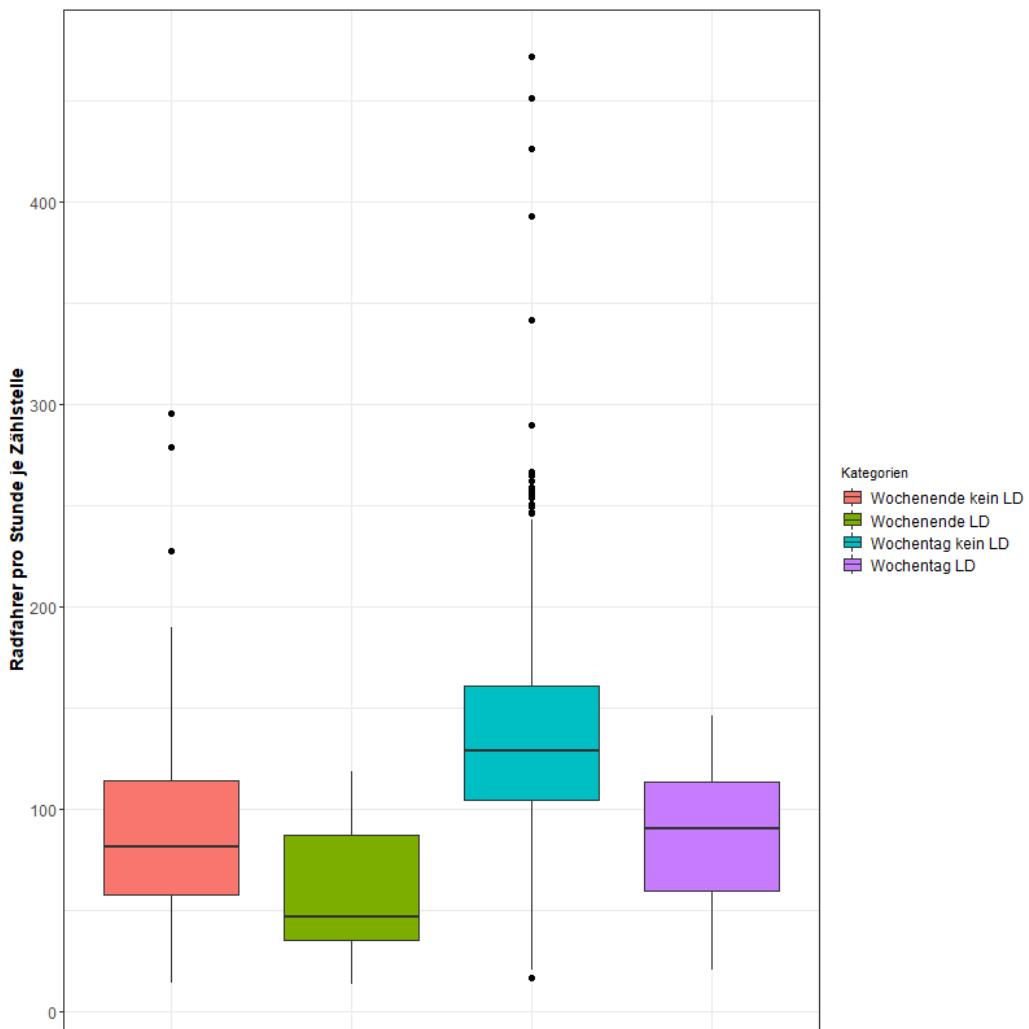


Fig. 3.9: Radverkehr im Lockdown

zwiespältigen Effekt, wie die Abbildung 3.9 zeigt. Sowohl am Wochenende, als auch unter der Woche ist das Aufkommen des städtischen Radverkehrs gesunken. Dies kann auch an der Auswahl anderer Städte für den Datensatz liegen oder aber an der Auswahl der Variablen, da Möllers et al. (2021) genauere Daten zu den Corona Lockdowns verwenden konnten, die genauer auf die Öffnungszeiten örtlicher Geschäfte eingegangen sind.

### 3.5 Open-Street-Map-Daten

Im vorherigen Abschnitt wurden die Datenquellen zu stadspezifischen Daten erläutert. Variablen zur Demographie oder dem Autobesitz nach Städten sind wertvoll, um die Unterschiede im Radverkehr zwischen den Städten einzuschätzen zu können. Dies hilft jedoch nicht bei der räumlichen Unterscheidung. Verwendet man keine räumlichen Merkmale, so erhält man eine einheitliche Vorhersage je Stadt. Die räumliche Verteilung des Radverkehrs innerhalb jeder Stadt ließe sich so nicht abbilden. Die einzige räumliche Variable, die bisher erhoben wurde, ist die Entfernung zum Stadtzentrum, beruhend auf den Daten des Gemeindeverzeichnis von 2012 bis 2021. Doch selbst mit dieser Variablen gingen Abweichungen durch lokale Anlaufpunkte wie bei Bahnhöfen verloren.

Der daraus resultierende Schluss ist, Koordinaten zu solchen Anlaufpunkten oder auch Points of Interest (POIs) dazu zu nutzen, Entfernungen zu den Radzählstationen zu berechnen und die Anzahl nach Radius und Entfernung zum nächstgelegenen POI mit in das Modell aufzunehmen. Die beste Datenquelle hierfür bietet Open-Street-Map (OMS). Wie schon im Kapitel 2.2.4 beschrieben, handelt es sich bei Open-Street-Map um ein internationales Projekt, das in gemeinsamer Freiwilligenarbeit geographische Daten sammelt und öffentlich zur Verfügung stellt.

*Listing 3.1: OSM Daten Abfrage*

```

1 q <- getbb(toString(rawData$Town[1])) %>%
2 opq() %>%
3 add_osm_feature("amenity", "cinema")
4
5 cinema <- osmdata_sf(q)

```

Der Zugriff auf die OSM-Daten funktioniert mit dem R-Paket „osmdata“ von Padgham et al. (2017). Dieses Paket bietet die Möglichkeit Daten verschiedener OSM-Kategorien abzufragen, wie der Quellcode 3.1 zeigt. Dabei bezeichnet rawData\$Town[1] den ersten Eintrag in der Variable Town für

den Grunddatensatz rawData. Da jede Stadt ihr eigenes Skript hat, ist dies jeweils die relevante Stadt. Für diese Stadt wird eine Anfrage gestellt, die bestimmte Features erfüllen muss. Im Code Beispiel werden Kinos gesucht. Mithilfe des R-Pakets „sf“ von Pebesma (2018) werden die Daten der Anfrage in ein Simple-Feature-Format gespeichert in cinema. Dieses Format beinhaltet Koordinatenpunkten, mit jeweiliger ID und teils Namen und Adressen der verschiedenen Einrichtungen. Darüber hinaus können aber auch Koordinaten der Linien und Grundflächen ausgelesen werden, falls es sich z.B. um Straßennetze handelt, oder um Koordinaten der jeweiligen Polygone der Gebäudegrundrisse.

### 3.5.1 Ausgewählte POIs

Dieser öffentliche Zugang zu kartographischen Daten beinhaltet nicht nur Straßenverläufe, sondern auch Lageparameter zu öffentlichen Einrichtungen, Geschäften, Verkehrsknotenpunkten von Bus und Bahn, Ampeln und Straßenübergängen und vielem mehr. Es gibt also eine große Auswahl an Kategorien, die sich in das Modell aufnehmen ließen. Die Auswahl begrenzte sich dabei auf Kinos, Schulen, Universitätsgebäude, Supermärkte, Kleidungsgeschäfte und Fahrradwerkstätte. Die Auswahl dessen fand rein intuitiv statt. Schüler und Studenten sind häufiger Fahrradfahrer als der Rest der Bevölkerung. Kleidungsgeschäfte sind oft räumlich stark konzentriert im Stadtzentrum und bilden hierfür einen guten Indikator. Hat eine Stadt ein hohes Angebot an Fahrradwerkstätten, ist dies ein guter Hinweis darauf, dass hier ein hoher Radverkehr zu finden ist. Die Auswahl von Kinos und Supermärkten ist rein zufällig. Auch unterliegen etwaige kausale oder korrelative Zusammenhänge dieser Variablen der persönlichen Mutmaßung, die sich entweder im Verlauf des Projektes bestätigen, oder aber auch nicht.

Noch empfehlenswerter wäre es, den Datensatz um noch mehr Kategorien zu erweitern und genau zu evaluieren, welche Kategorien die Performance des Modells erhöhen. Jedoch hätte dies den Arbeitsumfang dieser Masterarbeit nochmals drastisch erhöht. In Abwägung des zeitlichen Aufwands und dem Nutzen für das Modell, beschränkt sich deswegen der Datensatz auf die er-

wählten POIs.

In jeder Kategorie wurde jeweils die Entfernung der jeweiligen Radzählstation zum jeweiligen POI berechnet. Darüber hinaus wurden in jeder Kategorie die Anzahl an Einrichtungen in zwei verschiedenen Radien berechnet. Für Kinos und Fahrradwerkstätten wurde ein ein 1-Kilometer und ein 2-Kilometer Radius gewählt. Für Schulen, Universitätsgebäude und Kleidungsgeschäfte wurde ein 500-Meter und ein 2-Kilometer Radius gewählt. Für Supermärkte wurde ein 500-Meter und ein 1-Kilometer Radius genutzt. Die Auswahl dieser Berechnungsgrenzen sind wiederum rein zufällig. Eine genaue Evaluation dieser Auswahl fand nicht statt. Zu Universitätsgebäuden, Supermärkten und Kleidungsgeschäften finden sich die Abbildungen 7.3, 7.4 und 7.5 im Anhang. Für Städte, die über keine Universität verfügen, wurde angenommen, dass die nächste Universität mindestens 50 km weit entfernt sei.

*Listing 3.2:* Speichere die OSM Koordinaten

```

1  cinmat=matrix ( 1:3 *length( cinema$osm_polygons$osm_id ) ,
2                  nrow = length( cinema$osm_polygons$osm_id ) ,
3                  ncol = 3 )
4
5  for ( i  in  1:length( cinema$osm_points$name )) {
6      cinmat [ i ,1]=cinema$osm_polygons$osm_id [ i ]
7      cinmat [ i ,2]=as . data . frame (
8          cinema$osm_polygons$geometry [[ i ]][ 1 ]) [ 1 ,1 ]
9      cinmat [ i ,3]=as . data . frame (
10         cinema$osm_polygons$geometry [[ i ]][ 1 ]) [ 1 ,2 ]
11
12 }
13 }
```

Der Quellcode 3.2 zeigt die Umwandlung der Daten. Von Interesse sind nur die ersten Koordinatenpunkte der jeweiligen Polygone der Kinogebäude. Dazu wird die Matrix „cinmat“ erstellt, die 3 Spalten hat und deren Reihenanzahl der Anzahl an ID's in `length(cinema$osm_polygons$osm_id)` entspricht, damit in diesem Fall alle Kinos berücksichtigt werden. Mittels

eines `for`-Loops werden alle Daten eingespeichert, so der Längen- und Breitengrad in `cinema$osm_polygons$geometry[[i]][1])[1,]` und die jeweilige ID in `cinema$osm_polygons$osm_id[i]`. Diese Matrix wird danach in ein Dataframe umgewandelt, um damit einfacher arbeiten zu können.

Nun muss nur noch für jede einzelne Zählstation die jeweilige Distanz zu den verschiedenen Kinos berechnet werden. Dazu bietet das R-Paket „geosphere“ von Hijmans (2021) die Funktion `distm`, mit deren Hilfe man die Entfernung zwischen zwei Koordination in Metern berechnen lassen kann. Pro Zählstation wird der `For`-Loop im Quellcode 3.3 ausgeführt. Im Code sind zwei Indexnummern zu finden, dabei steht `i` für die jeweilige Zählstation und `j` für das jeweilige Kino. Im Dataframe `d` sind die Koordinaten der jeweiligen Zählstationen gespeichert. Zusammen mit den Koordinaten der Kinos im `cinmat`-Datenframe kann die Funktion `distm` die Entfernung beider Koordinaten zueinander berechnen. Diese Entfernung wird dann im Vector `dists` gespeichert.

*Listing 3.3:* Berechnung der Entfernung

```

1 for (j in 1:length(cinmat$id)) {
2   cindist=distm(c(d$Lon[i],d$Lat[i]),
3                 c(cinmat$lon[j],cinmat$lat[j]),
4                 fun=distGeo)
5   distc[j]=cindist
6 }
```

Nun ist ein Vector vorhanden, der alle Entfernungen gespeichert hat. Dieser kann nun dazu genutzt werden, die benötigten Variablen zu berechnen. Zusammen mit dem Namen der aktuellen Station, der im Subset Datensatz `d[1,1]` zu finden ist, wird die jeweilige Variable gespeichert. Dabei lässt sich mit der Funktion `min(distc)` die Entfernung zum nächstgelegenen Kino finden. Die Anzahl an Kinos in einem 1 km Radius findet sich mit der Funktion `sum(distc < 1000)`.

*Listing 3.4:* Berechnung der Entfernungsvariablen

```

1 distmat_closest[i,1]=d[1,1]
2 distmat_closest[i,2]=min(distc)
3
4 distmat_1kmradius[i,1]=d[1,1]
5 distmat_1kmradius[i,2]=sum(distc < 1000)

```

Die so berechneten Variablen lassen sich einfach verbinden mit dem Befehl `merge`.

*Listing 3.5:* Füge neue Variablen dem Datensatz hinzu

```

1 rawData = merge(x = rawData, y = distmat_closest,
2 by = c("Station"),
3 all = FALSE)
4
5 rawData = merge(x = rawData, y = distmat_1kmradius,
6 by = c("Station"),
7 all = FALSE)

```

### 3.5.2 Ausgestaltung des öffentlichen Verkehrs

Neben POIs, die als Anlaufstellen des Radverkehrs dienen, ist der öffentliche Nahverkehr ein weiterer Faktor, der den Radverkehr beeinflussen sollte mit verschiedenen Effekten. Anzunehmen ist, dass ein breites Angebot im öffentlichen Nahverkehr substitutionel auf den Radverkehr wirkt. Das heißt, dass viele Pendler auf Straßen- oder U-Bahnen ausweichen, wenn diese flächen-deckend angeboten werden und dadurch weniger Pendler mit dem Fahrrad fahren. Dann handelt es sich also dabei um entgegengesetzte Effekte. Eine weitere interessante Variable hierfür wären auch Preistarife des Nahverkehrs, die aber nicht im Datensatz dieser Masterarbeit enthalten sind. Stattdessen zählt der Datensatz die Dichte von Stationen des öffentlichen Nahverkehrs. Darüber hinaus gibt es auch komplementäre Effekte vor allem zwischen Bahnhöfen und Fahrrädern, da an Bahnhöfen der Nahverkehr vom Fahrrad zum Fernverkehr mit der Bahn wechselt. Hier ergänzen sich beide Verkehrsfor-

men.

Im Datensatz vertreten sind Busstationen, Straßenbahnstationen, U-Bahn Stationen und Bahnhöfe. Der Datensatz enthält jedoch nicht nur Informationen über den öffentlichen Nahverkehr. Auch Informationen zum Ausbau und zur Gestaltung des Straßennetzes sind enthalten, denn schon Heinen et al. (2010) haben aufgezeigt, dass die bauliche Substanz z.B. in Form von Ampeln und von Verkehrsschildern einen Einfluss haben und Vandenbulcke et al. (2014) zeigen, dass die komplizierte Gestaltung von Kreuzungen einen Einfluss auf Radverkehrsunfälle haben. Deswegen sind Daten zu Ampeln und Straßenübergängen ohne Ampeln im Datensatz vorhanden. Zu allen Daten ist wieder die nächste Entfernung angegeben, sowie die jeweilige Anzahl in zwei verschiedenen Radien. Für Ampeln, Straßenübergänge ohne Ampeln, Busstationen, Straßenbahnstationen und U-Bahnstationen betragen die Radien jeweils 250 Meter und 1 Kilometer. Für Bahnhöfe betragen die Radien 1 und 3 Kilometer. Bei den Bahnhöfen wurden nur diejenigen ausgewählt, deren Betreiber die DB-Netz-AG ist. Im Anhang finden sich zu Busstationen, Ampeln, Straßenbahnstationen und Bahnhöfen die Abbildungen 7.6, 7.7, 7.8 und 7.9. Für Städte, die nicht über Straßenbahnen oder U-Bahnen verfügen, wurde angenommen, dass die nächsten Stationen jeweils mindestens 50 Kilometer weit entfernt seien.

Im Quellcode hier besteht kein großer Unterschied zum Quellcode 3.2, außer dass nun einzelne OSM-Punkte relevant sind und keine Polygone z.B. für die Ampeln.

### 3.5.3 Straßentypen

Neben Informationen zu Gebäuden, Einrichtungen und einzelnen Punkten, wie Ampeln, bietet Open-Street-Map auch Daten zum Straßennetz. Der Datensatz, der zur Berechnung der Vorhersagen genutzt werden soll, sollte Informationen zum jeweiligen Straßennetz berücksichtigen. Primär gehört dazu die Information des Straßentyps, wobei es zahlreiche Kategorien gibt. Eine begrenzte Auswahl an Straßentypen wird im ersten Datensatz berücksichtigt. Dazu zählen Radwege, Pfade, Wohngebietstraßen, Straßen in einem

verkehrsberuhigten Bereich oder auch Spielstraßen, sekundäre und primäre Hauptstraßen. Die Aufteilung von Straßentypen nach Stationen im Datensatz sieht man auch in der Abbildung 3.10a, wobei festzustellen ist, dass einige Stationen doppelt zugeteilt worden sind, durch sich kreuzende Straßen und Brücken. Auch Brücken wurden im Datensatz berücksichtigt und getestet, wie weit die nächstgelegene Brücke zur Zählstation entfernt ist. Eine Darstellung des Zusammenhangs zwischen dem Radverkehr und Brücken findet sich im Anhang in der Abbildung 7.10.

Ob eine Zählstation zu einem Straßentyp gehört oder nicht, wurde getestet, indem berechnet wurde, inwieweit eine Zählstation von einer Straße des betroffenen Straßentyp entfernt ist. Ist diese Entfernung geringer als 5 Meter, wurde angenommen, dass die Zählstation sich auf oder an einer solchen Straße befindet.

*Listing 3.6: Teste den Straßentyp*

```

1 for(i in 1:nlevels(as.factor(rawData$Station))){  

2   dist_mat$cycleways[i] =  

3     min(st_distance(DT2$geometry,  

4       DT3cycleways))  

5  

6   if(dist_mat$cycleways[i]<5){  

7     bool_mat$cycleways[i]=1  

8   }  

9 }
```

Dazu muss die jeweils kürzeste Entfernung zwischen einem Punkt und einer Linie berechnet werden. Dies funktioniert mithilfe der **st\_distance** Funktion. Diese Funktion benutzt die Liste aller Koordinaten aller Straßenabschnitte des jeweiligen Straßentyps DT3cycleways und der Koordinate der Zählstation DT2\$geometry, um daraus die Entfernung eines jeden Straßenabschnitts zum Punkt der Zählstation zu berechnen. Davon ist allein das Minimum interessant. Ist das Minimum geringer als 5 Meter, ist klar, dass die Zählstation nahe des jeweiligen Straßentyps liegt.

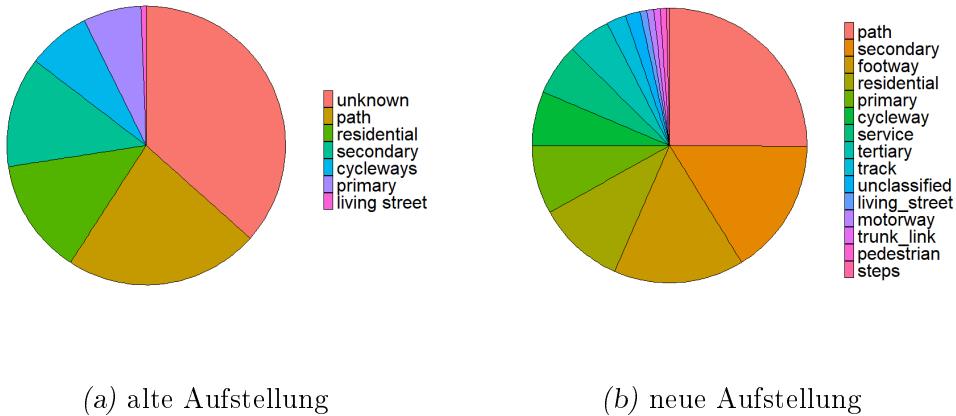


Fig. 3.10: Verteilung von Straßentypen im Datensatz nach der Version

Im zweiten Datensatz, der erstmals auch Corona Daten inkludiert, wurde dieses Problem anders gelöst, dargestellt in der Abbildung 3.11. In einem For-Loop wurden alle Zählstationen betrachtet. Dabei wurden alle Straßen, die ein kleines Viereck (in der Abbildung 3.11 rot) um die Zählstation (blauer Punkt) herum berührten, betrachtet. Die Zählstation wurde jeweils der nächstliegenden Straße (blaue Linie) zu geordnet. Das in der Abbildung gewählte Beispiel stammte von einer Zählstation in Hamburg. Auf diese Weise können aber nicht nur Daten zum Straßentyp festgestellt werden, sondern auch zur Straßenlänge, zum Straßenbelag, zur geltenden Höchstgeschwindigkeit und die Anzahl der Straßenspuren.

Genauer wird dieser Vorgang auch im Quellcode 3.7 beschrieben. In jedem Loop-Durchgang wird der Datensatz gefiltert nach Zählstationen und die Beobachtungen einer Station landen im Dataframe d. Für jede Station muss das rote Rechteck wie in der Abbildung 3.11 berechnet werden. Dies geschieht wie im Quellcode 3.7 mittels der Variable radius und den Koordinaten aus d, also den Koordinaten der jeweiligen Zählstation. Die Koordinaten des Vierecks sind so in myLocation gespeichert.

Listing 3.7: Berechnung der Straßenvariablen

```
1 d=BikeData[BikeData$Station %in%
2   toString(levels(as.factor(
3     BikeData$Station)))[i]],]
4
5 radius = 0.0012
6 myLocation <- c(d$Lon[1]-radius,d$Lat[1]-radius/1.8,
7   d$Lon[1]+radius,d$Lat[1]+radius/1.8)
```

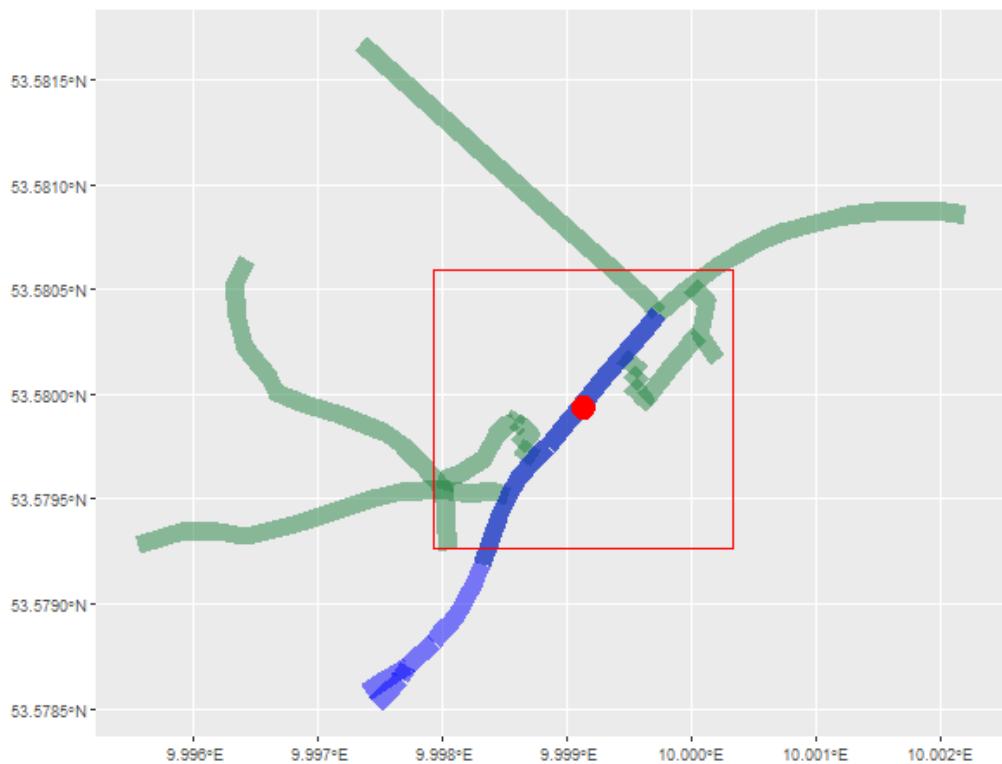


Fig. 3.11: Beispiel der Straßentypanalyse

Mit diesen Koordinaten kann nun eine Abfrage von Open-Street-Map-Daten zu der Klasse „highway“ gestellt werden. Wie eine solche Abfrage gestellt wird, ist auch im Quellcode 3.1 zu sehen. Darauf hin muss die Zählstation der nächstgelegenen Straße zugeordnet werden. Dies passiert im Quellcode 3.8. Dieser besteht aus zwei kürzeren For-Schleifen. Die erste wandelt alle Koordinaten der Straßen in ein passendes GPS-Format um und ermittelt die jeweils kürzeste Distanz der Straße zu den Koordinaten der Zählstation, welche gespeichert sind in der Variable `count_point$geometry`. Für den Fall, dass mehrere Straßen auf der selben Position liegen, ermittelt die Funktion `which(dist==min(dist))`, welche dieser Straßen dies sind und wählt von diesen dann diejenige aus, bei der ein Straßename angegeben ist.

*Listing 3.8:* Zuordnung zur nächsten Straße

```

1 j=1
2 for (j in 1:nrow(streets$osm_lines)){
3   street_Points = st_transform(
4     streets$osm_lines$geometry[j], 4269)
5   dist[j] = min(st_distance(
6     count_point$geometry, street_Points))
7 }
8
9 for (j in which(dist==min(dist))){
10   if (length(streets$osm_lines$name[j])>0){
11     if (!is.na(streets$osm_lines$name[j])){
12       nearest=j
13     }
14   }
15 }
```

Die Variable `nearest` gibt nun an, welche Straße die nächstgelegene zur Zählstation ist und mit diesen Informationen lassen sich die weiteren notwendigen Variablen berechnen, wie auch im Quellcode 3.9 beschrieben wird. Wie sich die Straßentypen unter den Zählstationen verteilen, zeigt die Abbildung 3.10b. Weiter zeigt die Abbildung 3.12 die Aufteilung des Radver-

kehrsaufkommens nach der jeweiligen Straßenbeschaffenheit, so nach Straßenbelag und Straßentyp, wobei die Beschreibungen dem englischen Original der Open-Street-Map Beschreibung entspricht, denen spezifische Definitionen zu Grunde liegen. Die wichtigsten Straßen im Straßennetz entsprechen der Klasse „trunk“, dann folgen Straßen der Klasse „primary“, „secondary“ und „tertiary“. Diese drei Klassen beschreiben oft die verschiedenen Hauptstraßen einer Stadt. Ein Beispiel für Berlin, dass nur diese drei Straßenarten zeigt, ist in Abbildung 7.11b im Anhang zu sehen. Die Klassen „cycleway“, „footway“, „pedestrian“ und „residential“ entsprechen in ihrer Bedeutung jeweils der Übersetzung. Die Klasse „livingstreet“ kann in Deutschland am ehesten als Spielstraße umschrieben werden. Kompliziert ist die Verwendung der Klasse „service“, die eigentlich Zufahrten beschreiben soll, die rein geschäftlichen Zwecken dienen. Jedoch werden auch einige Fahrradwege so bezeichnet. Diese Missklassifikation wird in der graphischen Darstellung noch für Probleme sorgen, wie der Abschnitt zur Modellprojektion zeigen wird. Bei „tracks“ und „path“ handelt es sich meistens um unbefestigte Straßen, wobei zweitens meistens von Fußgängern und Radfahrern genutzt wird.

*Listing 3.9:* Berechnung der Straßenvariablen

```

1 street_type = streets %>% st_is_highway %>% st_nearest()
2 surface = streets %>% st_is_surface %>% st_nearest()
3 lanes = streets %>% st_is_lanes %>% st_nearest()
4 maxspeed = streets %>% st_is_maxspeed %>% st_nearest()
5 streetlengths = st_length(street_Points %>% st_nearest())

```

### 3.6 Sonstige Variablen

Weitere wichtige Variablen sind Feier- und Ferientage. Dazu wurde eine Liste dieser Tage händisch für alle Bundesländer angelegt und diese automatisch in den Datensatz übertragen. Informationen zu den Ferien- und Feiertagen seit 2012 nach den jeweiligen Bundesländern stammen aus dem Internet<sup>4</sup>.

---

<sup>4</sup> [www.kalenderpedia.de](http://www.kalenderpedia.de), letzter Zugriff 17.2.2023.

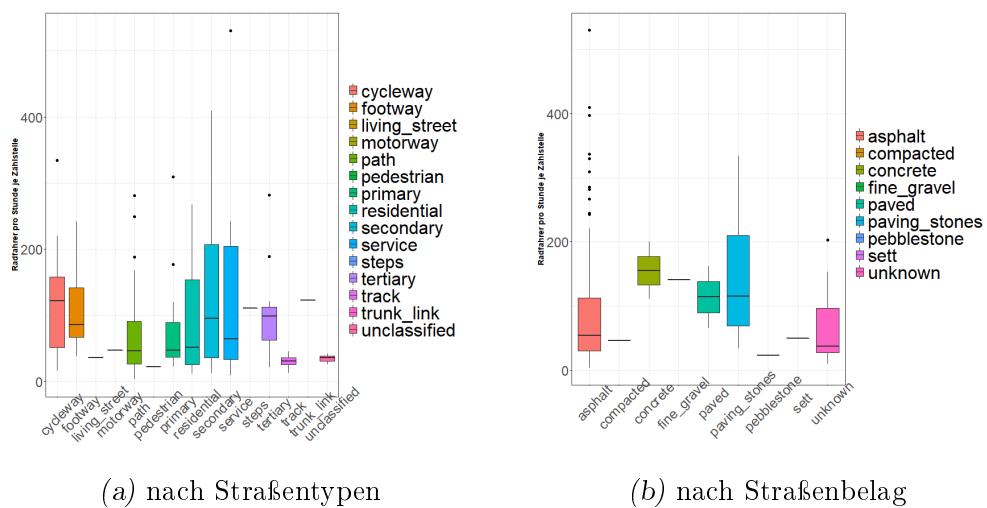


Fig. 3.12: Radverkehr nach Straßenbeschaffenheit

## 4. VERWENDETE METHODEN

Das erste Kapitel gab einen genauen Überblick über die bestehende Literatur, die sich mit der Vorhersage des Aufkommens von Fahrrädern in urbanen Zentren beschäftigt. Immer häufiger wurden dabei Machine-Learning-Methoden zur Schätzung eines Modells verwendet. Um die Übersichtlichkeit der Arbeit zu gewährleisten, wurden Erläuterungen über die verschiedenen verwendeten Methoden bis zu diesem Zeitpunkt aufgespart, um einen gegliederten Überblick zu ermöglichen.

Im Folgendem wird die Funktionsweise von Regressionssystemen, Support Vector Systemen, Entscheidungsbaum Varianten und neuronalen Netzwerken erklärt. Die Auswahl dieser Schwerpunkte beruht auf der Verwendung in der bisher dargestellten Literatur. Hat man ein geeignetes Modell ausgewählt, lässt sich dessen Effizienz mit dem richtigen Validierungsverfahren noch weiter steigern. Deswegen beinhaltet dieses Kapitel ebenfalls eine kurze Erläuterung der Cross Validation.

### 4.1 Probleme zur Autokorrelation

Vorab muss das Problem der Autokorrelation erwähnt werden. Autokorrelation bezeichnet die Korrelation innerhalb einer Variable und nicht wie sonst die Korrelation zweier verschiedener Variablen. Diese Autokorrelation kann räumlich und zeitlich auftreten. Bei der zeitlichen Autokorrelation ähneln sich Werte, die zeitlich nah bei einander liegen. Bei der räumlichen Autokorrelation ähneln sich Werte, die räumlich nah bei einander liegen.

Dies kann zu Problemen führen. So beschreiben Liu et al. (2022), dass räumliche Autokorrelation die Annahme von unabhängig und identisch verteilten Zufallsvariablen verletzt. Dies kann zu Overfitting oder einem Bias der

Vorhersagen führen. Im Fall von OLS-Regressionen ist der Standardfehler nicht mehr konsistent, was Aussagen über die Signifikanz des Modells wertlos macht. So beschreiben es auch Stock und Watson (2015a). Da das Ziel dieser Masterarbeit jedoch nur die Vorhersage ist, ist die Signifikanz nicht entscheidend. Um gute Vorhersagen zu treffen, ist es jedoch notwendig, das Risiko von Overfitting oder einem Bias zu minimieren. Dazu gibt es drei Lösungsansätze. Diese wären die Verwendung von Lagged Variablen, Feature Engineering und Resampling.

#### 4.1.1 Lagged Variablen

Der Begriff Lagged Variable bezeichnet die Aufnahme einer versetzten Beobachtung, die als zusätzliche Variable mit in das Modell aufgenommen wird. Im Bereich von Zeitreihendaten wird also eine zeitlich vorhergehende Beobachtung der eigentlich abhängigen Variable in das Modell aufgenommen. Im Fall der Regression nennt man ein solches Vorgehen Autoregression. Dies beschreiben z.B. Stock und Watson (2015b). Die mathematische Formulierung einer allgemeinen Autoregression lautet:

$$y_t = \beta_0 + \beta_1 * y_{t-1} + \beta_2 * y_{t-2} + \dots + \beta_p * y_{t-p} + u_t \quad (4.1)$$

Diese Formulierung kann natürlich nicht nur in OLS-Regressionen angewendet werden, sondern auch bei allen weiteren Machine-Learning-Algorithmen, über die hier noch geschrieben wird. Eine Weiterentwicklung dieses Ansatzes ist die ARIMA-Modellierung, was die Abkürzung für Auto-Regressive-Moving-Average ist.

Problem an diesem Ansatz ist, dass er für die Vorhersage von unbekannten Out-Of-Sample Orten nicht anwendbar ist, weil an unbekannten Orten auch vorhergehende Beobachtungen unbekannt sind, die man in das Modell zur Vorhersage mit aufnehmen müsste. Deswegen ist die Aufnahme von Lagged Variablen nicht geeignet für diese Masterarbeit.

### 4.1.2 Erklärende Variablen

Eine weitere Möglichkeit, Autokorrelation zu vermeiden, ist es, Variablen in das Modell aufzunehmen, die die Autokorrelation erklären können. Nimmt man in einem zeitlichen Modell z.B. die Stunden als Variable auf, die erklären können, dass in der Mittagszeit viele Fahrradfahrer auf den Straßen sind und Nachts nur sehr wenige, dann erklärt das auch, warum zu zwei Zeitpunkten, die nah beieinander liegen z.B. um die Mittagszeit herum, ähnlich hohe Werte vorhergesagt werden. Dies funktioniert auch für die räumliche Autokorrelation. Nimmt man als Variable die Entfernung zum Stadtzentrum auf, dann erklärt dies nicht nur, dass nahe des Stadtzentrums viele Radfahrer unterwegs sind und dafür weniger Radfahrer außerhalb des Stadtzentrums, es erklärt auch, wieso sich zwei nahe beieinander liegende Punkte ähneln, weil beide z.B. weit weg vom Stadtzentrum liegen.

Diesen Ansatz verfolgt auch die Studie von Liu et al. (2022), automatisiert jedoch diesen Prozess. Sie schlagen für ein Random-Forest-Modell z.B. vor, Probleme mit Autokorrelation zu vermeiden, in dem gewichtete räumliche Variablen durch das LASSO-Regressionsverfahren selektiert werden. Diese räumliche Variablen übernehmen die vorher besprochene Rolle. Die LASSO-Regression ist neben der RIDGE-Regression eines von zwei Regularisierungsverfahren. Dabei wird eine Regression aufgesetzt, in deren Optimierungsproblem nicht nur die quadrierte Summe der Fehlerterme steht, sondern dazu addiert die quadrierte (RIDGE) oder absolute (LASSO) Summe der Estimatoren selbst. Ziel einer solchen Optimierung ist es, die Estimatoren zu schrumpfen, wobei LASSO Estimatoren auf null setzen kann und somit Feature Selection betreiben kann.

Ein solches Modell aufzusetzen, übersteigt jedoch den Arbeitsumfang dieser Arbeit. Was die zeitliche Autokorrelation angeht, können die Variablen zur Stunde und zum jeweiligen Monat Aufschluss geben. Was die räumliche Autokorrelation angeht, stehen so viele Variablen zur Auswahl, dass es hier durchaus Sinn machen könnte, eine genauere Feature Selektion zu betreiben. Ein solches Verfahren jedoch erfordert leistungsfähigere Hardware, würde aber gleichzeitig die Performance des Modells nicht drastisch erhöhen.

#### 4.1.3 Resampling

Resampling bezeichnet eine Neuanordnung von Beobachtung eines Datensatzes, sodass dabei ein neuer Datensatz entsteht. Beispiele hierfür ist das Bootstrapping, das uns auch bei der Cross Validation begegnen wird. Bei Bootstrapping werden Mehrfachziehungen zufällig ausgewählter Beobachtungen vorgenommen, sodass ein neuer Datensatz entsteht. Daneben ist Resampling auch ohne Mehrfachziehung von Beobachtungen möglich. Diese führt dazu, dass man weniger Beobachtungsdaten zieht, als im Datensatz vorhanden sind. Dabei handelt es sich um Undersampling.

Undersampling kann auch dabei helfen, Autokorrelation zu vermeiden, denn verringert man die Dichte an Beobachtungen, spielt die zeitliche oder räumliche Nähe einzelner Beobachtungen zu einander eine geringere Rolle. Der große Nachteil dieser Methode ist, dass durch ein Verzicht dieser Daten auch Informationen verloren gehen und das Modell durch die kleinere Sample Größe eine schlechtere Performance vorweisen kann. Im Fall dieser Arbeit steht ein Datensatz zur Verfügung, der mehr als groß genug ist, um Undersampling zuzulassen. Zudem kommt hinzu, dass der technische Flaschenhals im Arbeitsspeicher der verwendeten Hardware es notwendig machen wird, die Sample Größe des Datensatzes ohnehin zu verkleinern.

## 4.2 OLS-Regression

Einer der einfachsten und geläufigsten Methoden ist die Regressionsanalyse. So nutzen Holmgren et al. (2017), Alattar et al. (2021) und Gao und Chen (2022) z.B. ein lineares Regressionssystem und Wessel (2020) ein log-lineares sowie ein negativ binomiales Regressionssystem.

Das Prinzip einer einfachen OLS (Ordinary-Least-Square) Regression besteht darin, den Zusammenhang zwischen zwei Variablen zu finden, der die Summe der quadrierten Fehlerterme, also die Abweichung tatsächlicher Beobachtungen zur Regressionsgerade, minimiert. Als Maß zur Validierung eines solchen Modells ließe sich z.B. das Bestimmtheitsmaß  $R^2$  nutzen, also der Anteil der Streuung, der durch die Regression erklärt werden kann, aber auch z.B. die

Wurzel der summierten quadratischen Fehler (RMSE). Ein solches Fehlermaß oder Bestimmtheitsmaß, dass die Performance des Modells bewertet ist nützlich, um einen Vergleich zu ziehen, verwendet man mehrere Modelle, wie es z.B. Holmgren et al. (2017), Broucke et al. (2019) oder Gao und Chen (2022) machen.

### 4.3 Support-Vector-Regression

In der bisher betrachteten Literatur haben Holmgren et al. (2017), Broucke et al. (2019), Xu et al. (2013) und Gao und Chen (2022) Support-Vector-Machines bzw Support-Vector-Regressionen verwendet. Pisner und Schnyer (2020) schildern Support-Vector-Machines als eine Supervised-Classification-Methode, die eine Hyperebene nutzt, um Daten nach unterschiedlichen Ausprägungsmustern zu trennen und so den jeweiligen Klassen zu zuordnen. Diese Vorgehensweise ist auch dargestellt in Abbildung 4.1. Nun könnte diese Trennlinie, auch Hyperplane genannt, so gewählt werden, dass sie das Margin, also den Abstand zwischen der Trennlinie selbst und der nächstgelegenen Beobachtung, maximiert. Dieses Vorgehen wird auch als Maximum-Margin-Classification bezeichnet. Jedoch falls innerhalb des maximalen Margin-Bereichs eine neue zufällige Beobachtung hinzu kommt, dann führt diese zu einer starken Veränderung der Trennlinie selbst. Ausreißerdaten können so zu einer starken Verzerrung führen und eine Generalisierbarkeit wäre nicht gegeben. Deswegen nutzt man oft Soft-Margins, die die Missklassifikation von Ausreißern zu lassen. Dabei stellt  $\xi$  die Variable für die Toleranz von Missklassifikationen dar, auch oft Slack Variable genannt. Ist  $\xi = 0$  so erhalten wir einen Hard-Margin-Classifier.

Mathematisch formulieren James et al. (2013a) eine Hyperplane in einem  $p$ -dimensionalen Raum wie folgt:

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p = 0 \quad (4.2)$$

Dabei liegt der Normalenvektor  $\beta = (\beta_1, \beta_2, \dots, \beta_p)$  in der orthogonalen Richtung zur Hyperplane. Das Optimierungsproblem hinter der Soft-Margin ( $M$ )

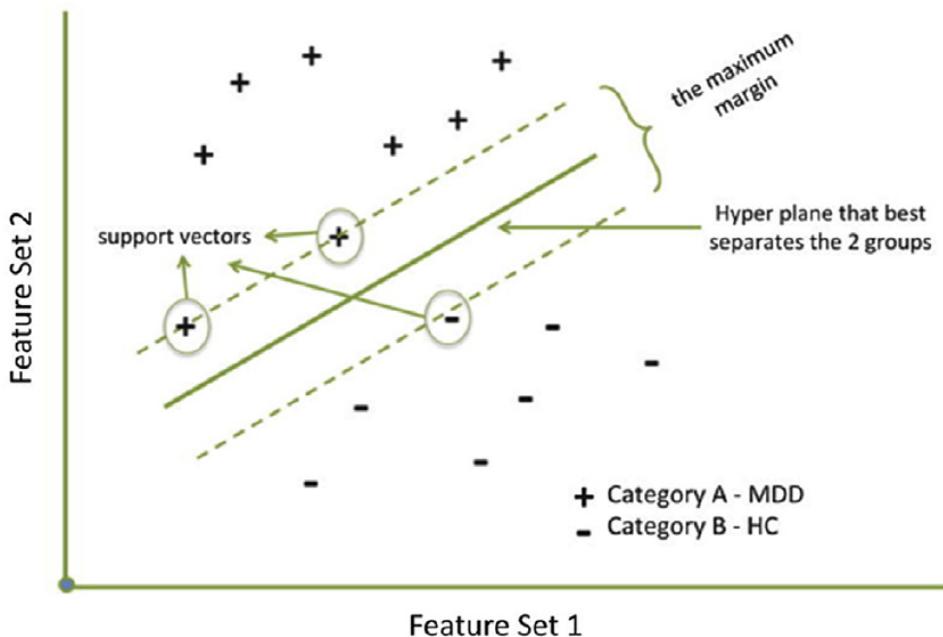


Fig. 4.1: Hyperebenen (Hyperplanes) in Support-Vector-Machines

Quelle: Pisner und Schnyer (2020)

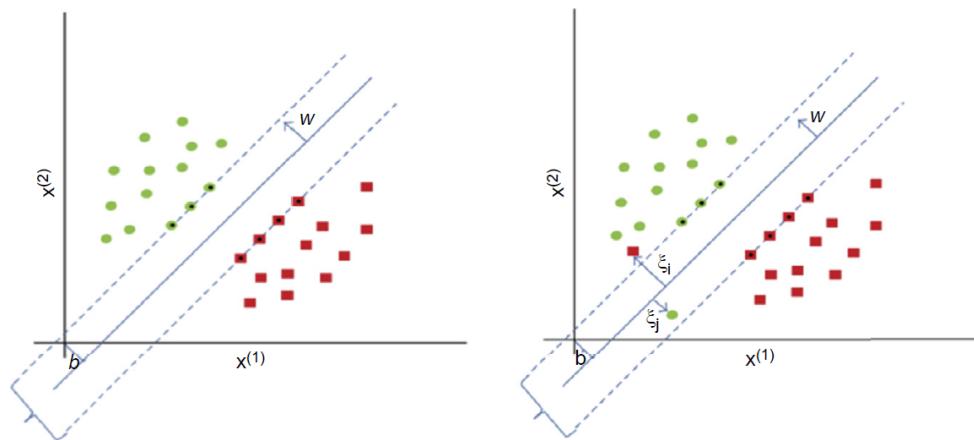


Fig. 4.2: Soft Margins in Support-Vector-Machines

Quelle: Pisner und Schnyer (2020)

Classification sieht wie folgt aus:

$$\begin{aligned}
 & \max_{\beta_0, \beta_1, \dots, \beta_p, M, \xi_1, \dots, \xi_n} M \\
 & \text{und} \sum_{j=0}^p \beta_j^2 = 1, \\
 & y_i(\beta_0 + \beta_1 x_{i1} + \dots + \beta_p x_{ip}) \geq M(1 - \xi_i), \\
 & \xi_i \geq 0, \sum_{i=0}^n \xi_i \leq D,
 \end{aligned} \tag{4.3}$$

Hier taucht nun die Slack Variable auf, die einen Toleranzbereich für Missklassifikation ermöglicht.  $D$  ist ein nichtlinearer Tuning-Parameter. Natürlich kann man in der Funktion des Hyperplanes (4.2) Polynomiale aufnehmen und so einen nicht linearen Classifier erhalten.

Grundlegendes Problem ist nun, dass in der Vorhersage des innerstädtischen Fahrradaufkommens kein Klassifikationsproblem besteht, so wie es eine Support-Vector-Machine lösen würde. Deswegen bietet es sich an, eine Support-Vector-Regression zu nutzen, so wie es z.B. Holmgren et al. (2017) machen. Während die herkömmliche OLS-Regression die summierten quadratischen Fehlerterme minimiert, minimiert die Support-Vector-Regression, so beschreiben es Awad und Khanna (2015), eine Loss-Funktion, die Fehlvorhersagen bestraft. Dieser Vorgang ist auch eine Verallgemeinerung des bisher beschriebenen Klassifikationsalgorithmus.

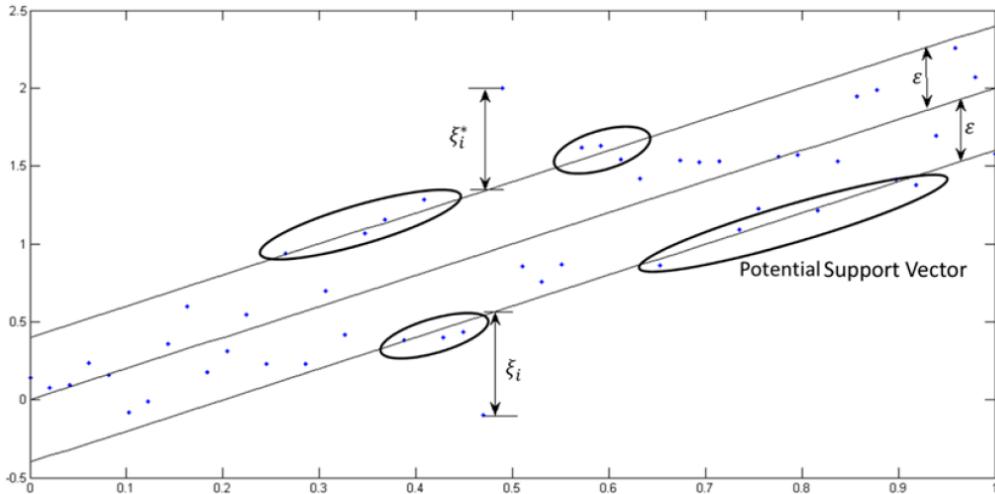


Fig. 4.3: Eindimensionale lineare Support-Vector-Regression

**Quelle:** Awad und Khanna (2015)

Wo im Soft-Margin-Classifier  $\xi$  genutzt worden ist, um auch Missklassifikationen zuzulassen, wird nun ein unempfindlicher Bereich von  $\varepsilon$  um die Hyperplane herum gelegt, sodass ein Großteil der Beobachtungen im eigentlichen Margin Bereich zu finden sind. Aus der Hyperplane, die eigentlich Beobachtungen trennen soll, wird so die regressive  $\varepsilon$ -Tube, eine wertbeständige

Funktion. Dargestellt ist dies auch in der Abbildung 4.3.

Die Slack Variable  $\xi$  wird so zur Toleranzvariable für Ausreißer von der  $\varepsilon$ -Tube und so zum Bestandteil der Loss-Function, die man versucht zu minimieren. Zusammen mit C der Gewichtung der Minimierung ergibt das:

$$\begin{aligned} \min_{\beta_0, \beta_1, \dots, \beta_p, M, \xi_1, \dots, \xi_n} & \frac{1}{2} \|\beta\|^2 + C \sum_{i=1}^N \xi_i + \xi_i^* \\ \text{und } & y_i - \beta^T x_i \leq \varepsilon + \xi_i^*, \quad i = 1 \dots N \\ & \beta^T x_i - y_i \leq \varepsilon + \xi_i \quad i = 1 \dots N \\ & \xi_i, \xi_i^* \geq 0, \quad i = 1 \dots N \end{aligned} \tag{4.4}$$

Der Vorteil dieser Vorgehensweise ist, dass man durch den Optimierungsprozess ein Modell erhält, das sehr robust auf Ausreißer reagiert trotz hoher Präzession durch die Nutzung von  $\xi$ . Denn die OLS-Regression im Vergleich beachtet bei der Ausrichtung ihrer Regressionslinie alle Beobachtungen auch die Ausreißer, dadurch dass die OLS-Regression die quadrierte Summe aller Abweichungen minimiert. Die Support-Vector-Regression hingegen macht die Ausrichtung der Regressionsgeraden nur von den Support Vektoren abhängig und ignoriert somit die Ausreißer. Dies macht dieses Verfahren stabiler.

#### 4.4 Random-Forests-Regression

Entscheidungsbaum Methoden finden sich bei Mitchell (2018) und Gao und Chen (2022). Dabei sind diese simpel und nicht besonders präzise laut James et al. (2013b). Es sei denn man nutzt komplexere Weiterentwicklungen wie Random-Forests. Diese nutzen auch Holmgren et al. (2017), Broucke et al. (2019) und Mitchell (2018).

Ein Vorteil für diese Hausarbeit ist es, dass Entscheidungsbaum Methoden sowohl zur Klassifikation, als auch zur Regression genutzt werden können so James et al. (2013b). Die Begrifflichkeit röhrt von der Struktur in der Entscheidungsbäume Daten aufbereiten, denn ihre Darstellungsweise erinnert an eine umgedrehte Baumkrone, in der jede Astgabelungen ein Statement darstellt, welches mit wahr oder falsch beantwortet werden kann. Die Beob-

achtungen aus dem Datensatz folgen entlang ihrer Ausprägung verschiedenen Astgabelungen und werden so einer Vorhersage zugeordnet. Dies kann z.B. aussehen wie in Abbildung 4.4.

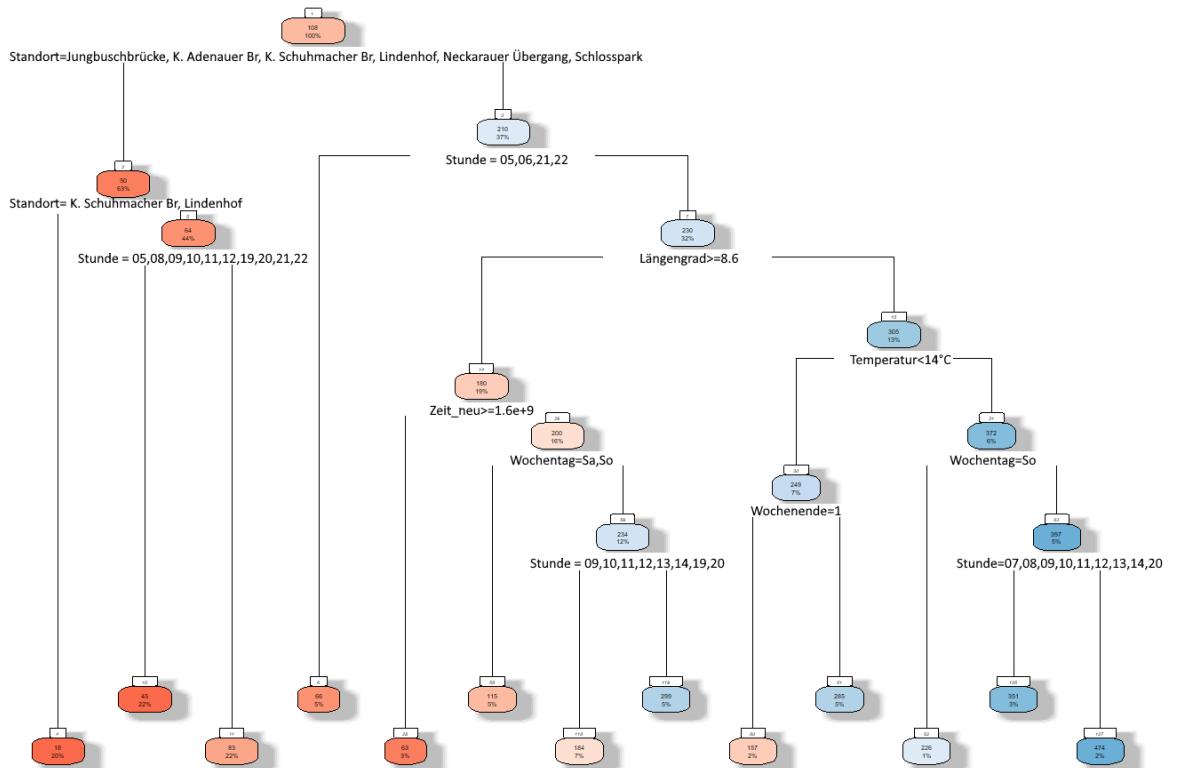


Fig. 4.4: Ein Entscheidungsbaum basierend auf Daten von Fahrradzählstationen in Mannheim und Daten des DWD 2016 bis 2022.

Quelle: Eigene Darstellung

Durch diese Entscheidungsgrenzen unterteilt der Entscheidungsbaum den Prädiktorenraum in distinkte sich nicht überlappende Regionen  $R_1, R_2, \dots, R_J$ . Das Optimierungsproblem besteht nun darin,  $R_1, R_2, \dots, R_J$  so zu wählen, dass die Summe der quadrierten Residuen (RSS) minimiert wird:

$$\sum_{j=1}^J \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2 \quad (4.5)$$

Auf diese Weise lässt sich ein Entscheidungsbaum erstellen. Bei den zuvor angesprochenen Random Forests wird diese Vorgehensweise mit einem Bootstraps Verfahren kombiniert. Bootstrapping bedeutet, dass durch zufällige Ziehungen von Beobachtungen aus einem bestehenden Datensatz, wobei mehrfach Ziehungen erlaubt sind, ein neuer Bootstraps-Datensatz erstellt wird, der dem ursprünglichen Datensatz ähnelt, jedoch davon abweicht. Diese Praxis führt man mehrere male durch und wendet auf jeden Bootstraps-Datensatz einen Entscheidungsbaum an, so das man eine große Anzahl von Entscheidungsbäumen erhält. Möchte man nun auf neuen Beobachtungen eine Vorhersage treffen, lässt man diese durch die Bootstraps Entscheidungsbäume laufen, wobei man sich auf die Vorhersage festlegt, die am meisten von den Bootstraps Entscheidungsbäumen bestimmt wurde. Dieses Verfahren führt zu einer deutlichen Steigerung der Vorhersagegenauigkeit.

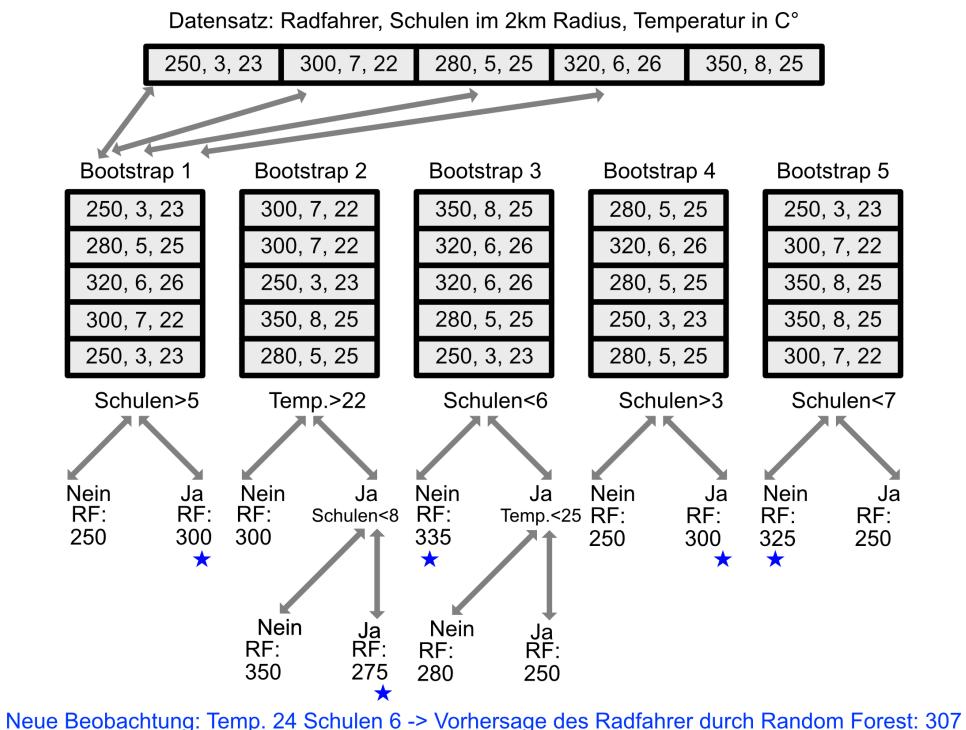


Fig. 4.5: Beispiel eines Random Forests

Die Abbildung 4.5 erläutert diesen Prozess nochmals anschaulicher. Zu

sehen ist ein Ausgangsdatensatz, der als Beispiel 5 Beobachtungen umfasst. Jede Beobachtungen beinhaltet die Anzahl der Radfahrer, die die Zählstation passierten, die Anzahl an Schulen, die sich in einem 2-Kilometer Radius um die Zählstation herum befinden und die Temperatur in C°. Die Bootstrap-Datensätze werden zufällig mit diesen Beobachtungen erstellt. So beinhaltet der erste Bootstrap-Datensatz die erste Beobachtungen zweimal, die zweite, die dritte und die vierte Beobachtung einmal und dafür die letzte Beobachtung gar nicht. So werden 5 Datensätze erstellt, die alle abweichende Entscheidungsbäume liefern. Möchte man nun auf diesen 5 Zufallsbäumen eine Vorhersage für eine neue Zählstation treffen, in deren Nähe sechs Schulen stehen und es 24 C° warm ist, dann entspricht diese Vorhersage dem Durchschnitt der fünf Vorhersagen der verschiedenen Entscheidungsbäume. Dies ergibt also 307 Fahrradfahrer.

## 4.5 Neuronale Netze

Deep-Learning-Algorithmen und die dazu gehörigen neuronalen Netzwerke sind mit die fortgeschrittenste Form im Bereich des Machine Learnings. Sie tauchen bei Broucke et al. (2019) und Mitchell (2018) auf.

### 4.5.1 Aufbau eines neuronalen Netzes

Zur Verdeutlichung der Funktionsweise von neuronalen Netzwerken beschränkt sich diese Arbeit aufgrund von Übersichtlichkeit auf Single-Layer-Perceptrons, diese sind die einfachste Form eines neuronalen Netzes und bestehen nur aus einer Schicht von Neuronen. Eine graphische Darstellung eines solchen Netzwerkes ist in Abbildung 4.6 zu sehen.

Mathematisch ausgedrückt, gleicht dieses Netzwerk einer Vektor Multiplikation. Im Input Vektor (Input Layer), sind die Variablen einer jeden einzelnen Beobachtungen zu finden. Diese werden mit dem Gewichten im Hidden Layer multipliziert. Wobei jeder einzelner Punkt in diesem Layer, auch Neuron genannt, über ein separates Gewicht verfügt. Diese sind so eingestellt, dass im Output eine möglichst zielgenaue Vorhersage zustande kommt. Als Formel

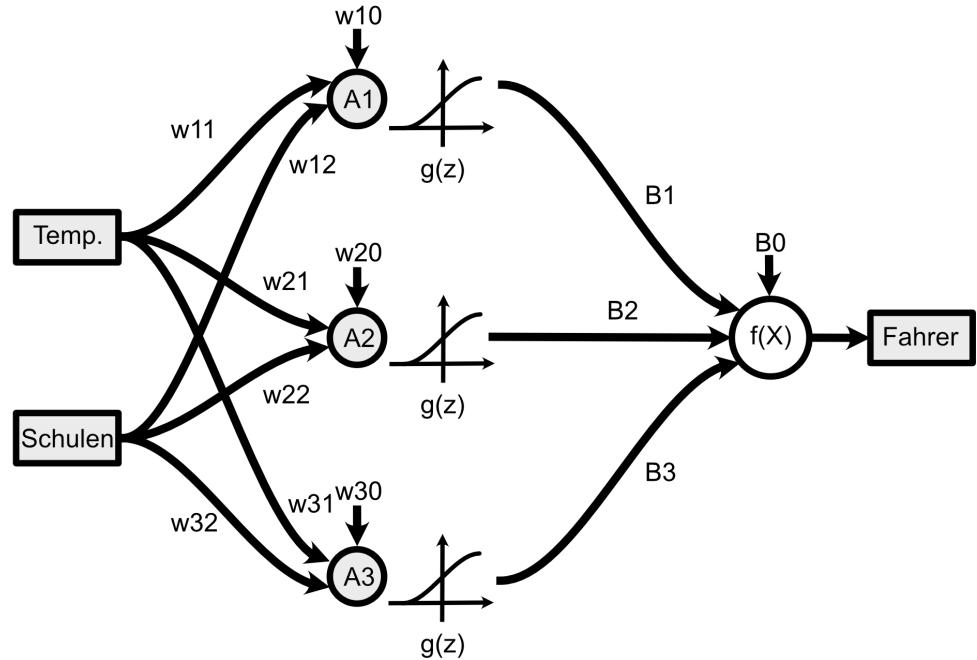


Fig. 4.6: Beispiel eines neuronalen Netzes

sieht diese Vorgehensweise wie folgt aus:

$$\begin{aligned}
 f(x) &= \beta_0 + \sum_{k=1}^K \beta_k h_k(X) \\
 &= \beta_0 + \sum_{k=1}^K \beta_k g(w_{k0} + \underbrace{\sum_{j=1}^p w_{kj} X_j}_{\text{Aktivierungsfunktion}})
 \end{aligned} \tag{4.6}$$

Ob und in welchem Umfang die Information, die ein Neuron passiert, weitergereicht wird, entscheidet die Aktivierungsfunktion  $g(z)$ . In ihr werden die Informationen des Inputs und die der Gewichte übertragen. Das kann z.B. durch eine Sigmoid Funktion (4.7) oder durch eine ReLU (Rectified Linear Unit) Funktion (4.8) geschehen.

$$g(z) = \frac{e^z}{1 + e^z} = \frac{1}{1 + e^{-z}} \tag{4.7}$$

$$g(z) = \begin{cases} 0 & \text{wenn } z < 0 \\ z & \text{sonst.} \end{cases} \quad (4.8)$$

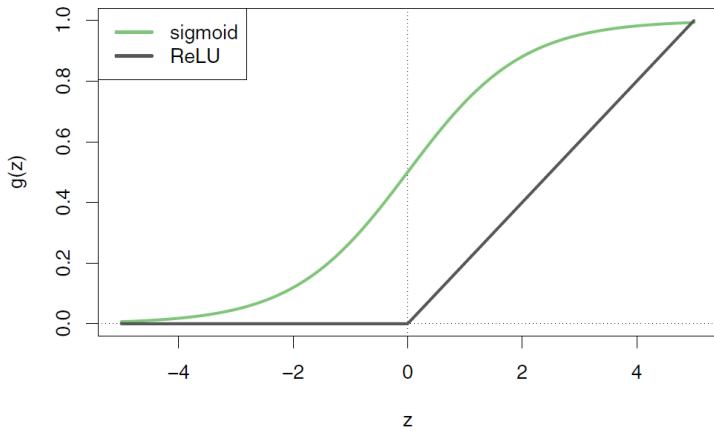


Fig. 4.7: Aktivierungsfunktionen

Quelle: James et al. (2021)

Die Sigmoid Funktion wird z.B. auch für die logistische Regression genutzt. Der Vorteil der ReLU Funktion besteht darin, dass Neuronen ausgeschaltet werden, sollte  $z$  zu klein sein, wodurch Rechenkraft gespart wird. Zu dem reagiert das Neuronale Netz stärker auf höhere Werte von  $z$  und lernt damit schneller.

#### 4.5.2 Berechnung eines neuronalen Netzes

Wichtig damit die Aktivierung eines Neurons zur richtigen Vorhersage führt, ist das richtig trainierte Gewicht  $w_{kj}$ . Sowohl die Parameter  $\beta_0, \dots, \beta_K$  als auch  $w_{10}, \dots, w_{Kp}$  müssen trainiert werden. Das Optimierungsproblem dazu besteht in:

$$\min_{\{w_k\}_1^K, \beta} \frac{1}{2} \sum_{i=1}^n (y_i - f(x))^2 \quad (4.9)$$

Um diese Optimierung vorzunehmen, verwendet das neuronale Netz den Vektor  $\theta$ , der alle zu optimierenden Variablen enthält. Der Fehlerterm ergibt

dann:

$$R(\theta) = \frac{1}{2} \sum_{i=1}^n (y_i - f_\theta(x))^2 \text{ mit } \theta = (\{w_k\}_1^K, \beta) \quad (4.10)$$

Zu Beginn des Optimierungsprozesses wird  $\theta^{t=0}$  angenommen, dass heißt der Vektor beginnt ungewichtet. Darauf hin folgen Wiederholungen eines Prozesses, bei dem ein Vektor  $\delta$  gefunden werden muss, sodass  $\theta^{t+1} = \theta^t + \delta$  zu  $R(\theta^{t+1}) < R(\theta^t)$  führt. In jedem Durchlauf dieses Prozesses wird  $t$  erhöht. Eine eindimensionale Veranschaulichung dieses Prozesses ist in Abbildung 4.8 zu finden.

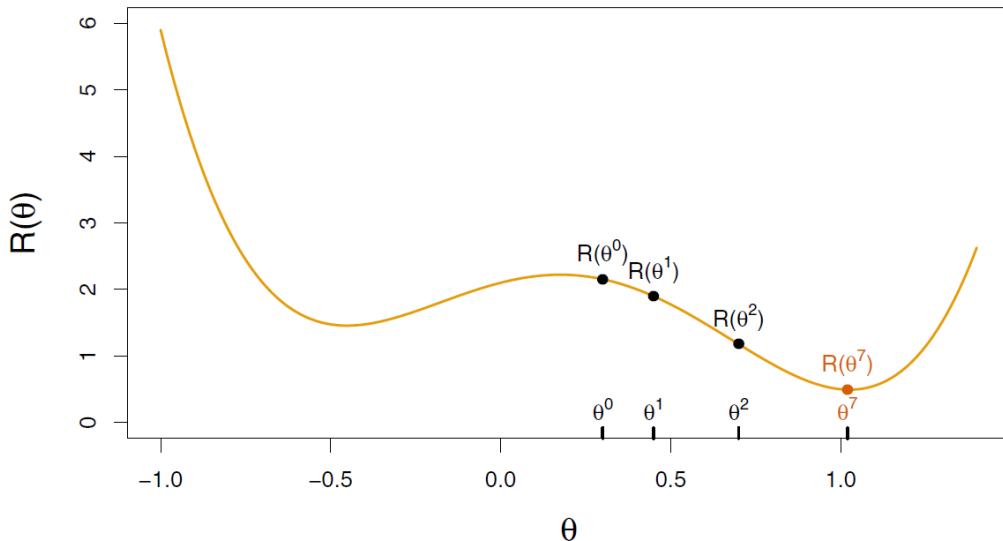


Fig. 4.8: Gradient Descent

**Quelle:** James et al. (2021)

Wie kann aber der Vektor  $\delta$  gefunden werden, sodass der Fehlerterm gesenkt wird? Dazu muss man den Vektor des Gradienten berechnen:

$$\Delta R(\theta^m) = \frac{\partial R(\theta)}{\partial \theta}|_{\theta=\theta^t} \quad (4.11)$$

Der partielle Vektor der Ableitung der aktuellen Schätzung von  $\theta^t$  zeigt aufwärts. Um also  $\delta$  zu finden gehen wir in die entgegengesetzte Richtung

gewichtet mit der Lernrate  $\rho$ . So ergibt sich:

$$\theta^{t+m} \leftarrow \theta^m - \rho \Delta R(\theta^m) \quad (4.12)$$

Die Ableitungen des Fehlerterms nach den Gewichten, die wir brauchen, um den Gradientenvektor in 4.11 zu bestimmen, kann man durch die Ketten Ableitungsregel vereinfachen:

$$\begin{aligned} \frac{\partial R_i(\theta)}{\partial \beta_k} &= \frac{\partial R_i(\theta)}{\partial f_\theta(x_i)} \frac{\partial f_\theta(x_i)}{\partial \beta_k} \\ &= -(y_i - f_\theta)g(z_{ik}) \\ \frac{\partial R_i(\theta)}{\partial w_{kj}} &= \frac{\partial R_i(\theta)}{\partial f_\theta(x_i)} \frac{\partial f_\theta(x_i)}{\partial g(z_{ik})} \frac{\partial g(z_{ik})}{\partial z_{ik}} \frac{\partial z_{ik}}{\partial w_{kj}} \\ &= -(y_i - f_\theta)\beta_k * g'(z_{ik})x_{ij} \\ \text{mit } z_{ik} &= w_{ko} + \sum_{j=1}^p w_{kj}x_{ij} \end{aligned} \quad (4.13)$$

## 4.6 Validation

Statistische Modelle, wie ein neuronales Netz, machen erstaunlich präzise Vorhersagen. Doch wie kann man eine solche Präzession messen? Dafür verantwortlich ist die Validierung. Mittels Maße, wie dem Bestimmtheitsmaß  $R^2$  oder dem RMSE, kann man die Performance der Vorhersagen im Vergleich mit den tatsächlichen Daten messen.

Doch wie validiert man die Vorhersagekraft eines Modells auf Daten, mit denen das Modell selbst nicht trainiert worden ist, um zu testen ob das Modell die eigenen Daten nicht overfittet? Hierzu ließe sich der Validation-Set-Approach nutzen. Teilt man die Beobachtungen in zwei unterschiedlichen Sets zufälligerweise auf, erhält man ein Set, mit dem man ein Modell trainieren kann und eines, mit dem man es testen kann. Die Quote, nach der man diese Beobachtungen aufteilt, nennt man Split. Ein häufig verwendeteter Split ist 80 % der Beobachtungen für das Trainingsset und 20 % für das Testset.

#### 4.6.1 Cross Validation

Eine Weiterführung dieses Konzepts ist die Cross Validation. Hier wird der Datensatz nicht in 2 sondern in  $k$  viele und gleichgroße Sets unterteilt. Es werden dann  $k$  viele Validierungen vorgenommen. Bei jedem Durchlauf wird eines der Sets zur Validierung und die restlichen Sets zum Training verwendet. Das bietet den Vorteil, dass der gesamte Datensatz zur Validierung und zum Training verwendet wird, während beim herkömmlichen Ansatz immer auf ein Teil der Daten verzichtet werden musste. Außerdem werden so mehr Stichproben erhoben. Man vergleicht nicht ein Modell, sondern  $k$ -viele Modelle. Es kann immer vorkommen, dass viele Modelle eine solide Performance bieten, während eines der Modelle in einem anderem Set zeigt, dass es zu drastischen Fehlern kommen kann. Das heißt die Wahrscheinlichkeit Overfitting aufzudecken ist größer. Der Prozess der Cross Validation wird in der Abbildung 4.9 mit einem Beispiel näher beschrieben, in dem  $k = 5$  ist und so fünf Sets gebildet werden aus den vorhandenen Beobachtungen im Datensatz. Mit jedem Set kann ein Test durchgeführt werden, wobei ein Set jeweils für den Test selbst genutzt wird und die übrigen Sets zum Training des Modells.

#### 4.6.2 Conditional-Validation-Set-Building

In dem Fall der Daten der Radzählstation gäbe es das zusätzliche Problem, dass auch die Validierung mit dem Validation-Set-Approach Overfitting wahrscheinlich nicht richtig erkennen ließe. Denn durch eine rein zufällige Aufteilung der Daten in verschiedene Sets ließe sich die wahre externe Validität des Modells nicht prüfen, denn auch dann träfe das Modell immer Vorhersagen für fixe räumliche Punkte, die bereits auch im Trainingsdatensatz vorkamen.

Um dieses Problem zu umgehen, muss man die Aufteilung der Daten in die verschiedenen Sets auf die Ausprägung der Stationen konditionieren. Das heißt, jede Zählstation darf in jedem Set nur einmal vorkommen, damit im Testset immer Orte vorhanden sind, die dem Modell unbekannt sind. Diese Aufgabe übernimmt der folgende Code:

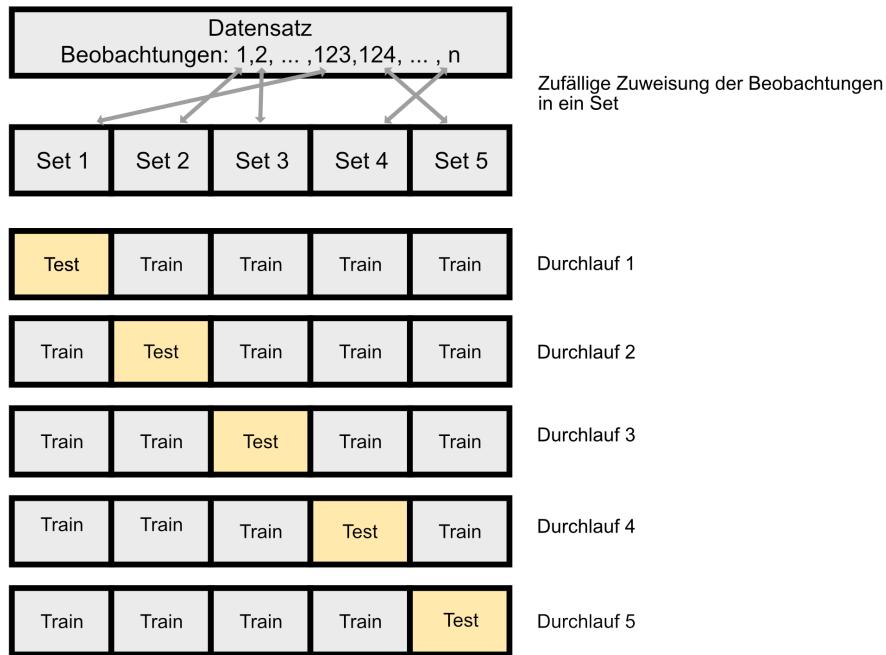


Fig. 4.9: Beispiel einer Cross Validation

Listing 4.1: Aufteilung der Zählstationen

```

1 while( length(stations_not_chosen)>0)
2 {
3     x1 <- sample(1:length(stations_not_chosen), 1)
4     newStation = stations_not_chosen[x1]
5
6     stations_splits[stations_splits$Station ==
7         newStation ,]$Split = actual_split
8
9     observations_per_splits @@
10    Observations[actual_split] =
11    observations_per_splits @@
12    Observations[actual_split] +
13    Obs_perStation[Obs_perStation$Station ==
14    newStation ,]$Observations

```

```

15
16     stations_not_chosen = stations_not_chosen[-x1]
17
18     if (median(observations_per_splits$Observations)
19         <= observations_per_splits$Observations[actual_split]){
20
21         actual_split = actual_split + 1
22         if (actual_split>validation_splits)
23             {actual_split=1}
24
25     }
26
27 }
```

In dem Quellcode 4.1 werden innerhalb einer `while()`-Schleife alle Stationen einem der k-vielen Sets hinzugefügt, bis die Schleife die Anzahl der zu verordnenden Zählstationen in `stations_not_chosen` herunter gezählt hat. Die Anzahl von k ist 5 und ist in der Variable `validation_splits` gespeichert. Damit alle Validation-Sets ungefähr gleich viele Beobachtungen haben, zählt der Quellcode 4.1 mit, wie viele Beobachtungen mit jeder Station einem Set hinzugefügt werden. Diese Anzahl wurde zuvor in der Tabelle `Obs_perStation` gespeichert. Ob im folgendem Durchlauf der `while()`-Schleife das nächste Validation-Set ausgewählt wird oder nochmal dasselbe, hängt von der Anzahl der Beobachtungen ab, die bereits im aktuellem Set vorkommen. Jedes Validation-Set wird solange mit Zählstationen befüllt, bis der Median an Beobachtungen aller Sets gleich ist, oder überschritten wurde. Dies wird in der `if()`-Schleife abgefragt.

Auf diese Weise erhält man fast gleichgroße unterschiedliche Validation-Sets, in denen jede Zählstation nur einmal vergeben ist. Teilweise kommt es auch vor, dass ganze Städte nur in je einem Validation-Set vertreten sind, weil es Städte gibt, die nur eine Zählstation mit sich bringen.

#### 4.6.3 Weighted-Subset-Building

Im Verlauf dieser Masterarbeit sind einige Probleme durch die Größe des Datensatzes entstanden. Da der Datensatz über vier Millionen Beobachtungen beinhaltet und viele Variablen inkludiert, ist dessen Speicherplatz auf bis zu 2,7 GB angestiegen. Ein Problem von R ist es, dass alle Daten mit denen gerechnet wird, im Arbeitsspeicher passen müssen. Die Hardware mit deren Hilfe die Modelle der Arbeit entstanden sind, verfügt jedoch nur über 16 GB. Je komplexer die Modelle wurden, desto öfters versagten Skripte mit der Fehlermeldung "run out of memory". Es gäbe an dieser Stelle verschiedene Lösungsansätze. Einer wäre Cloud Computing oder High-Performance-Computing. Diese Lösungsansätze sind jedoch mit technischen Problemen verbunden.

Der Amazon Web Service bietet z.B. Cloud-Computing-Dienste an, die auf das Machine-Learning-Training spezialisiert sind, ist jedoch mit der benötigten Leistung auch kostenpflichtig. Darüber hinaus bietet die Westfälische-Wilhelms-Universität-Münster High-Performance-Computing-Ressourcen an, die über eine Linux Kommandozeile steuerbar sind. Jedoch müsste dafür erst der Umgang mit Linux vertraut sein.

Beachtet man dies, ist die einfachste Lösung die Größe des Datensatzes zu reduzieren durch Undersampling. Dies mag kontraintuitiv erscheinen, ist jedoch sogar mit Vorteilen verbunden. Denn erstellt man ein gutes Subset, dass den Datensatzes mit einem Teil von Beobachtungen darstellt, kann dies dabei helfen Overfitting zu verhindern oder besser zu entdecken.

Der Ansatz, der hier dabei verfolgt wurde, war es, möglichst gleich viele Beobachtungen von allen Zählstationen zufällig auszuwählen, dabei aber die Größe jedes Validation-Sets auf 200000 Beobachtungen zu begrenzen, sodass der Datensatz auf eine Million Beobachtungen schrumpft. Durch diese Vorgehensweise ist der Anteil an Beobachtung im Datensatz nach räumlicher Verteilung stärker gleichverteilt als zuvor, was dabei helfen sollte, Overfitting besser aufzudecken und auch zu verhindern.

## 5. ERGEBNISSE

Das vorherige Kapitel erklärte die Funktionsweise verschiedener statistischen Methoden, die sich zur Vorhersage des Radverkehrs anwenden ließen. Viele haben ihre Vor- und Nachteile, alle wurden in der bisherigen Literatur, die mit diesem Thema verwandt ist, verwendet. Folglich wurden alle vier Methoden angewendet und mit der Cross Validation evaluiert und lassen sich so in ihrer Vorhersagekraft vergleichen. Eine Gegenüberstellung der Ergebnisse findet sich hier.

### 5.1 OLS-Regression

Das simpelste Modell ist die OLS-Regression. Diese Methode benötigt nur wenig Rechenkraft und stößt somit nicht an technische Flaschenhälse wie andere Methoden, leistet dafür jedoch nur ungenaue Vorhersagen. Dies ist auch zu entnehmen der Abbildung 5.1. Zum Vergleich wurden sechs unterschiedliche Log-Lin-Regressionsmodelle erstellt und jeweils validiert, wobei jedes weitere Modell neue Variablen hinzufügt. Das erste Modell enthält nur zeitliche Variablen wie Jahr, Monat, Stunde, Wochenende, Nacht und Feiertage. Das zweite Modell fügt dem die fünf Wetter Variablen Regen, Temperatur, Bedeckung, relative Feuchte und Windgeschwindigkeit hinzu. Im dritten Modell werden demographische Daten hinzugefügt, also die Altersgruppen, Bevölkerungszahl, Relation PKWs zu Einwohnern, Ausländeranteil und Stadtgröße. Im vierten Modell waren es räumliche Variablen. Das umfasst die Daten zu den POIs, wie Schulen, Universitäten oder Geschäften, die Anzahl an Ampeln und Straßenübergängen und den Straßentyp. Im fünften Modell werden einige quadrierte Variablen hinzugefügt und im letzten kubische Variablen. Jedes Modell wurde fünf mal erstellt, für die fünf unterschiedlichen Trainings- und

Testsets. Außerdem nutzt jedes Modell den ersten Datensatz, welcher noch nicht die Corona-Effekte beinhaltet.

Die roten Linien in der Abbildung 5.1 zeigen den Verlauf des Bestimmtheitsmaßes  $R^2$  für die Trainingsdatensätze und zeigen, dass je mehr Variablen in das Modell aufgenommen werden, desto weiter steigt der erklärte Anteil des Modells, wobei ein abnehmender Grenzeffekt der Variablenanzahl zu sehen ist. Die blauen Kurven hingegen zeigen den Verlauf des Bestimmtheitsmaßes in den Testdatensätzen. Damit beweisen sie die Out-Of-Sample-Validität der jeweiligen Modelle. Hier ist festzustellen, dass kein anhaltender steigender Trend zu erkennen ist und sogar im Gegenteil, dass die Modelle mit mehr Variablen zu schlechteren Ergebnissen führen. Dies zeigt ganz klar, dass hier Overfitting stattfindet.

Dass hier aber Overfitting auftritt ist kein Wunder. Die Zählstationen, aus deren Beobachtungen der Datensatz zum Trainieren des Modells besteht, werden nicht rein zufällig gesetzt von den Kommunen, sondern meist an Stellen, an denen Sie einen hohen Radverkehr vermuten. Das heißt, die Beobachtungen des Datensatzes sind nicht unabhängig und gleich verteilt, womit das OLS-Modell natürlich invalide Vorhersagen trifft, die die eigentlichen Beobachtungen zwar zutreffend vorhersagen, nicht aber darüber hinaus. Die Frage ist ob andere Machine-Learning-Methoden hier eine bessere Performance bieten, als das OLS-Modell. Diese Frage klärt sich in den kommenden Sektionen. Grundsätzlich deckt das Validierungsverfahren Overfitting auf, dass durch die Ungleichverteilung der Daten entsteht. Ein Modell, dass im Test Set eine gute Performance zeigt, wäre unbedenklich. Die Performance des OLS Modells hier ist jedoch unbrauchbar. Bei allen folgenden Machine-Learning-Algorithmen muss berücksichtigt werden, dass die Gefahr zum Overfitting groß ist, und das nie die Performance im Testset erreicht wird, die sonst im Trainingsset erreicht werden kann. Die Tabelle 5.1 zeigt die Performance des endgültigen OLS Modells durch den Bestimmtheitswert und den RMSE in jedem einzelnen Split. Split 3 zeigt die tatsächliche Tragweite des Overfittings, denn wo hier im Trainingsset ein RMSE von 125 steht, nimmt der RMSE im Testset astronomische Größen an. Dieses Beispiel zeigt auch den

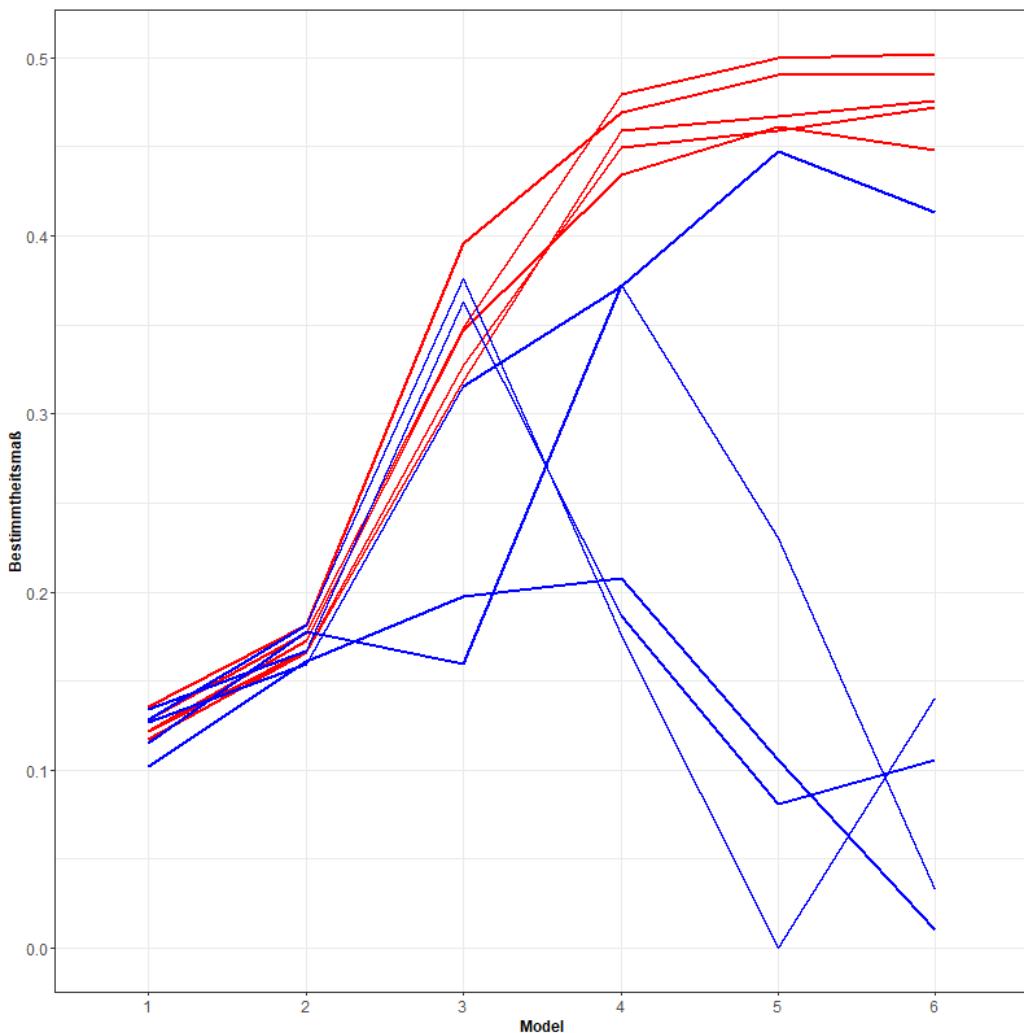


Fig. 5.1: Feature Selection Ergebnis der OLS-Regression in 6 Modellen

Vorteil der Cross Validation, denn dadurch, dass bei der Cross Validation 5 unterschiedliche Test Sets genutzt werden, steigt die Wahrscheinlichkeit, solche Modellfehler offenzulegen.

Es wird also deutlich, dass sich ein Log-Lineares-OLS-Modell nicht dazu eignet, räumliche Vorhersagen für einen ganzen Stadtbereich zu machen. Weiterhin lohnt es sich nicht, hier weitere Versuche zu unternehmen. Und damit bleiben noch drei weitere Algorithmen übrig, wobei der nächst folgende die Support-Vector-Regression ist.

Sets	Train $R^2$	Train RMSE	Test $R^2$	Test RMSE
1	0,448	127,441	0,105	184,489
2	0,502	124,486	0,413	113,692
3	0,49	124,969	0,01	153419513741494000
4	0,472	124,189	0,036	2239,056
5	0,476	119,869	0,141	196,462
Schnitt	0,478	124,191	0,14	30683902748299400

Tab. 5.1: Performance des OLS-Modells

## 5.2 Support-Vector-Regression

Unter allen Methoden ist die Support-Vector-Regression jene, die die größten Ähnlichkeiten hat zur herkömmlichen OLS-Regression. Vorteile sind aber, dass die Support-Vector-Regression weniger stark auf Ausreißer reagiert als die OLS-Regression. Neben diesen Vorteil hat die Support-Vector-Regression den Nachteil, dass die Berechnung des Modells deutlich mehr Zeit in Anspruch nimmt und deswegen leistungsfähigere Hardware erfordert.

Das R-Paket „e1071“, dass die passende Funktion bietet, stammte von Meyer et al. (2021). R hat den Nachteil, dass es eine Single-Threaded-Sprache ist. Es spielt also keine Rolle, wie leistungsfähig die verwendete CPU ist, solange sich deren Leistung auf verschiedene CPU Kerne aufteilt. R nimmt zur Berechnung immer nur einen CPU Kern. Diese Limitation ließe sich umgehen mit dem Paket „foreach“ von Microsoft und Weston (2022) und „doParallel“ von Microsoft und Weston (2022), jedoch wird dieses Vorgehen verhindert durch die vorhandene Limitation im Arbeitsspeicher. R legt alle Daten, mit denen ein Skript rechnet, in den Arbeitsspeicher ab. Würden verschiedene Modelle gleichzeitig von verschiedenen CPU Kernen berechnet werden, dann müssten auch verschiedene Tests und Trainings Datensätze erstellt werden müssen und dies würde in Summe den zur Verfügung stehenden Arbeitsspeicher von nur 16 GB überlasten.

Eine Möglichkeit dieses Problem zu umgehen, wäre es z.B. auf den Amazon-Web-Service zurückzugreifen, die Server zum Trainieren von Machine-Learning-Algorithmen anbieten, wie bereits erwähnt. Einfacher ist es jedoch, nur einen kleinen zufällig ausgewählten Anteil des Datensatzes zu verwenden. Dabei

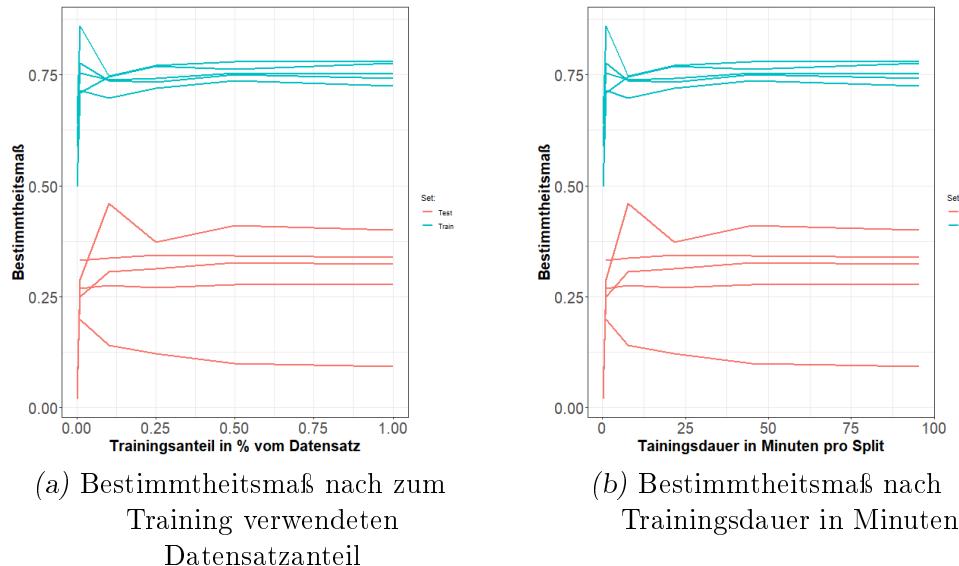


Fig. 5.2: Support-Vector-Regression-Performance nach Anteil des Datensatzes

besteht natürlich die Gefahr, dass die Vorhersagekraft des Modells leidet, wenn man zu wenig Datenpunkte des Datensatzes verwendet. Der Datensatz beinhaltet insgesamt 4.472.091 Beobachtungen. Davon wurden bei der OLS-Regression jeweils 80 % zum trainieren des Modells genutzt. Reduziert man diese Trainingsdaten weiter und wählt zufällige Beobachtungen aus, dann verhält sich die Performance des Modells so, wie in der Abbildung 5.2 zu sehen, wobei blau die Performance in den Trainingssets zeigt und rot die Performance in den Testsets.

Dabei wurde das Modell einmal mit 0.001 %, 0.01 %, 0.1 %, 0.25 %, 0,5 % und 1 % der Beobachtungen im Trainingsset getestet. Als erstes fällt auf, dass die Support-Vector-Regression bereits bessere Ergebnisse aufweist als die OLS-Regression, so steigt der Bestimmtheitswert für das Trainingsset auf bis zu 75 % im Schnitt an, wo es zuvor noch um die 47 % beim OLS-Modell waren. Aber auch hier findet wieder Overfitting statt, denn der Bestimmtheitswert im Test Sets steigt nur auf 25 % durchschnittlich an. Erhöht man den Anteil an Beobachtungen, den man zum Training des Modells verwendet, auf 0,25 %, dann findet kaum noch eine nennenswerte Veränderung in der Performance statt. Nur die Dauer die zur Berechnung benötigt wird, steigt

signifikant an. Nutzt man 0,25 % der Daten zum Training, was ungefähr 9000 Beobachtungen entspricht, dann benötigt der Computer mit einer Intel Core i7-8750H CPU und 2,2 GHz Leistung im Schnitt 21 Minuten je Modell. Bei 1 % verwendeter Daten, was ungefähr 37.800 Beobachtungen entspricht, benötigt die Berechnung bereits im Schnitt 90 Minuten pro Modell. Da jeweils 5 Modelle trainiert werden, auf 5 unterschiedlichen Cross-Validation-Sets und das sechs mal zu unterschiedlichen Anteilen am Datensatz, hat die Erstellung der Ergebnisse in Abbildung 5.2 14,2 Stunden benötigt.

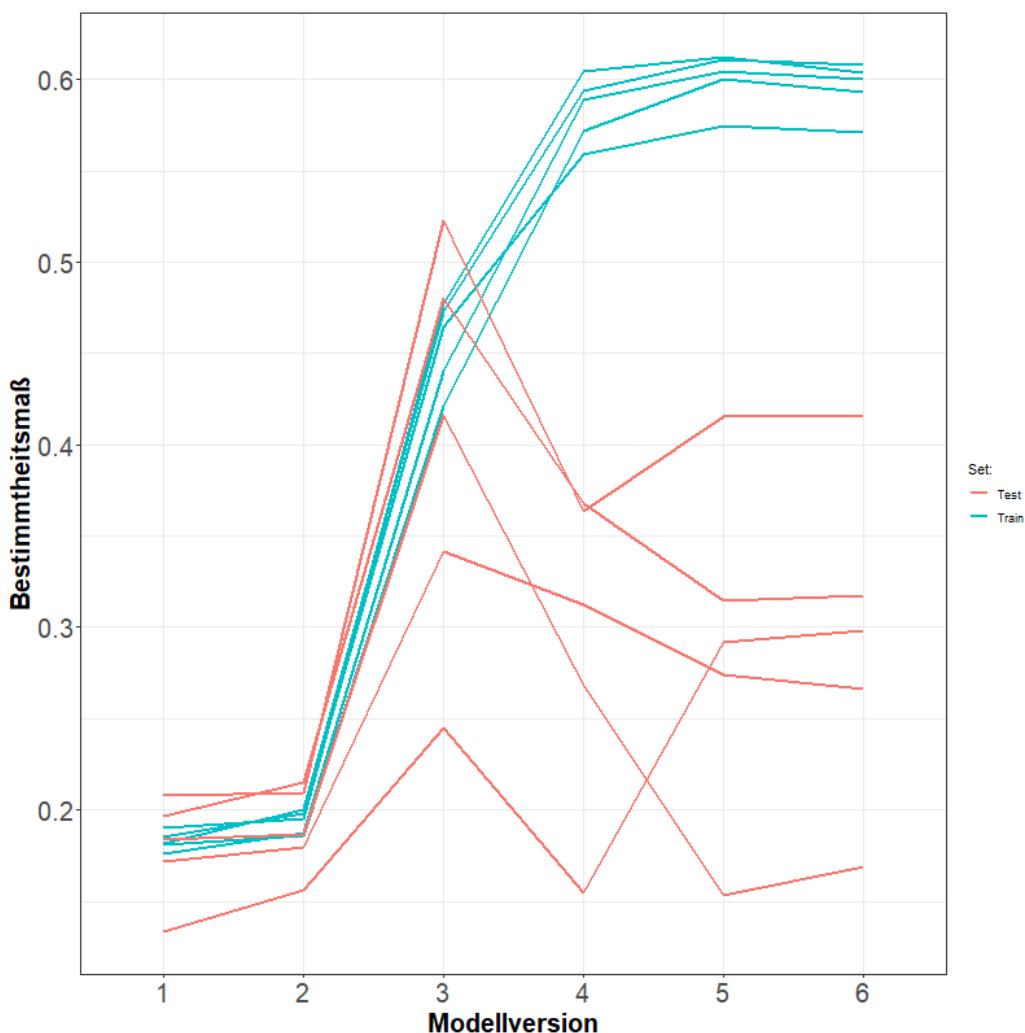


Fig. 5.3: Support-Vector-Regression Ergebnis in 6 Modellen

Darüber hinaus kann man auch hier die Performance des Modells anhand

der Auswahl der Variablen im Modell vergleichen. Auch hier werden im ersten Modell ausschließlich zeitliche Variablen verwendet, im zweiten kommen die fünf Wettervariablen hinzu. Das dritte Modell wird erweitert mit Variablen die Demographie, Stadtausdehnung, Autobesitz und Fahrradklima anzeigen. Im vierten Modell werden die räumliche Variablen von Open-Street-Map hinzugefügt, im fünften Modell Variablen zu den Straßentypen und im sechsten Modell nicht lineare Variablen. Kubische Variablen sind dieses Mal in keinem Modell enthalten, weil dies zu einer Überlastung des Arbeitsspeichers führte und nach den Erfahrungen beim OLS-Modell nicht davon auszugehen ist, dass sie signifikante Besserungen mit sich bringen. Alle Modelle wurden mit einem Trainingsanteil von 0,05 % berechnet, also ungefähr 17.900 Beobachtungen und stammen wiederum aus dem ersten Datensatz, der keine Corona-Daten beinhaltet. Den Vergleich zwischen den Modellen zeigt die Abbildung 5.3. Diese zeigt deutlich, dass das geringste Maß an Overfitting im dritten Modell stattfindet, erst danach nimmt das Overfitting zu. Das dritte Modell allein ist jedoch von wenig Nutzen, denn ohne die räumlichen Variablen, gibt es keine Möglichkeit, räumliche Unterschiede in der Vorhersage hervorzuheben, was der eigentliche Nutzen des Modells sein soll. So ist auch die Support-Vector-Regression nicht dazu geeignet, räumliche Vorhersagen zum urbanen Fahrradverkehr zu treffen.

Dies bestätigt auch ein weiterer Blick in die Tabelle 5.2, die die Performance des endgültigen Support-Vector-Modells zeigt. Im Vergleich zum OLS-Modell sehen wir keine übermäßigen Ausreißer mehr im RMSE der Testdatensätze, aber die Performance verbessert sich nicht ausreichend genug, um damit vernünftige Aussagen treffen zu können.

### 5.3 Random-Forest-Regression

Die bisherigen Versuche, ein schlüssiges Modell zu berechnen, förderten leider nur unzufriedenstellende Ergebnisse hervor. Deswegen wird es Zeit leistungsfähigere Methoden zu verwenden. Das R-Paket „randomForest“ von Liaw und Wiener (2002) bietet eine solche Methode. Ähnlich wie Support-Vector-Regressionen sind Random Forest deutlich rechenintensiver als eine OLS-

Sets	Train $R^2$	Train RMSE	Test $R^2$	Test RMSE
1	0,741	86,075	0,129	169,814
2	0,768	85,717	0,281	129,19
3	0,765	83,096	0,319	132,127
4	0,742	84,941	0,372	138,756
5	0,745	80,34	0,307	169,438
Mean	0,752	84,034	0,282	147,865

Tab. 5.2: Performance des SVR-Modells

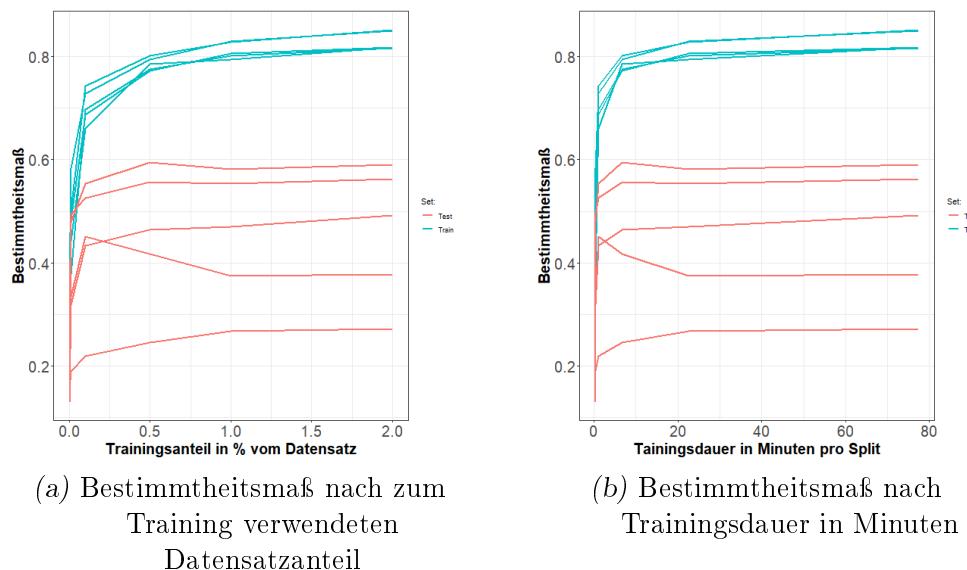


Fig. 5.4: Random-Forest-Performance nach Anteil des Datensatzes

Regression. Deswegen wird auch hier nur ein Teil der Daten zum Training verwendet. Wie sich die Performance des Random-Forest-Models nach Anteil der zur Berechnung verwendeten Daten ändert, zeigt die Abbildung 5.4, deren Berechnung insgesamt 9,35 Stunden benötigte. Dabei wird sichtbar, dass qualitativ wie auch quantitativ die Random Forests performanter sind, als die vorherige Support-Vector-Regression. In Abbildung 5.2 war noch zu sehen, dass die Berechnung mit 1 % der Daten ca. 90 Minuten pro Modell benötigte. Hier sind es für 2 % nur noch 80 Minuten.

Auch hier findet wieder Overfitting statt. Dennoch ist die Vorhersagekraft bedeutend besser als zuvor. Im Trainingsset erreicht das Random-Forest-

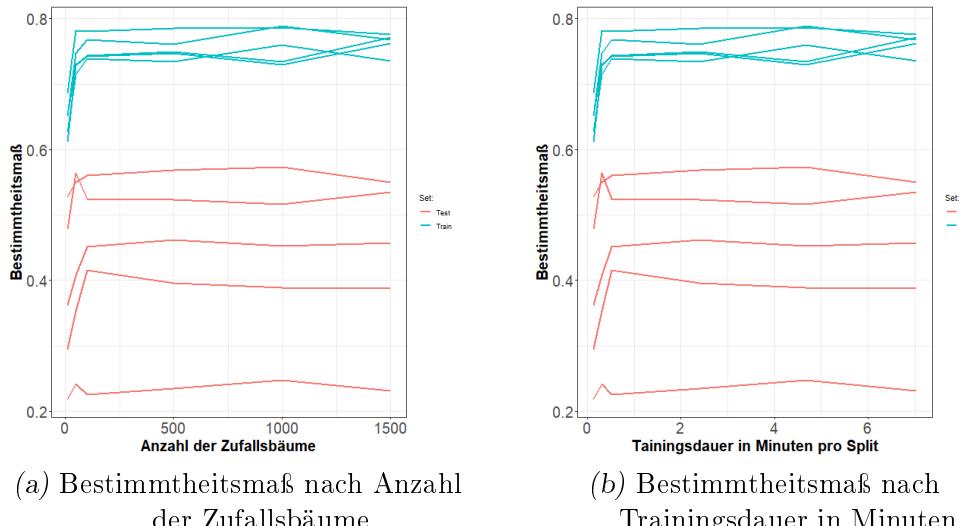


Fig. 5.5: Random-Forest-Performance nach Anzahl der Zufallsbäume

Modell einen Bestimmtheitswert von 80 %, während die Support-Vector-Regression 75 % maximal erreichte. Im Testset sieht der Unterschied noch besser aus. Hier werden nun bis zu 60 % erreicht. Zur Berechnung der Abbildung 5.4 wurden im Modell 500 Zufallsbäume verwendet. Diese Anzahl lässt sich variieren. Wie sich die Performance des Modells mit der Anzahl an verwendeten Zufallsbäumen verändert, zeigt die Abbildung 5.5, deren Berechnung 1,27 Stunden gebraucht hat. Dabei fällt schnell auf, dass das Maximum an Performance mit bereits relativ wenigen Zufallsbäumen erreicht ist.

Darüber hinaus bleibt nur noch der Vergleich von Modellen mit verschiedenen Variablen übrig. Dazu fand die selbe Modellauswahl wie im Abschnitt zur OLS-Regression statt. Das heißt das Modell 1 greift auf Variablen zum Jahr, Monat, der Stunde, zum Wochenende, Nacht, und den Ferien- wie auch Feiertagen zu. Modell 2 fügt Wettervariablen hinzu. Modell 3 nutzt zusätzlich demographische Variablen, Bevölkerungsanzahl, Fläche, Autobesitzrate, Immigrantenanteil und Fahrradklimaindex. Im Modell 4 kommen die Open-Street-Map-Variablen hinzu. In den darauf folgenden Modellen kommen noch die nicht linearen Effekte hinzu. Zu sehen ist der Vergleich in der Performance der unterschiedlichen Modelle in der Abbildung 5.6. Einen ganz signifikanten Anstieg in der Performance ist zwischen Modell 2 und 3 zu sehen. Danach

findet kaum eine große Veränderungen in der Performance statt sowohl im Trainingsdatensatz als auch im Testdatensatz.

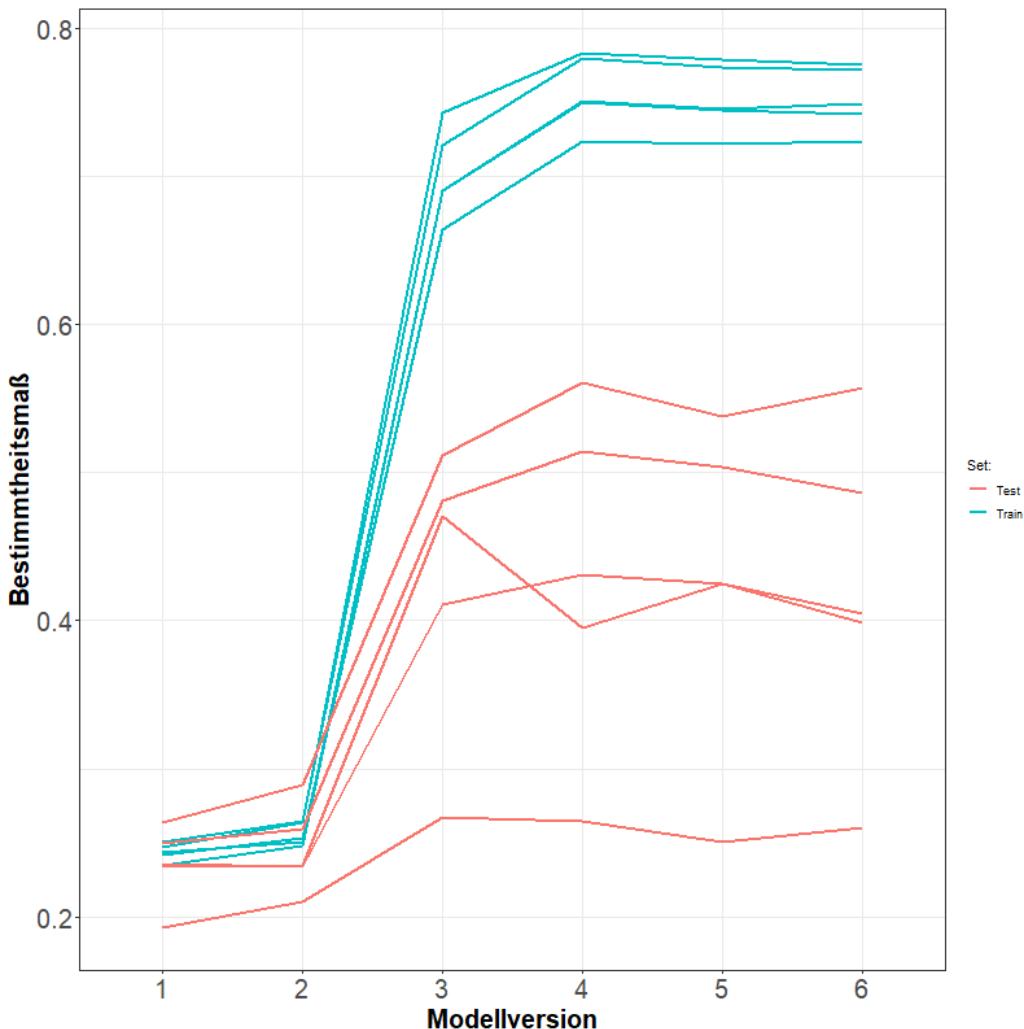


Fig. 5.6: Feature Selection Ergebnis der RF Regression in 6 Modellen

Kombiniert man die so gewonnen Erkenntnisse zur Performance von Random Forests, dann kommt man zu den Ergebnissen der Cross Validation in der Tabelle 5.3. Dieses Modell nutzte 0,5 % der Trainingsdaten, 250 Zufallsbäume, verzichtet auf nicht lineare Variablen und benötigte zur Berechnung 24 Minuten. Im Test Datensatz erreicht das Random-Forest-Modell ein Bestimmtheitsmaß von 44,5 % im Schnitt. Das ist mehr als dreimal so viel wie im OLS-Modell und mehr als 1,5 mal so viel wie mit der Support-Vector-

Sets	Train $R^2$	Train RMSE	Test $R^2$	Test RMSE
1	0,77	83,783	0,368	139,04
2	0,809	79,122	0,473	116,077
3	0,813	78,826	0,265	134,78
4	0,778	84,372	0,542	140,007
5	0,784	77,452	0,575	154,054
Mean	0,791	80,712	0,445	136,791

Tab. 5.3: Performance des RF Modells

Regression. Auch übertrifft dies knapp die Ergebnisse von Alattar et al. (2021), die nur 42 % erreichten. Auch wenn diese Werte bereits deutlich besser sind, als die der vorherigen Modelle, kann dies nicht darüber hinweg täuschen, dass sowohl die Höhe des Bestimmtheitsmaßes als auch des RMSE noch zu wünschen übrig lässt. Doch in Anbetracht der schwierigen Aufgabe, die dieses Modell zu erfüllen hat, sind die Ergebnisse dennoch interessant, denn das Modell hat nur wenige Beobachtungsorte, auf deren Grundlage es Vorhersagen für ein ganzes Stadtgebiet treffen soll. Die letzte Methode, die nun noch übrig bleibt und die vielleicht in der Lage ist, noch bessere Vorhersagen zu treffen, ist das neuronale Netz.

Wendet man auf dieses Modell den zweiten Datensatz an, der Corona-Daten und detaillierte Daten über die Straßenbeschaffenheit beinhaltet, dann sinkt das Bestimmtheitsmaß auf 41,16 % im Testset und der RMSE steigt leicht auf 137,43, während im Trainingsset die Werte besser aussehen mit einem Bestimmtheitsmaß von 81,52 % und einem RMSE von 74,67. Das heißt hier ist das Overfitting sogar noch ein wenig größer. Aber welches der beiden Modelle besser ist, hängt auch von der graphischen Kartendarstellung ab, die die Modelle am Ende liefern. Abschließend geklärt wird dies erst im Abschnitt 5.5.

#### 5.4 Neuronales Netz

Im Vergleich zu den vorherigen Methoden sind neuronale Netze die populärste Methode, vor allem bekannt für die verblüffende Präzession ihrer Vorher-

Sets	Train $R^2$	Train RMSE	Test $R^2$	Test RMSE
1	0,9223	162,471	0,069	220,811
2	0,937	376,894	0,005	620,447
3	0,9484	250,083	0,042	243,739
4	0,946	475,38	0,276	662,502
5	0,925	201,328	0,05	280,962
Mean	0,936	293,231	0,089	405,692

Tab. 5.4: Performance des neuronalen Netzes

sagen. Doch verblüffen die neuronalen Netze auch bei der räumlichen Vorhersage des urbanen Radverkehrs? Das hängt vor allem von den vielen Variablen ab, die man bei einem neuronalen Netz einstellen kann. Zum einen kann man festlegen, wie viele Ebenen ein neuronales Netz haben soll und wie viele Knotenpunkte jede Ebene haben darf, die Informationen an die nächste Ebene weiter geben.

Leider ist jeder Durchlauf eines neuronalen Netzes sehr zeitintensiv. Diese Limitation erschwert den Vergleich in der Performance vieler verschiedener Modelle. So dauerte die Durchlaufzeit des Modells aus der Tabelle 5.4 ganze 25,7 Stunden mit einer Anzahl an Beobachtungen von 20000. Würde man die Anzahl an Beobachtungen erhöhen, könnte das neuronale Netz unter Umständen bessere Vorhersagen treffen, dies würde jedoch auch die Durchlaufzeit erhöhen.

Grundsätzlich ist das Problem dieses Modells aber Overfitting. Von allen betrachteten Modellen weißt das neuronale Netz das stärkste Overfitting auf, betrachtet man die Unterschiede zwischen dem  $R^2$ -Wert im Trainingsset, der im Schnitt bei 93 % liegt und im Testset, wo der selbe Wert im Schnitt nur 9% beträgt. Die Differenz könnte nicht größer sein.

Das verwendete neuronale Netz nutzte sechs Neuronenreihen mit je 48, 26, 16, 8, 4 und 2 Neuronen. Hier müsste evaluiert werden, ob dieser Aufbau sich verbessern ließe. Die Anzahl von Trainingsschritten war auf 100000 begrenzt. Ein häufig auftretendes Problem war, dass das neuronale Netz keine Konvergenz zum Schwellenwert für die partielle Ableitung der Fehlerfunktion herstellen konnte, ganz gleich wie viele Trainingsschritte gemacht wurden. Deswegen ist auch der Schwellenwert auf 0.025 hoch gesetzt. Dieser Wert ist

zu hoch, war aber notwendig, um überhaupt Ergebnisse hervor zu bringen. Je Modell wurden drei wiederholte Trainingsansätze genutzt. Diese Anzahl könnte man noch erhöhen, um mehr Generalisierung zu erzielen.

Im Vergleich aller Modelle brachte das Random-Forest-Modell die besten Vorhersagen hervor. Im weiteren Verlauf wird also immer von dem Random-Forest-Modell die Rede sein.

## 5.5 Modellprojektion

In den vorangegangenen Sktionen wurden die Ergebnisse vier verschiedener Machine-Learning-Modelle vorgestellt. Dies waren die Ergebnisse der OLS-Regression, der Support-Vector-Regression, von Random-Forests und die Ergebnisse eines neuronalen Netzes. Dabei hat der Random Forest die Ergebnisse mit der höchsten Plausibilität geliefert, was getestet wurde durch die Cross Validation.

Dieses Modell lässt sich nun dazu nutzen, Vorhersagen auf ein ganzes Straßennetz zu projizieren. Die Daten des Straßennetzes stammen wiederum von Open-Street-Map. Der erste Schritt ist es, einen Kartenbereich auszuwählen mithilfe zweier Koordinaten wie im Quellcode 5.1. Dabei ist anzumerken, je größer dieser Kartenausschnitt sein soll, desto höher ist natürlich auch die Datenlast. Bei großen Stadtgebieten ist es zu empfehlen, nur einzelne Straßentypen auszuwählen, so wie dies in Abbildung 7.11 im Anhang zu sehen ist für Berlin, da Berlin mit seiner Fläche einfach zu groß ist. Alternativ könnte man den Prozess auch in mehrere Ausschnitte aufteilen, um die Ergebnisse für eine große gesamte Karte am Ende zusammenzufügen. Aus zeitlichen Gründen, ist dies nicht passiert. Die Koordinaten werden in myLocation gespeichert. Der hier dargestellte Quellcode 5.1 würde einen Ausschnitt des Ringbereichs in Münster auswählen.

*Listing 5.1:* Wahl des Kartenausschnitts

```

1 myLocation <- c(7.597514856738869, 51.94573812395569,
2                               7.652382675482133, 51.9756143280805)
3
4 q <- myLocation %>%
5 opq() %>%
6 add_osm_feature("highway")
7
8 streets <- osmdata_sf(q)

```

Danach wird eine Anfrage an Open-Street-Map gestellt, die Daten des Typs „highway“ im angegebenen Kartenbereich zu downloaden. Dieser Vorgang ähnelt dem Prozess, wie wir ihn im Abschnitt 3.5 kennen gelernt haben. Darauf folgend müssen die ausgelesenen Daten in einem Format gespeichert werden, das in einen Datensatz für das Modell zur Kartenprojektion verwendet werden kann. Zunächst bestehen die Open-Street-Map Daten aus verschiedenen Straßenlinien, deren Koordinaten in `streets$osm_lines$geometry` gespeichert sind. Jede Straßenlinie besteht aus mehreren und zusammenhängenden Vektoren, und diese Vektoren bestehen je aus zwei Koordinaten. Zunächst müssen die einzelnen Vektoren der Straßenlinien voneinander getrennt werden, damit jeder Vektor mit einem seiner beiden Koordinaten im Datensatz wie eine Zählstation behandelt werden kann. Dabei wird die erste Koordinate in `streetPositions$Lon` und `streetPositions$Lat` gespeichert und die zweite Koordinate in `streetPositions$Lon2` und `streetPositions$Lat2`. Auf Grundlage der ersten Koordinate können dann Vorhersagen getroffen werden. Die zweite Koordinate wird im Datensatz gespeichert, damit die Modellprojektion zwischen beiden Koordinaten eine Linie auf der Karte ziehen kann. Der Quellcode 5.2 zeigt diesen Prozess.

Dabei werden zwei `for`-Loops verwendet.

Listing 5.2: Speicherung der Koordinaten der Straßen

```

1 for (i in 1:length(streets$osm_lines$geometry)){
2   l = length(streets$osm_lines$geometry[[i]])
3
4   for (j in 1:(length(streets$osm_lines$geometry[[i]])/2 - 1)){
5
6     streetPositions$Lon[k]=streets$osm_lines$geometry[[i]][[j]]
7     streetPositions$Lat[k]=streets$osm_lines$geometry[[i]][[l/2+j]]
8
9     streetPositions$Lon2[k]=streets$osm_lines$geometry[[i]][[j+1]]
10    streetPositions$Lat2[k]=streets$osm_lines$geometry[[i]][[l/2+j+1]]
11
12  }
13}
14
15}
16}

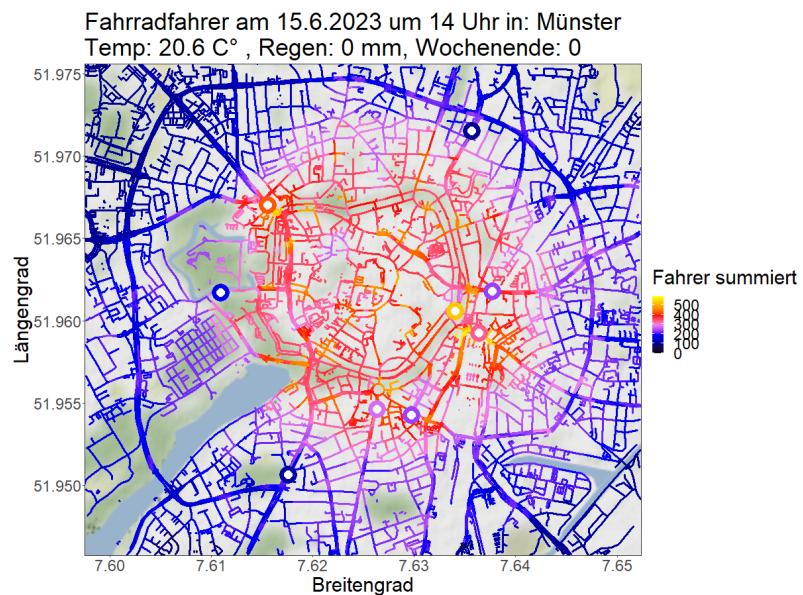
```

Der erste `for`-Loop geht alle Straßenlinien durch. Der zweite `for`-Loop geht alle Vektoren innerhalb der jeweiligen Straßenlinie durch. Innerhalb der zweiten Schleife werden dann die Einzelwerte der jeweiligen Koordinate in dem Dataframe `streetPositions` gespeichert. Dieser Dataframe wird darauffolgend als Grundlage für die Vorhersagen genutzt. Die Variablen die zur Vorhersage benötigt werden, werden ähnlich wie im Kapitel 3 gebildet. Darüber hinaus kann man einzelne Faktoren auch überschreiben, um hypothetische Fälle zu betrachten wie z.B. die Darstellung des Radverkehrs bei verschiedener Wetterlage zum selben Zeitpunkt, wie die Abbildung 7.12 im Anhang zeigt.

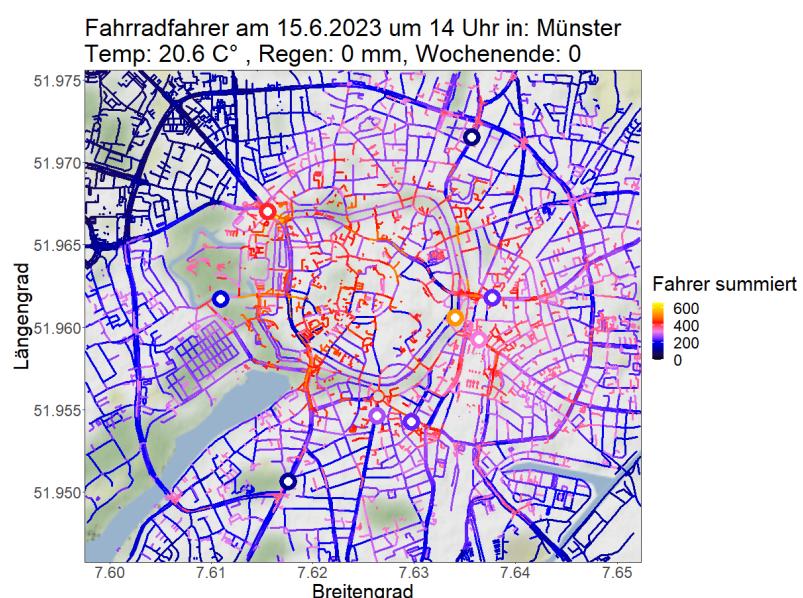
Hat man nun einen solchen Datensatz, der die Vorhersagen des jeweiligen Modells enthält, ist die beste und anschaulichste Präsentation dieser Ergebnisse eine graphische Übersicht des Straßennetzes mit farblicher Markierung der Fahrradauslastung. Eine solche Darstellung wird mit den beiden Ergebnissen der zwei Random-Random-Forest Modelle aus Abschnitt 5.3 berechnet, zu sehen in Abbildung 5.7. Zusätzlich wurden die im Datensatz enthaltenen

Zählstellen als Ringe auf der Karte dargestellt, deren farblich Markierung die durchschnittliche Radauslastung an der jeweiligen Zählstation zeigt.

Wie in der Abbildung 5.7 zu sehen ist, unterscheiden sich beide Modelle sehr stark. Während in der Abbildung 5.7a des ersten Modells zu sehen ist, dass das Aufkommen des Radverkehrs in der räumlichen Vorhersage flächenmäßig relativ gleichverteilt ist, sich aber auch auf das Stadtzentrum konzentriert, ist die Vorhersage des zweiten Modells in der Abbildung 5.7b weniger gleichmäßig. In der zweiten Abbildung ist ein stärkeres Rauschen zu sehen. Dabei ergibt das Rauschen der Vorhersage an manchen Stellen Sinn, an anderen jedoch werden Vorhersagen getroffen, die den Verkehr stark überschätzen. Nützlich am Modell 2 ist, dass bestimmte einzelne Straßenzüge stärker betont werden, vor allem auch spezifische Kreuzungen im Bereich der Ringstraße Münsters. Es fallen z.B auf die Kreuzung Niedersachsenring/Bohlweg, Niedersachsenring/Gartenstraße, Kaiser-Wilhelm-Ring/Warendorfer Straße oder auch die Kreuzung Kolde-Ring/Weseler Straße. Dies sind jedoch auch wichtige Verkehrsadern der Stadt Münster. Dass hier eine hohe Radverkehrsdichte im Bereich der Ring Straßen herrschen muss, scheint trivial. Doch macht das Modell spezifische vorhersagen zur Lage des Verkehrs und dass dies nicht im ganzen Ring Bereich so sein muss. Für den Nordwesten des Ringbereichs sagen beide Modelle eine niedrige Radverkehrsdichte hervor. Ohne weitere Zählstation lässt sich dies nicht abschließend prüfen, wir kennen jedoch auch die gemessene Validität der Modelle. Das bedeutet, die ungefähre Treffsicherheit des Modells ist bekannt, durch die vorgenommene Validierung. Leider ist die Validität nicht hoch genug, um den Vorhersagen blind zu vertrauen. Darüber hinaus macht das zweite Modell für einige Nebenstraßen sehr hohe Vorhersagen. Diese erscheinen nicht plausibel. Eine Möglichkeit wäre es, diese Nebenstraßen und Parkgassen in der Anzeige auszuschließen, so wie es z.B. auch mit Gehwegen und Fußgängerzonen geschehen ist. Die grafische Ausgabe der Modellvorhersagen lässt sich also noch verbessern. Die abweichenden hohen Vorhersagen für Hinterhofstraßen, die in Wirklichkeit nicht genutzt werden, entstehen, weil Open-Street-Map hier den selben Straßentyp „service“ verwendet, der an anderen Stellen auch Radwege bezeichnet. Natürlich stehen nur auf den Radwegen Zählstationen zur Verfügung, nicht



(a) nach Modell 1



(b) nach Modell 2

Fig. 5.7: Räumliche Modellprojektion: Modellvergleich

Sets	Train $R^2$	Train RMSE	Test $R^2$	Test RMSE
1	0,842	72,581	0,357	97,473
2	0,84	69,092	0,508	142,219
3	0,866	63,878	0,282	153,957
4	0,844	67,856	0,56	139,983
5	0,832	69,49	0,538	142,357
Mean	0,845	68,579	0,443	135,198

Tab. 5.5: Performance des dritten RF Modells

aber auf Betriebszugangsstraßen, die auch in Hinterhöfen zu finden sind. So mit kann das Modell an diesen Stellen nur falsche Vorhersagen treffen. In Zukunft müsste auf die graphische Darstellung der Hinterhofstraßen ganz einfach verzichtet werden. Vorerst muss der Betrachter diese ignorieren.

Mit der Absicht, die Validität des Modells noch weiter zu erhöhen, ist ein drittes Modell entstanden, das alte und neue Variablen zum Straßentyp mit in das Modell aufnimmt und kombiniert und zusätzlich noch mehr nicht lineare Effekte und eine größere Stichprobenmenge von 80000 Beobachtung nutzt. Dieses dritte Modell hat in der Berechnung 28 Stunden benötigt. Die Ergebnisse zeigt die Tabelle 5.5 und die Abbildung 5.8. Graphisch ist dieses Modell fast identisch zum zweiten Modell. Da Modell 2 und 3 brauchbarere Vorhersagen machen, weil beide einzelne Straßenzüge stärker hervorheben und weil Modell 3 eine höhere Treffsicherheit als Modell 2 hat und mit mehr Beobachtungen trainiert worden ist, wird für weitere Vorhersagen Modell 3 verwendet.

Die Abbildungen 5.9 und 5.10 zeigen beispielhaft die Verkehrsprognosen für Hamburg, Leipzig, Mannheim und Oberhausen. Es wird offensichtlich, dass das Modell sich dem Verkehrsniveau der Städte anpasst. An manchen Stellen scheint das Modell das Verkehrsaufkommen zu überschätzen, vergleicht man es zu den markierten Stationen. Die markierten Stationen zeigen jedoch nur den Durchschnitt über alle Messungen hinweg an. Da die Modellprojektionen ein Datum in der Zukunft gewählt haben im Sommer, an dem Spitzenwerte auftreten, ist hier ein Vergleich schwierig.

Um das Modell einem besonderen Stress test zu unterziehen, müssen Vorhersagen für eine ganze Stadt gemacht werden, die dem Modell unbekannt ist.

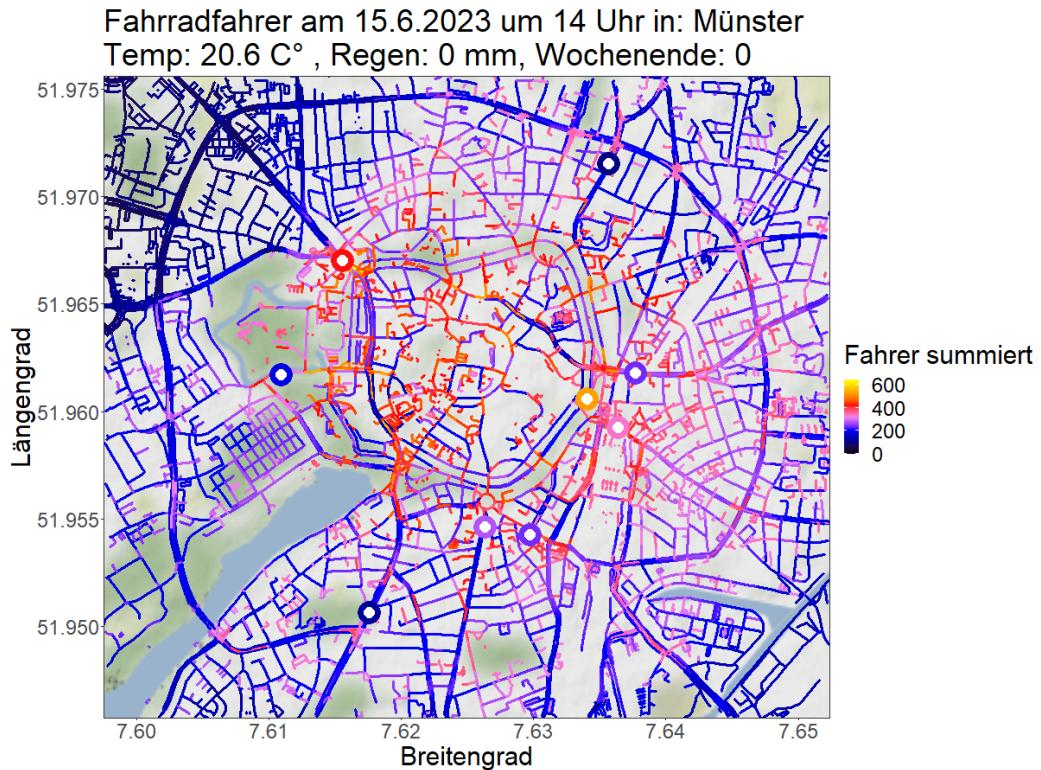
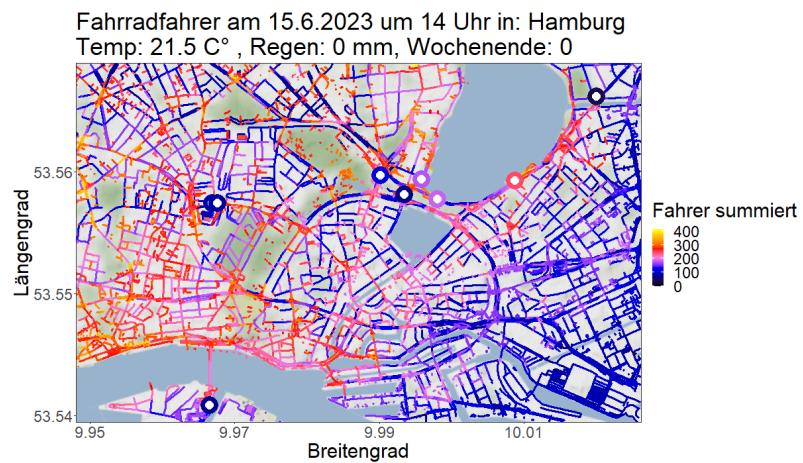
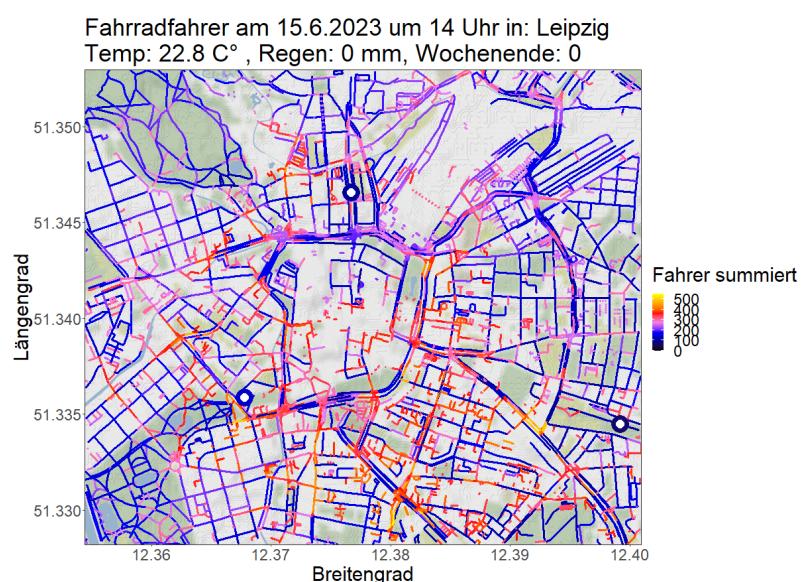


Fig. 5.8: Feature-Selection-Ergebnis der RF Regression in 6 Modellen

Weil im Verlauf der Masterarbeit die Daten aus Dresden erst spät zugänglich waren, konnten diese Daten selbst nicht mit in das Modell einfließen, sind nun aber nützlich für eine zusätzliche Validierung. Diese Daten aus Dresden stammen von 8 verschiedenen Zählstationen von 2018 bis 2021 und umfassen insgesamt 261576 Beobachtungen. Das Modell schätzt hier in Dresden über alle Beobachtungen den Durchschnitt der Radfahrer auf 40.95. Der tatsächliche Durchschnitt entspricht jedoch 83.66 und ist somit doppelt so hoch. Der vorhergesagte Medianwert liegt bei 27.06 und der tatsächliche Medianwert liegt bei 43. Das Modell scheint also den Verkehr in Dresden zu unterschätzen.

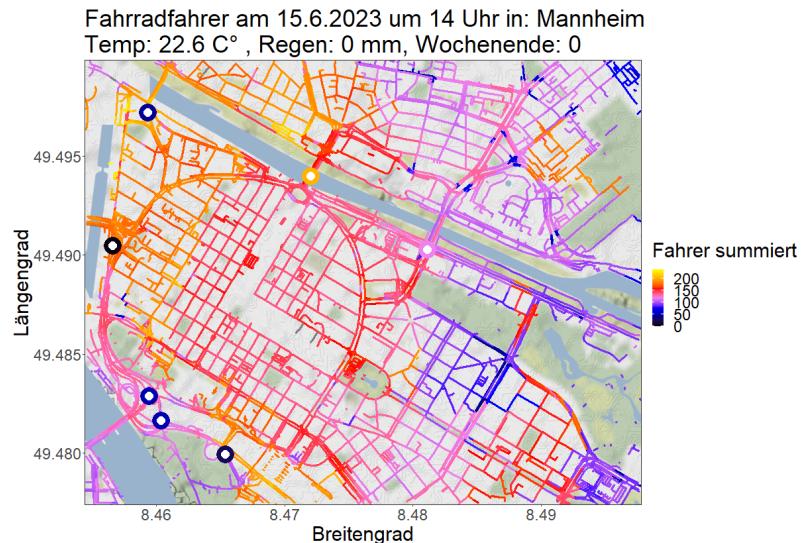


(a) Hamburg

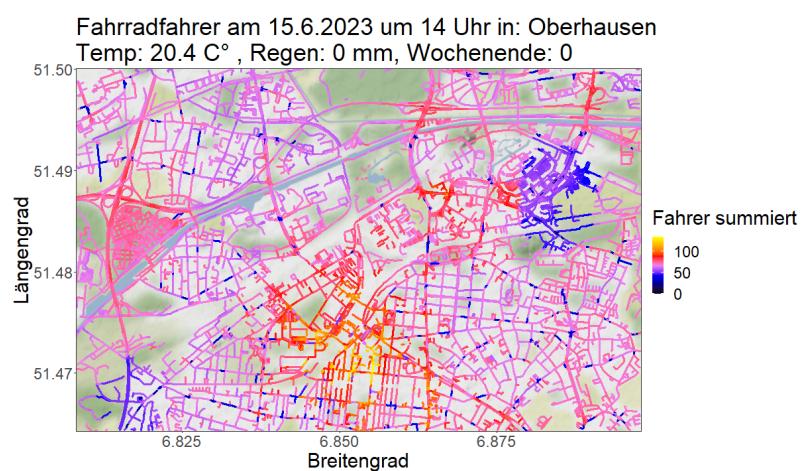


(b) Leipzig

Fig. 5.9: Modellprojektionen in Hamburg und Leipzig



(a) Mannheim



(b) Oberhausen

Fig. 5.10: Modellprojektionen in Mannheim und Oberhausen

Der RMSE für diese Daten liegt bei 115.76, was im Bereich der vorherigen Evaluationen lag, jedoch liegt das Bestimmtheitsmaß  $R^2$  nur noch bei 0.06. Dies sind Hinweise darauf, dass Modellprojektionen in unbekannten Städten unzuverlässig sind. Diese Vermutung bestätigt sich beim graphischen Modeloutput in Abbildung 5.11. Dieser Modeloutput zeigt im Zentrum Dresdens wenig Verkehr, in Randbezirken hingegen mehr und wirkt deswegen unglaublich. In Verbindung mit den Validierungswerten des RMSE und des Bestimmtheitswertes sowie des unterschiedlichen Medians und Durchschnitts in vorhergesagten und tatsächlichen Verkehrsdaten zeigt das Dresdner Beispiel die Grenzen des Modells. Wo die Modellprojektionen für das Modell bekannte Städte glaubwürdig validiert werden konnten, sind die Modellvorhersagen für die unbekannte Stadt Dresden unglaublich.

## 5.6 Modellprognosen

Auch rudimentäre Prognosen lassen sich mit dem Modell erstellen, wenn man Trends der Bevölkerungsentwicklungen als beständig annimmt und auf Grundlage dieser für zukünftige Werte weitere Vorhersagen für verschiedene Jahre berechnet. Die Vorhersagen, die diese Masterarbeit für die Berechnung erstellt, zeigt die Abbildung 5.12 für die Freie- und Hansestadt Hamburg. Natürlich ist das ein sehr einfaches Prinzip zur Stadtentwicklungsprognose, beruhend auf einer einfachen Log-Lin-Regression, die als einzige Variable die Jahre verwendet und so nur eine Trendanalyse betreibt, wie dies Abbildung 5.12a zeigt. Die Trends der letzten 10 Jahre werden so fortgesetzt. Hier könnten bessere Modelle zur Bevölkerungsprognose verwendet werden, dies ist jedoch nicht Thema dieser Masterarbeit. Hier geht es generell nur darum, die Möglichkeiten des Modells aufzuzeigen.

Die aus diesen Vorhersagen resultierende Radverkehrsentwicklung zeigt Abbildung 5.13.

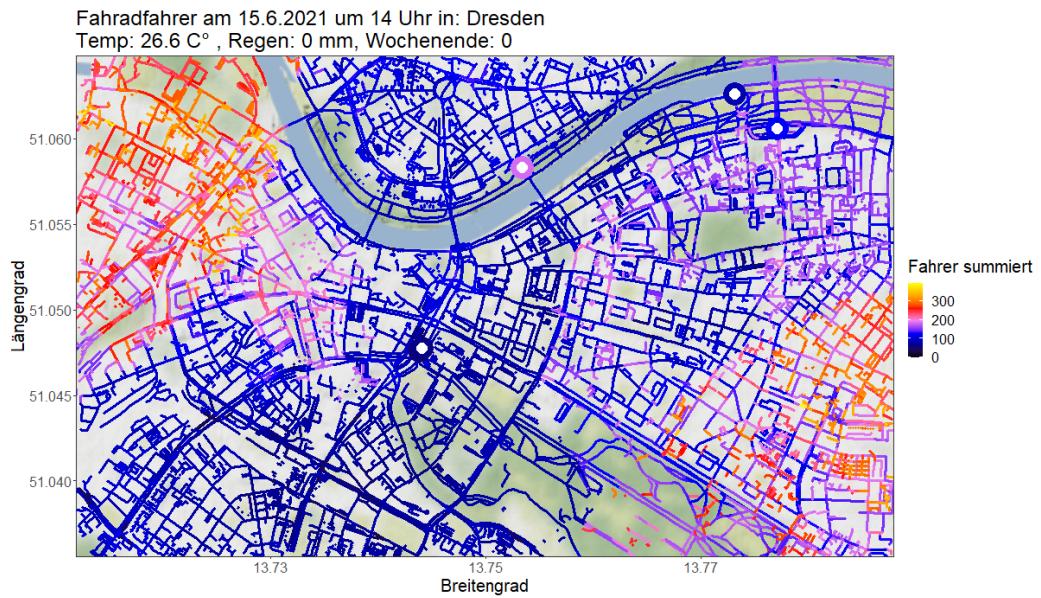


Fig. 5.11: Modellprojektionen in Dresden

## 5.7 Räumliche Korrelation zu Verkehrsunfällen

Auch wenn wie im vorherigen dargestellten Abschnitt die Vorhersagen für unbekannte Städte unbrauchbar waren, ist die Verwendung für die 22 deutschen Städte, die im Modellsatz vorkommen, hilfreich. Eine weitere interessante Frage ist, ob die Modellvorhersagen räumlich mit Radverkehrsunfällen korrelieren. Sollte dies der Fall sein, könnte dies das Modell zum einen weiter bestätigen, aber auch dabei helfen Straßenstellen ausfindig zu machen, die über ihr Verkehrsniveau hinaus gefährlich sind z.B. durch falsches Straßen-

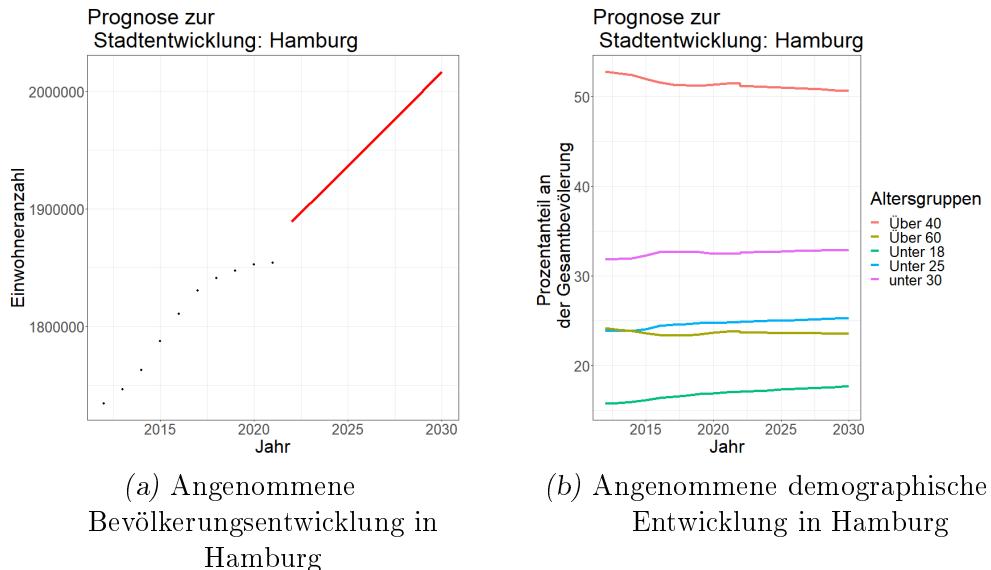


Fig. 5.12: Stadtentwicklungsprognose

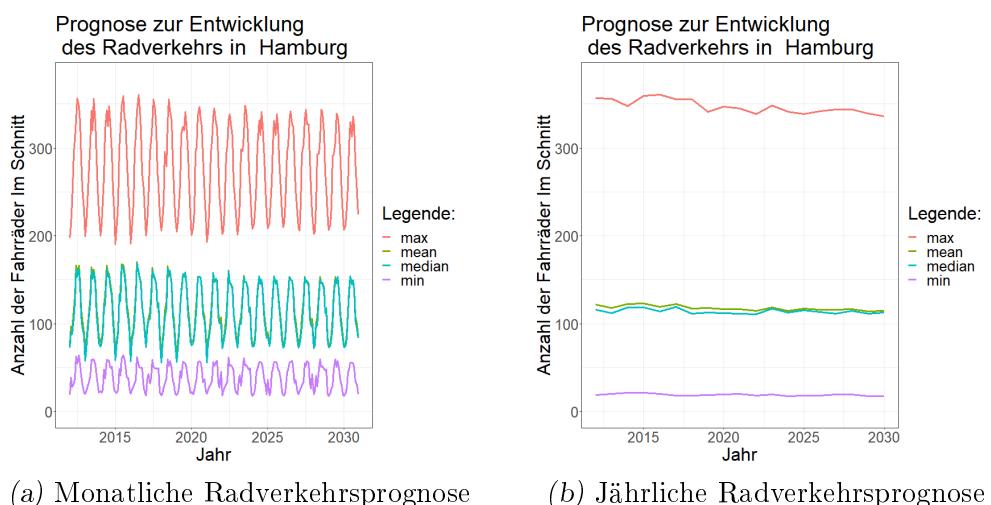
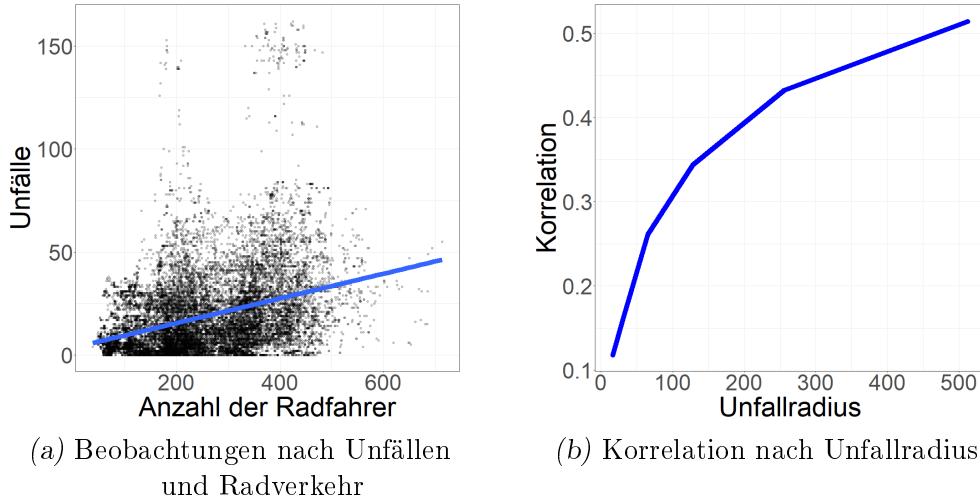


Fig. 5.13: Radverkehrsentwicklungsprognose für Hamburg



*Fig. 5.14: Räumliche Korrelation von Unfällen und Radverkehr*

design.

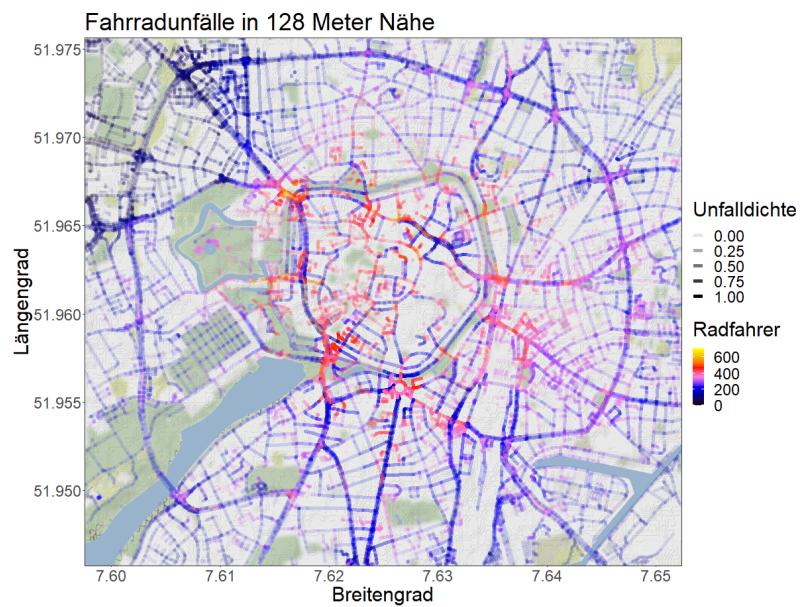
Für den Fall in Münster stellt die Organisation Code-For-Münster<sup>1</sup> Verkehrsunfalldaten zur Verfügung. Diese Daten aus Münster reichen von 2008 bis 2018 und umfassen 6676 Unfälle, bei denen Fahrräder involviert waren und bei denen Koordinaten mit aufgezeichnet wurden. Diese Daten lassen sich clustern, in dem man jeden Punkt im Straßennetz zuordnet, wie viele Unfälle in einem gegebenen Radius stattgefunden haben. Für einen Radius von 64 Metern zeigt dies die Abbildung 5.15b. Nachdem man dem Straßennetz die jeweiligen Unfallzahlen zu gewiesen hat, kann man auf dieses Straßennetz Modellvorhersagen zur Radverkehrsdichte treffen, um die räumliche Korrelation der Radverkehrsvorhersage des Modells mit den Unfalldaten zu messen. Dabei bildet jeder Straßenknotenpunkt eine Beobachtung ab, indem sich die geclusterten Unfalldaten und die Modellvorhersage, die als Referenzdatum auf den 15. Juni 2018 14 Uhr datiert, miteinander verbinden

Diese Beobachtungen sind dargestellt in der Abbildung 5.14a. Dabei zeigt sich ein ansteigender korrelativer Zusammenhang. Beim Clustern der Unfall-

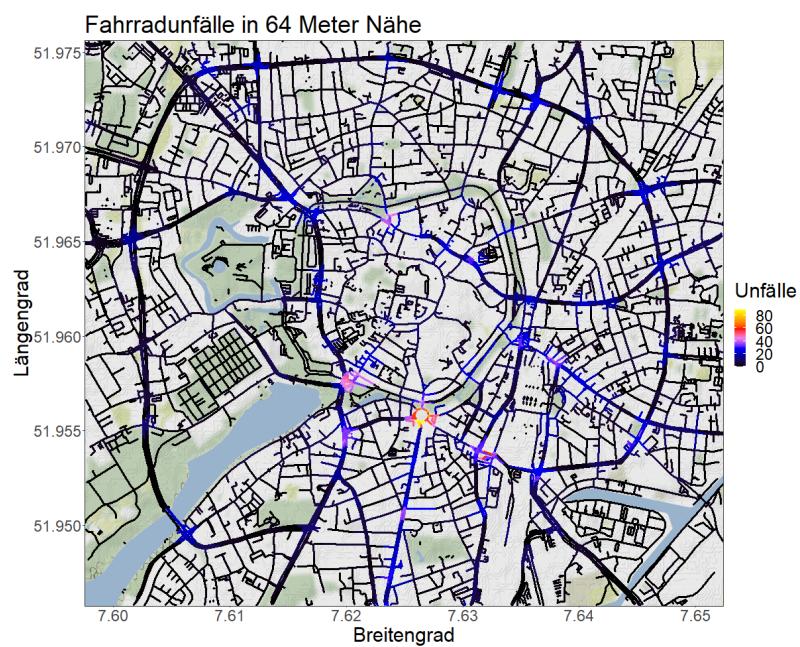
<sup>1</sup> <https://crashes.codeformuenster.org> (Stand: 9.2.2023)

daten lässt sich bestimmen, in welchem Radius um den Straßenpunkt herum die Unfälle kumuliert werden. Dabei lässt sich beobachten, dass je höher dieser Radius ist, desto höher ist die Spearman Korrelation dieser Daten, auch zu sehen in Abbildung 5.14b. Bei einem Unfallradius von 8 Metern beträgt die Spearman Korrelation 0,118 und bei einem Unfallradius von 512 Metern beträgt die Spearman Korrelation 0,514. Diese Korrelationen sind jeweils statistisch signifikant. Dieser Zusammenhang bestätigt das Modell weiter und zeigt anhand der Kurve vielleicht auch, in welcher Auflösung das Modell präzise Vorhersagen trifft. Eventuell kann man daraus ablesen, dass Unterschiede in 100 Meter Entfernung zueinander schlechter vorhergesagt werden als in 500 Meter Entfernung.

Jedoch zeigt die Karte 5.15a auch Abweichungen. Diese Karte zeigt farblich markiert die Radverkehrsdichte, die das Modell vorhersagt, und markiert durch die Transparenz der Straßenlinien die Unfalldichte des Radverkehrs. An vielen Stellen korreliert beides, wie z.B. am Ludgeri Kreisel südlich der Altstadt. Dies ist die Stelle in Münster mit den meisten Radunfällen und gleichzeitig weißt diese Stelle auch einen hohe Radverkehr in der Vorhersage auf. Jedoch zieht sich eine weitere hohe Unfalldichte über die Länge der Hammerstraße hinweg, die vom Ludgerikreisel in Richtung Süden wegführt. Das Modell sagt hier aber eine geringe Verkehrsdichte vorher. Auch im Nordosten im Bereich der Ringstraße stehen Unfalldichte und Radverkehrsdichte in Dissonanz. Dafür kann es zwei Erklärungen geben. Entweder trifft das Modell falsche räumliche Vorhersagen an diesen Stellen, oder aber diese Verkehrsstellen, sind besonders gefährlich und führen auch mit weniger Verkehr zu vielen Verkehrsunfällen.



(a) Radverkehr und Unfälle



(b) Unfallkarte

Fig. 5.15: Karten zu Unfällen und Radverkehr

## 6. DISKUSSION

Die Forschungsfrage dieser Masterarbeit war es, ob es möglich ist, ein räumliches Modell zu entwickeln, das für ein vollständiges Straßennetz oder für ausgewählte flexible Knotenpunkte zu bestimmten Zeiten Vorhersagen zum Fahrradverkehr machen kann? Das Resultat dieser Masterarbeit ist das zuvor viel beschriebene Modell, das mithilfe einer Random-Forest-Regression entstanden ist. Um die Forschungsfrage zu beantworten, muss der Wert des Modells diskutiert werden und so, wie es die Absicht der Masterarbeit war, im besonderen Bezug zur kommunalen Infrastrukturplanung und Unfallvermeidung betrachtet werden.

Der schwierigste Aspekt in der Modellierung war die räumliche Interpolation einzelner Zählstationen, von denen es nur begrenzt viele gibt. Ein grundsätzliches Problem dabei ist, dass Zählstationen durch Kommunen oft nicht zufällig gesetzt werden, sondern mit Absicht an Stellen, wo ein hoher Verkehr erwartet werden kann. Dies verletzt aber die Annahme von unabhängig und gleich verteilten Beobachtungen. Dies ist der Grund für das massive Overfitting, welches den Modellen allen zu Grunde liegt. Jedoch ließ sich durch eine auf die räumliche Lage der Stationen konditionierte Cross Validation dieses Overfitting aufdecken. Der Wert, der für die externe Validität bemessen wurde, steht bei einem Bestimmtheitswert von 44,3 % und übertrifft knapp die Validität bei Alattar et al. (2021). Mit einem Bestimmtheitswert von durchschnittlich 84,5 % in den Trainingssets erzielt das Modell auch ähnliche Ergebnisse wie bei Studien, die keine räumliche Interpolation betrieben haben, wie bei Holmgren et al. (2017), Broucke et al. (2019) oder auch Wessel (2020). Daraus wird klar, dass die räumliche Interpolation die größte Herausforderungen darstellt. Ein Random-Forest-Modell, das mit den selben Daten und Variablen erstellt wird, aber nicht mittels einer auf Stationen bedingten

Cross Validation validiert wird, sondern mit einer herkömmlichen Cross Validation, erzielt einen Bestimmtheitswert im Testset von 83,4 % und einen RMSE im Testset von 54,1, wobei die Werte im Trainingsset ebenso hoch sind. Zeitliche Unterschiede sind somit perfekt erklärbar, während räumliche Unterschiede zu einem massivem Overfitting führen. Die Performance, die das räumliche Modell dieser Masterarbeit erreicht, ist so niedrig, weil es versucht eine räumliche Vorhersage zu treffen, und dabei eben nur eine mittelmäßige Treffsicherheit erzielen kann.

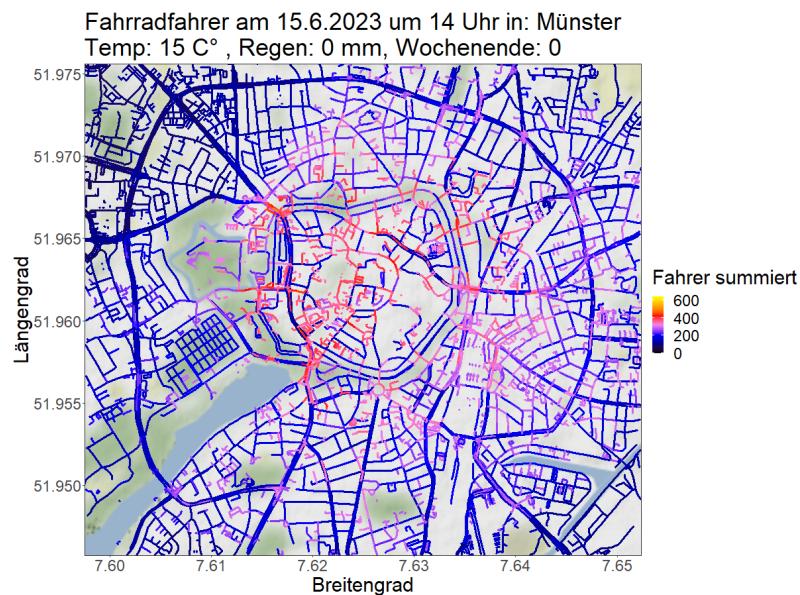
Die übrig bleibende Frage ist, ob die erzielte mittelmäßige Performance der Anwendung zur Verkehrsplanung und zur Unfallvermeidung ausreicht. Auszuschließen ist die Verwendung für Städte, die nicht im Datensatz vorhanden sind. Das heißt zur Vorhersage braucht das Modell eine Anzahl von Zählstellen innerhalb der Stadt, die man betrachten möchte, um schlüssige Vorhersagen zu treffen, wie das Beispiel aus Dresden gezeigt hat. Davon abgesehen liefert das Modell jedoch schlüssige Antworten darauf, wie sehr sich der Verkehr in einzelnen Stadtteilen unterscheidet und welche Kreuzungen am stärksten frequentiert werden und zeigt somit auch den eventuellen Bedarf an Radverkehrsinfrastruktur. Auch ein räumlicher Abgleich von Verkehrsunfällen kann zeigen, ob einzelne Hotspots für Fahrradunfälle mit dem Radverkehr korrelieren, oder einfach nur signifikant gefährlich sind. Solche Hinweise müssen jedoch einzeln geprüft werden. Darüber hinaus zeigte sich bei der räumlichen Korrelation zu Verkehrsunfällen, dass je größer der Radius war, in dem man die Verkehrsunfälle um einen Straßenpunkt herum aufsummerte, desto stärker stieg die Korrelation an. Bei diesem Anstieg war ein abnehmender Grenzeffekt festzustellen. Dies kann zeigen, in welchem Radius das Modell räumliche Unterschiede im Verkehrsaufkommen von Fahrradfahrern erklären kann. Für das Beispiel aus Münster betrug dieser Radius ca. 500 Meter. Unterschiede im Verkehrsaufkommen von Fahrrädern an Positionen, die weniger weit entfernt sind, scheinen weniger valide.

Um in einem engeren Radius bessere Vorhersagen zu treffen, bedarf das Modell einer Weiterentwicklung und zeigt dafür Problematiken auf. Was z.B. nur begrenzt möglich ist, ist eine Vorhersage beruhend auf der Veränderung der Infrastruktur, die dem Modell als Daten zugrunde liegen. Im Modell sind

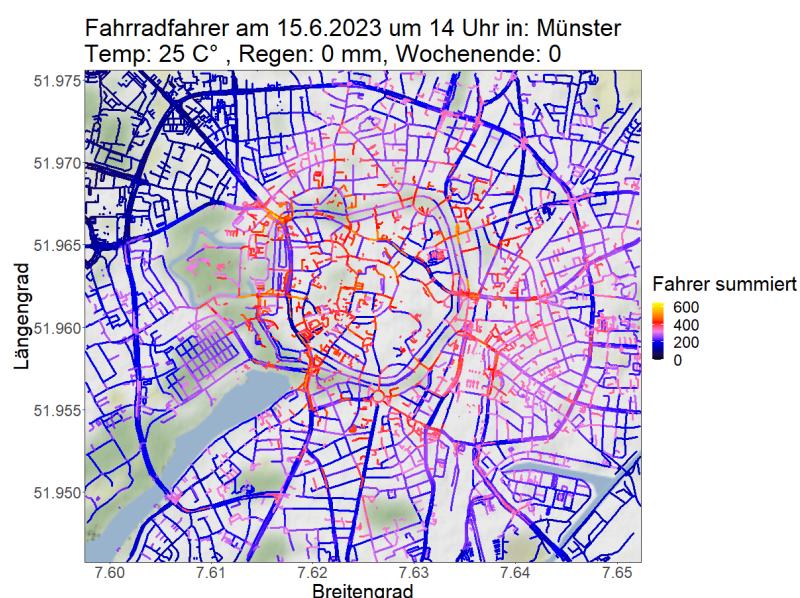
Variablen wie die Anzahl von Geschäften und Stationen des öffentlichen Nahverkehrs in der Umgebung der Straßenpunkte enthalten. Dabei wird es sich aber um rein korrelative und nicht kausale Zusammenhänge handeln, weil diese Variablen selbst auf die Verdichtung des Verkehrs hinweisen. Ändert man also die Anzahl an Busstationen in einem Bereich, hat dies im Zweifel keine Auswirkungen auf den Radverkehr, nur weil der Radverkehr ähnlich verdichtet ist wie der öffentliche Nahverkehr im Zentrum einer Stadt und die Veränderung im Nahverkehr kurzfristig nichts an der Verdichtung im Stadtzentrum selbst ändert.

Kausale räumliche Einflussfaktoren sind noch nicht ausreichend genug untersucht. Das Modell kann nur die Vorhersage liefern, nicht aber die Erklärung für räumliche kausale Zusammenhänge. Dies war jedoch auch nicht zwingend das Ziel dieser Masterarbeit. Ändern könnte man dies z.B., wenn man auf die Veränderungen in der Umgebung von Zählstationen einginge. Fragen die man hier betrachten könnte, wären z.B., wie die Errichtung zusätzlicher Ampeln und Geschäfte wie auch Schulen den Verkehr an bestimmten Zählstationen ändert. Nachteil an den Daten von Open-Street-Map ist leider, dass sie jeweils nur am Tag der Erhebung aktuell sind. So ist nicht hinreichend vermerkt, von wann bis wann bestimmte POIs aufgestellt waren. Was die bestehende Forschungsliteratur aber hergibt sind die kausalen Zusammenhänge von Wetter und Fahrradverkehr wie z.B. bei Wessel (2020). Hier ist das Modell sehr wohl in der Lage zuverlässige Antworten zu fördern, wie in Abbildung 6.1 zu sehen. Auch Unterschiede von Wochenende und Feier- wie auch Ferientagen hebt das Modell deutlich und zuverlässig hervor.

Was sind also Erkenntnisse, die Kommunen mitnehmen können, sollten sie vorhaben, selber Vorhersagen zu machen beruhend auf den Erkenntnissen dieser Masterarbeit? Sollte eine Kommune ein räumliches Modell erstellen wollen, dann ist eine solide Datengrundlage notwendig, deren Beobachtungen hinreichend räumlich verteilt ist. Dabei muss im Besonderen beachtet werden, dass die Verteilung der Daten unabhängig sein muss von eventuellen Faktoren. Wenn nur Zählstation an großen Verkehrsknotenpunkten eingerichtet werden, oder nur im Zentrum der Stadt, dann lassen sich darüber hinaus keine Vorhersagen machen. Wo hier in dieser Masterarbeit das Modell Daten



(a) Kalters Wetter



(b) Warmes Wetter

Fig. 6.1: Modellvorhersagen nach Wetter

von Open-Street-Map verwendet, haben Kommunen meist einen Zugang zu Geodaten, die deutlich zuverlässiger sind und weiter zurück reichen. Mit diesen Geodaten ließen sich zeitliche Veränderungen in der Infrastruktur auch berücksichtigen, die in diesem Modell keine Beachtung fanden.

All diese Maßnahmen könnten das Overfitting minimieren. Die Kombination mit anderen Daten könnte das entwickelte Modelle zusätzlich validieren. In dieser Masterarbeit dienten hier Unfalldaten aus Münster dazu als Beispiel. Aber auch der Abgleich mit der räumlichen Korrelation von Modellvorhersagen und GPS-Tracking-Daten könnten eine Möglichkeit sein, das Modell zu konkretisieren. Als Datenquelle könnte hier die DB Rad+ App der deutschen Bahn dienen. Diese App hat nicht dieselben Probleme wie die Apps im Kapitel 2.1.3 zu GPS und Handy Daten, da diese App sich nicht ausschließlich an Sportler wendet, sondern alle Fahrradfahrer mit Rabatten lockt. An dieser Aktion nehmen bereits 14 deutsche Städte teil, darunter auch Hamburg. Ein noch besserer Schachzug wäre es, die Daten der DB Rad+ App in das Modell als zusätzliche Variable aufzunehmen, um so die Information über häufig gewählte Fahrradrouten in den Modellvorhersagen widerzuspiegeln und die Daten der App auf das gemessene Niveau zu skalieren. Alattar et al. (2021) hatten bereits gezeigt, wie man GPS-Daten auch mit Daten von Zählstationen verbinden kann. Daten aus der App in Hamburg sind im Geo Portal der Stadt Hamburg<sup>1</sup> online zugänglich.

Die Fähigkeiten des Modells lassen sich rein graphisch nur begrenzt darstellen, da das Modell nicht nur räumlich, sondern auch zeitliche Vorhersagen treffen kann. Einen Eindruck dessen vermittelt das Video<sup>2</sup>, das im Zuge dieser Masterarbeit mit Vorhersagen des Modells erstellt worden ist. Alle R-Skripte, die verwendet worden sind zur Erstellung der Modelle, sind in einem für die Masterarbeit eigens erstelltem GitHub-Repository<sup>3</sup> zugänglich.

---

<sup>1</sup> <https://geoportal-hamburg.de>, letzter Zugriff: 1.3.2023

<sup>2</sup> <https://www.youtube.com/watch?v=ek0lsjMu0fc>, letzter Zugriff: 1.3.2023

<sup>3</sup> [https://github.com/MaxWeinhold/Masterthesis\\_BikeTrafficForecast](https://github.com/MaxWeinhold/Masterthesis_BikeTrafficForecast), letzter Zugriff: 2.3.2023

### 6.1 Fazit

Das Ziel dieser Masterarbeit war es, ein Fahrradverkehrsmodell für Städte zu entwickeln, das Antworten geben kann auf die räumlichen und zeitlichen Unterschiede des Aufkommens von Fahrrädern im urbanen Verkehr.

Als Datengrundlage für ein solches Modell dienten Daten von Fahrradzählstationen in 22 deutschen Städten, wie auch Daten über die Demographie der Städte, den Autobesitz, Daten zur räumlichen Verteilung von Infrastruktur wie Schulen, Universitäten, Geschäften oder Ampeln, Daten zur Beschaffenheit von Straßen, Wetterdaten und die Corona-Inzidenz. Mit diesen Daten wurden vier verschiedene Modelle trainiert. Ein Modell wurde mittels der OLS-Regression, der Support-Vector-Regression, mittels Random Forests und einem neuronalem Netz erstellt. Dabei hatte das Random Forest Modell die höchste Out-Of-Sample-Validität. Dennoch waren alle vier Modelle von einem hohem Maß an Overfitting betroffen. Am höchsten war das Overfitting bei dem neuronalem Netz.

Weiterhin sind die Vorhersagen des finalen Modells für Städte, die selbst nicht im Datensatz beinhaltet sind, nicht valide, wie der Vergleich mit Daten aus Dresden zeigte. Jedoch ändert das nichts daran, dass die Vorhersagen für die Städte, die im Datensatz beinhaltet sind, einen interessanten Einblick geben und teils zielgenaue Hinweise darauf geben, in welchen Stadtteilen und an welchen Kreuzungsbereichen der Radverkehr verdichtet ist. Mit Blick auf die Ergebnisse der Validierung ist es also möglich, ein räumliches und zeitliches Modell zur Vorhersage des urbanen Radverkehrs zu berechnen. Auch kann der räumliche Abgleich mit Radverkehrsunfällen Aufschluss darüber geben, welche Straßen und Kreuzungen über ein hohes Verkehrsaufkommen hinaus ein hohes Unfallrisiko mit sich bringen. Jedoch gibt es gleichzeitig noch Verbesserungspotential für das Modell.

Das Hauptproblem, das Overfitting, könnte noch weiter minimiert werden, durch eine sorgfältigere Auswahl darüber, an welchen Orten man Verkehrszählungen unternimmt. Hierbei ist der Charakter zufälliger und unabhängiger Stichproben entscheidend. Weiterhin müssen Variablen über die räumliche Verteilung von Infrastrukturpunkten kausal evaluiert werden. Außerdem bö-

te es sich an, die Daten der DB Rad+ App als weitere Variable im Modell zu verwenden.

Mit diesen Schritten ließen sich ausreichend valide Vorhersagen zum urbanen Radverkehr machen. Die gestellte Forschungsfrage kann deswegen trotz der schwierigen Bedingungen bejaht werden.

## 7. ANHANG

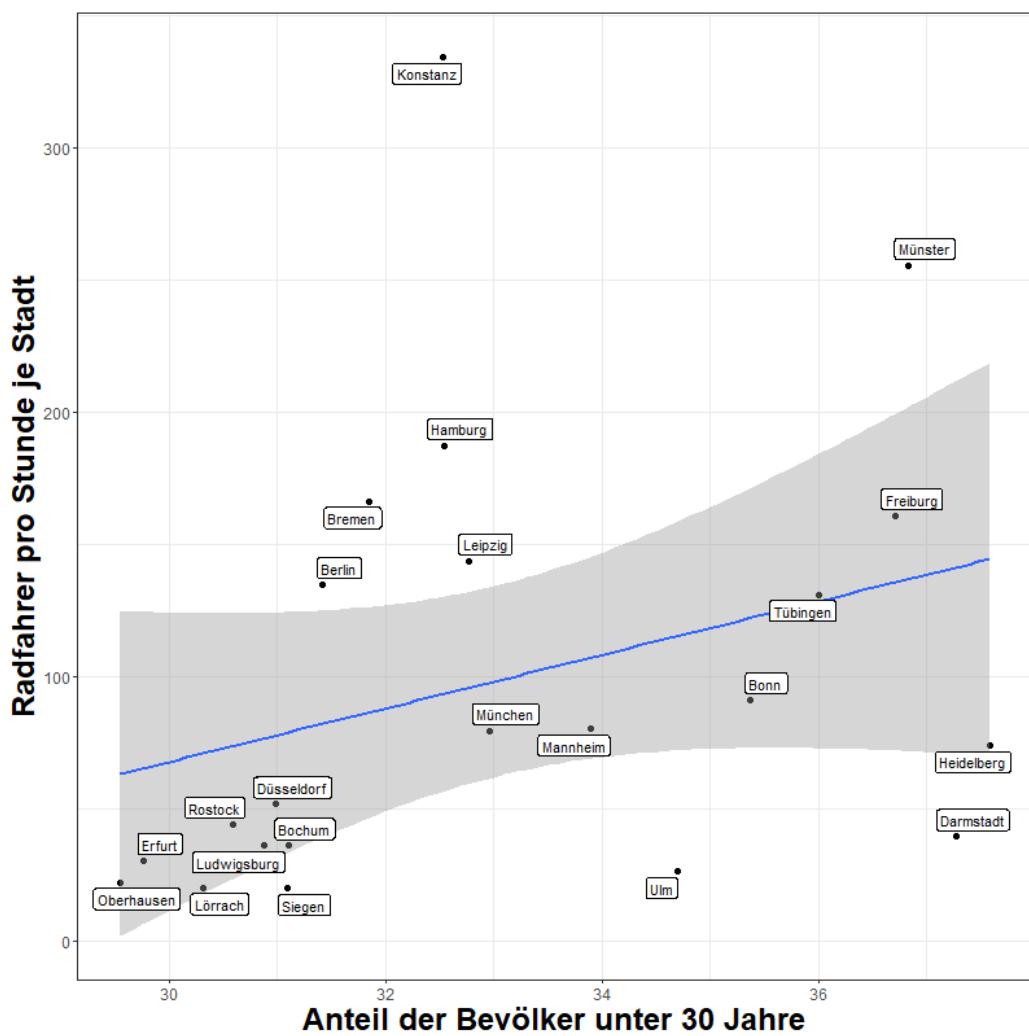


Fig. 7.1: Verteilung des Fahrradaufkommens nach Alter

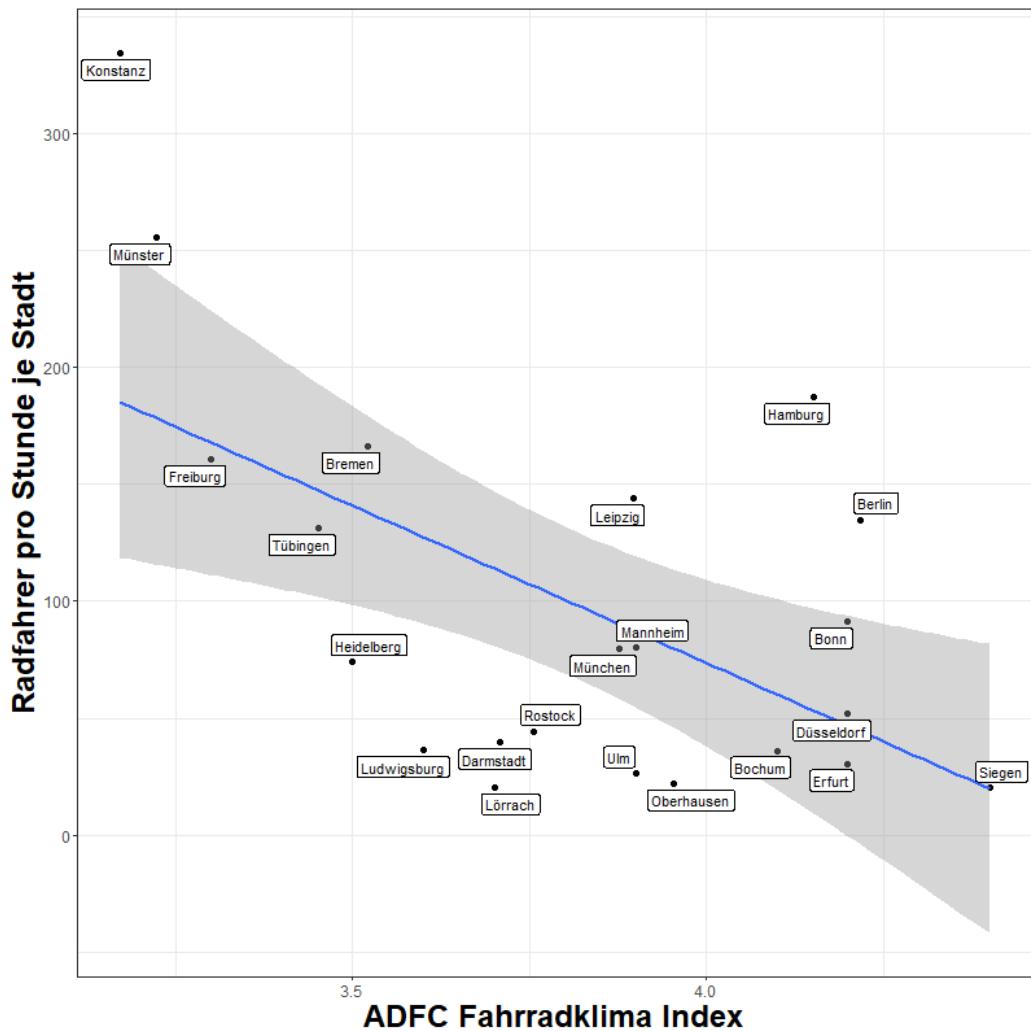


Fig. 7.2: Zusammenhang von Fahrradklima und Radverkehr

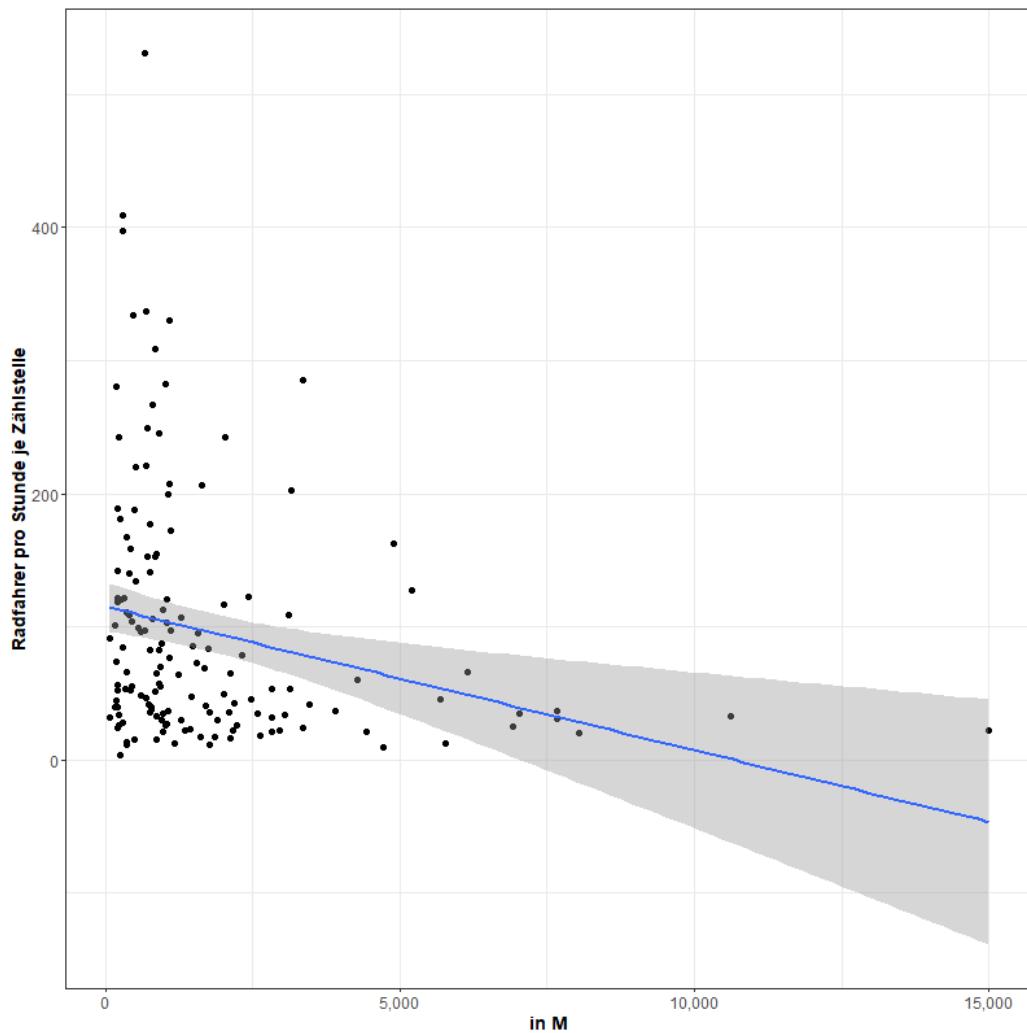


Fig. 7.3: Zusammenhang von Anzahl der Uni-Gebäude in einem 500 M Radius und Radverkehr

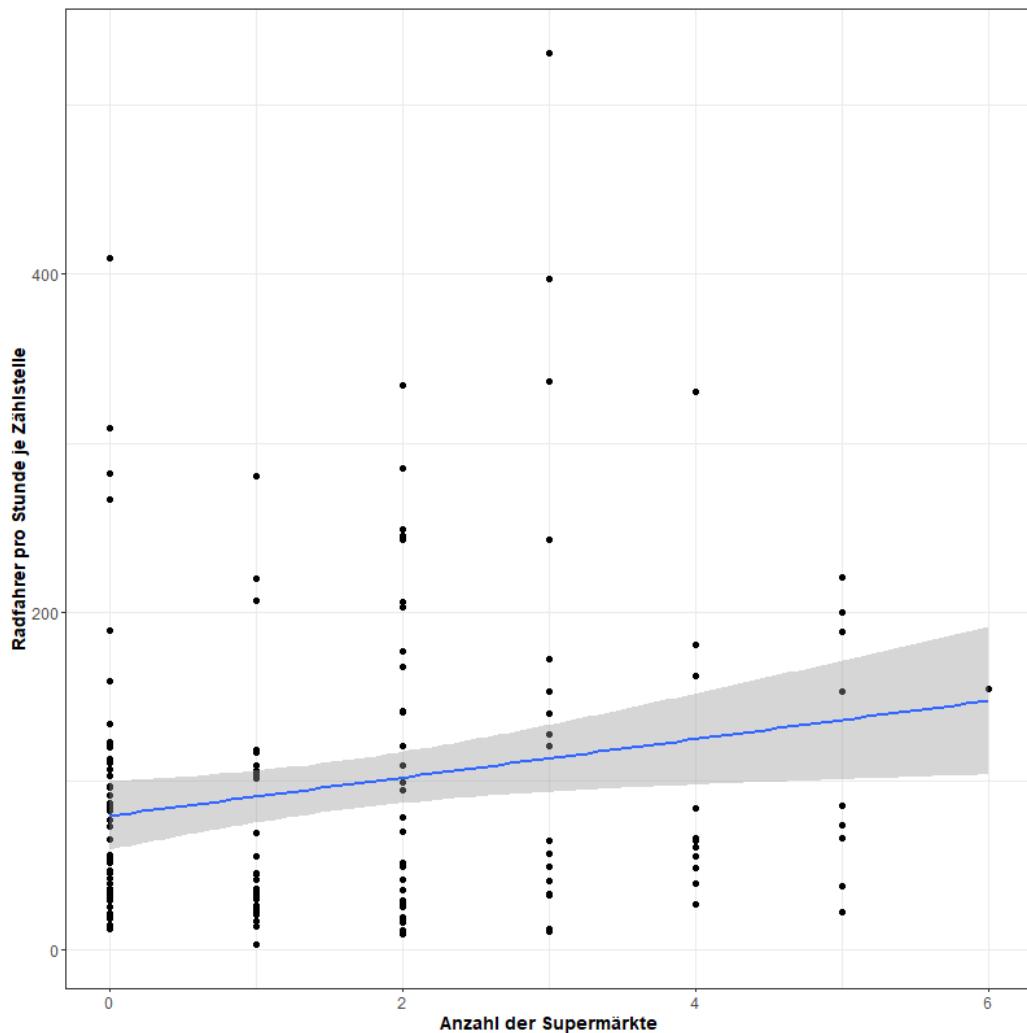


Fig. 7.4: Zusammenhang von Anzahl der Supermärkte in einem 1 km Radius und Radverkehr

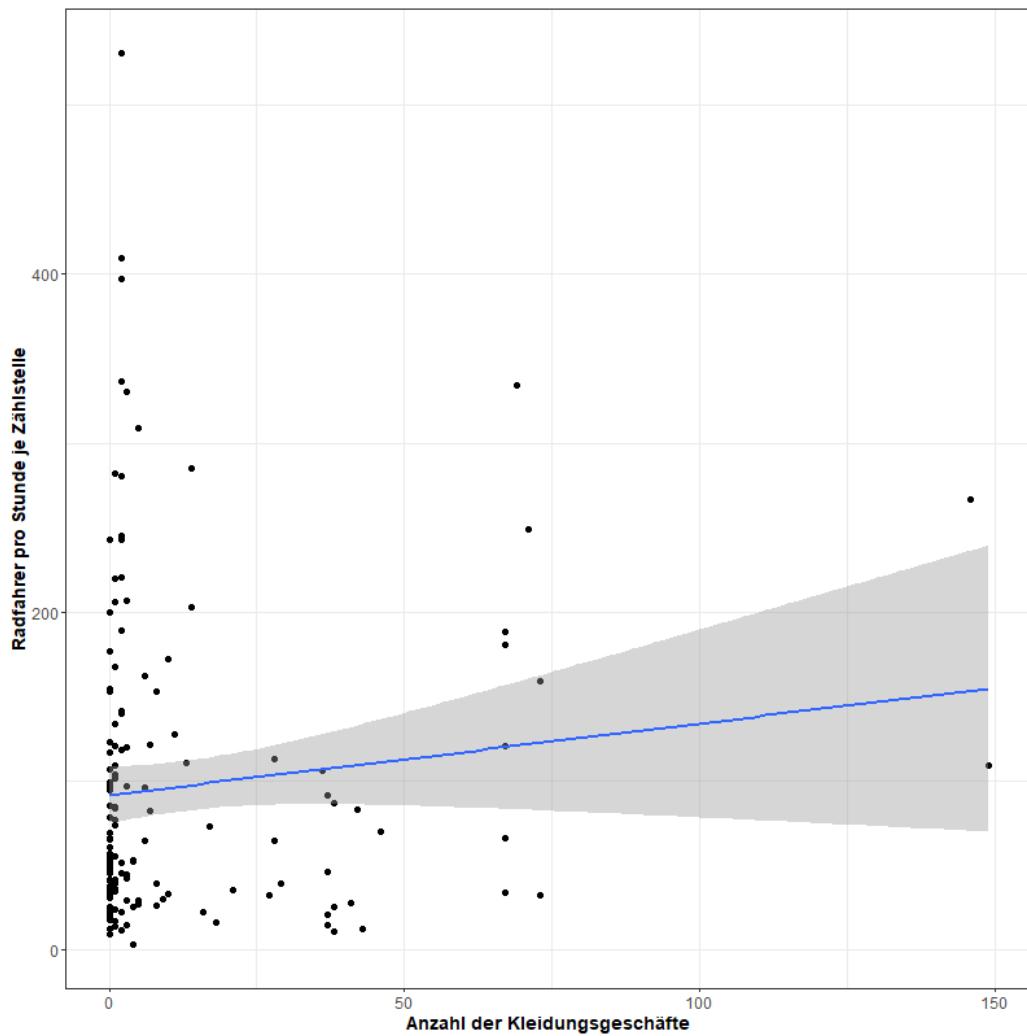


Fig. 7.5: Zusammenhang von Anzahl der Kleidungsgeschäften in einem 2 km Radius und Radverkehr

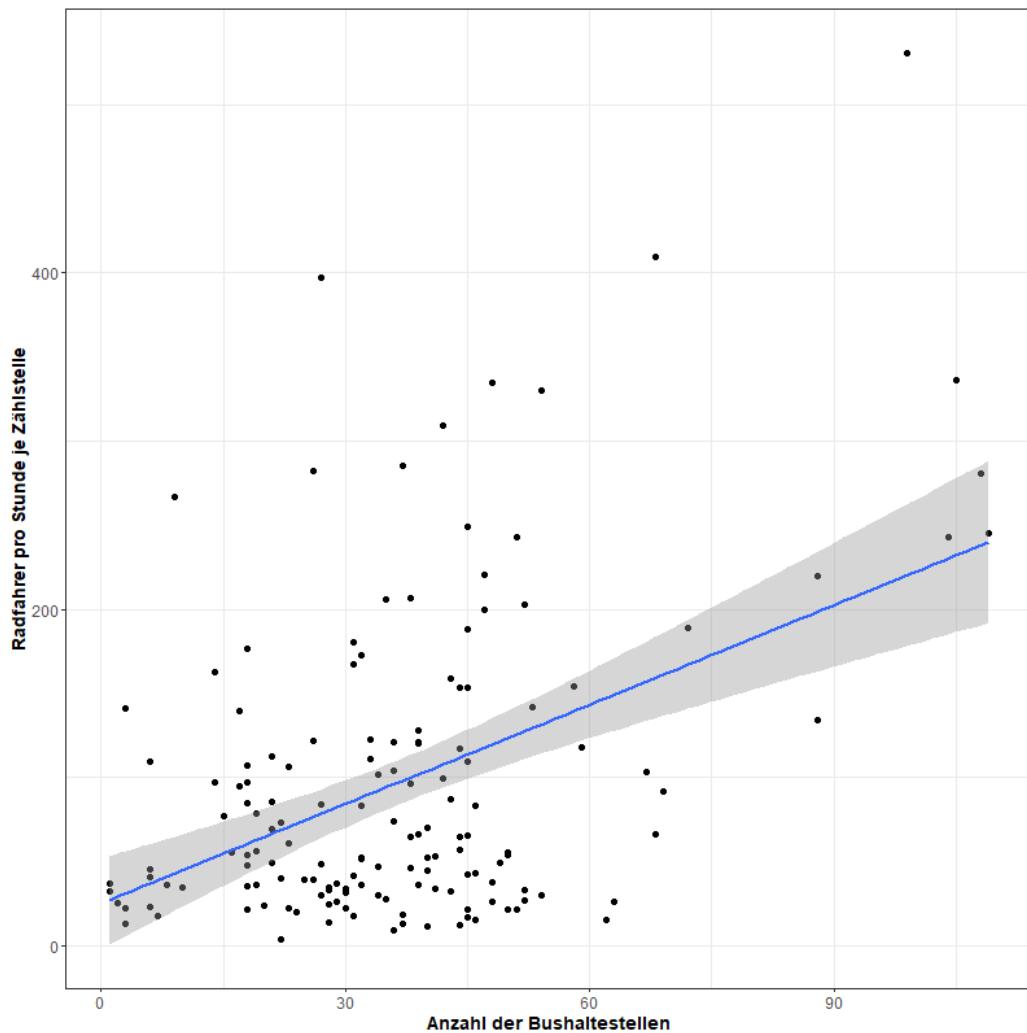


Fig. 7.6: Zusammenhang von Anzahl der Busstationen in einem 1 km Radius und Radverkehr

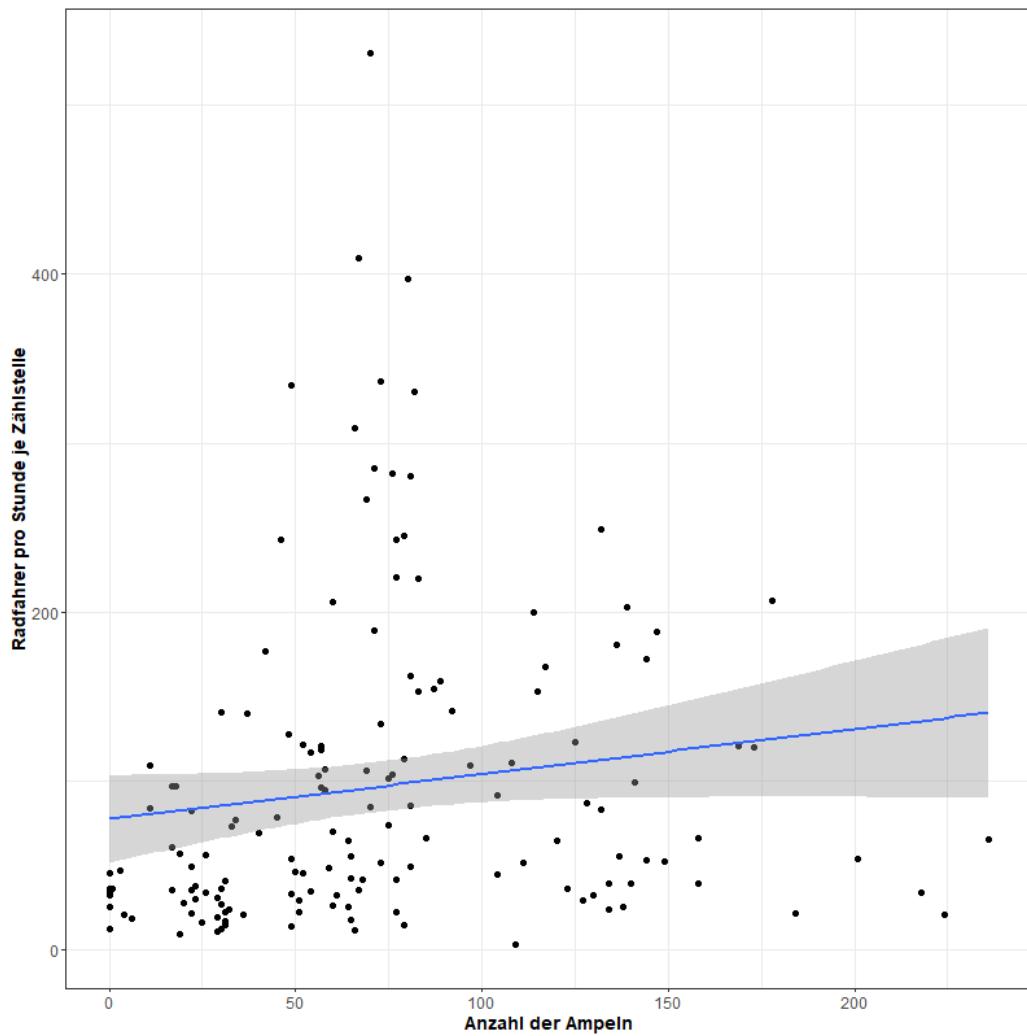


Fig. 7.7: Zusammenhang von Anzahl der Ampeln in einem 1 km Radius und Radverkehr

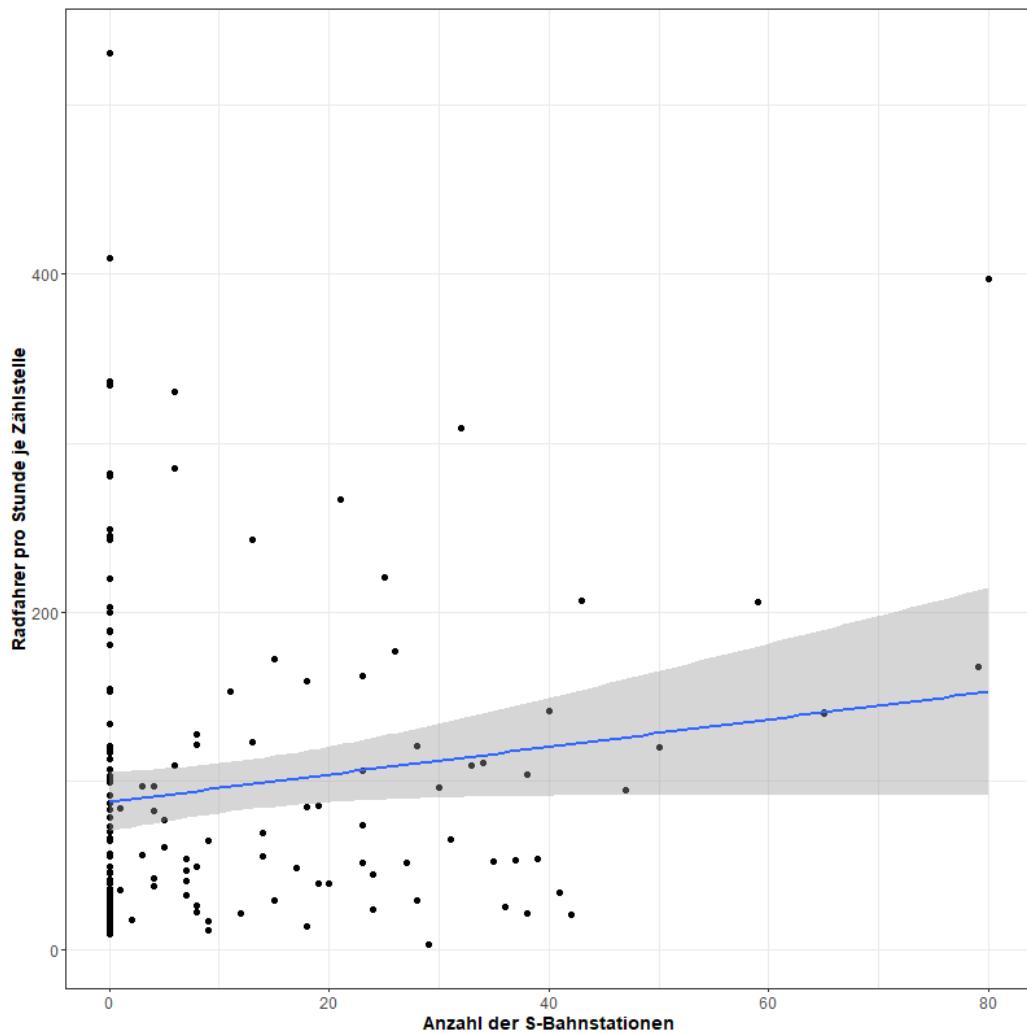


Fig. 7.8: Zusammenhang von Anzahl der Straßenbahnstationen in einem 1 km Radius und Radverkehr

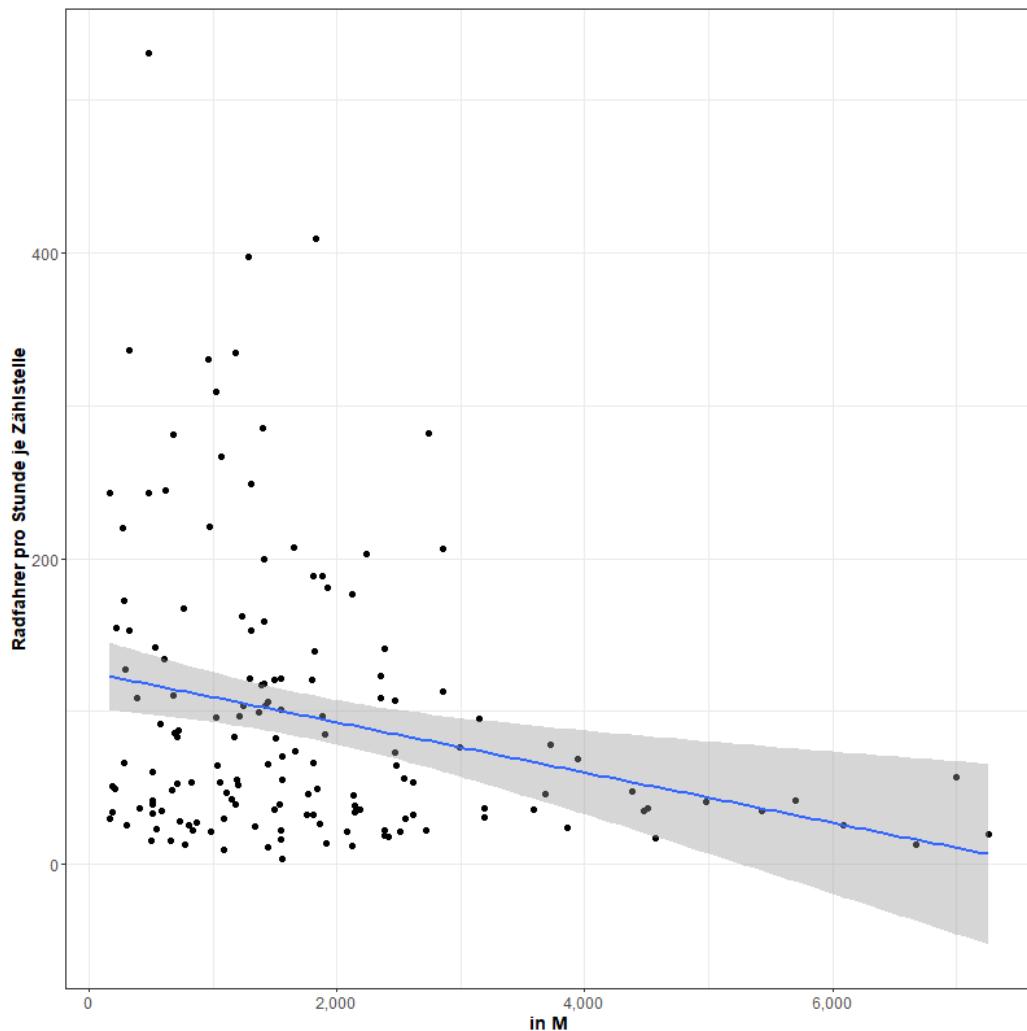


Fig. 7.9: Zusammenhang des nächsten Bahnhofes und dem Radverkehr

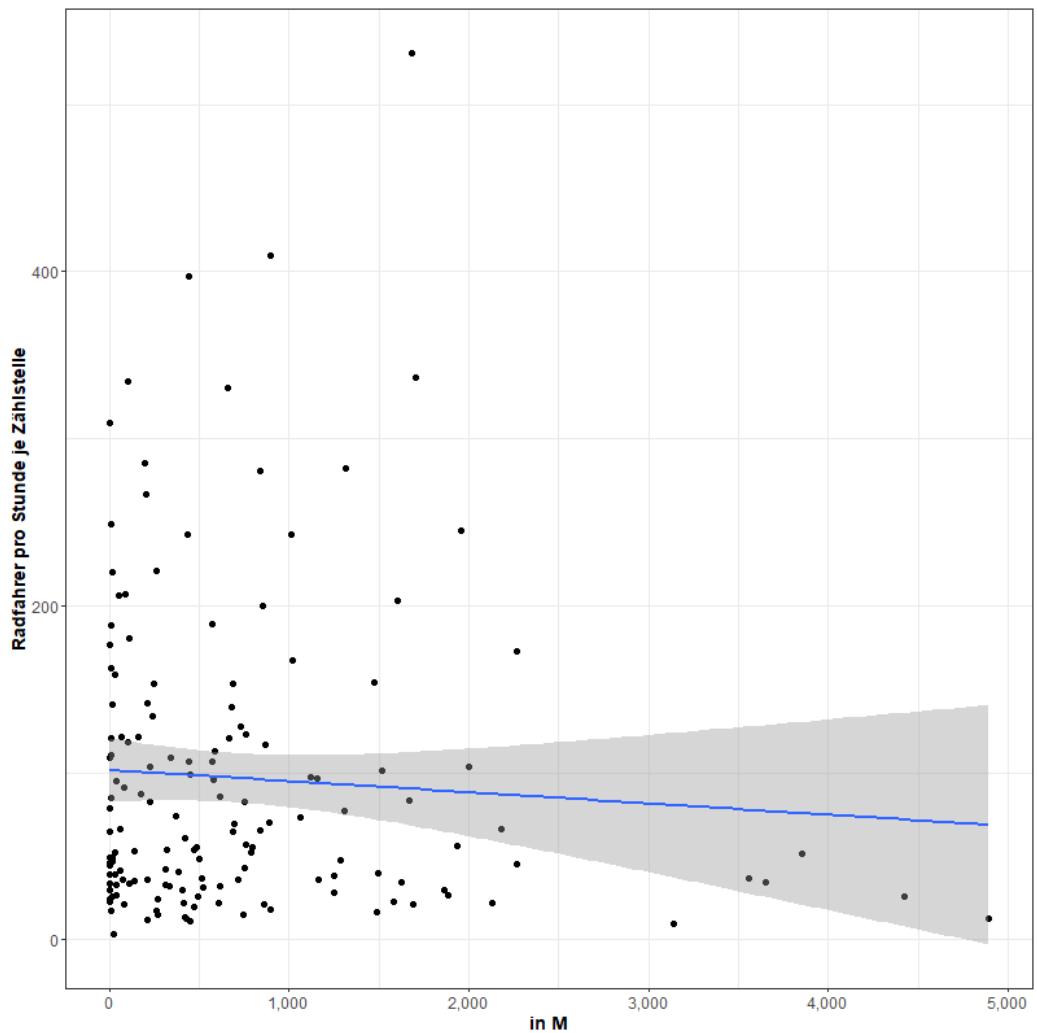
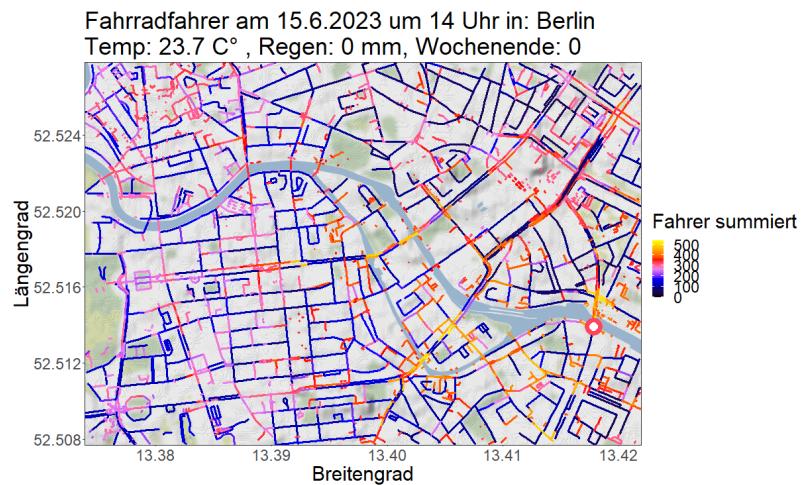
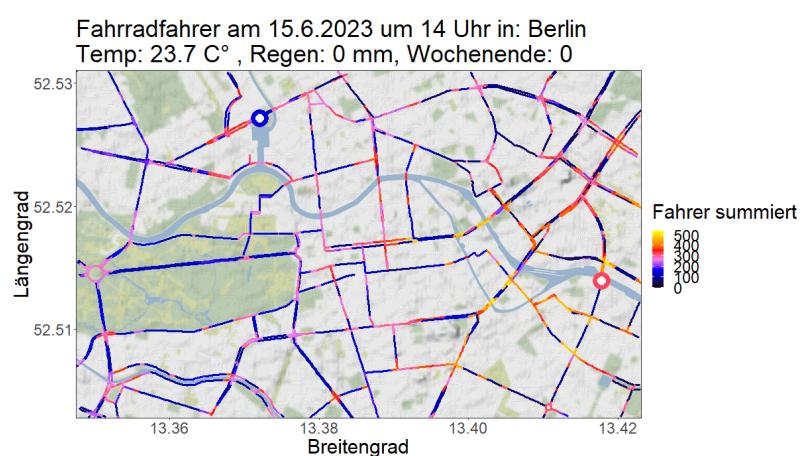


Fig. 7.10: Entfernung zur nächsten Brücke und Radverkehr

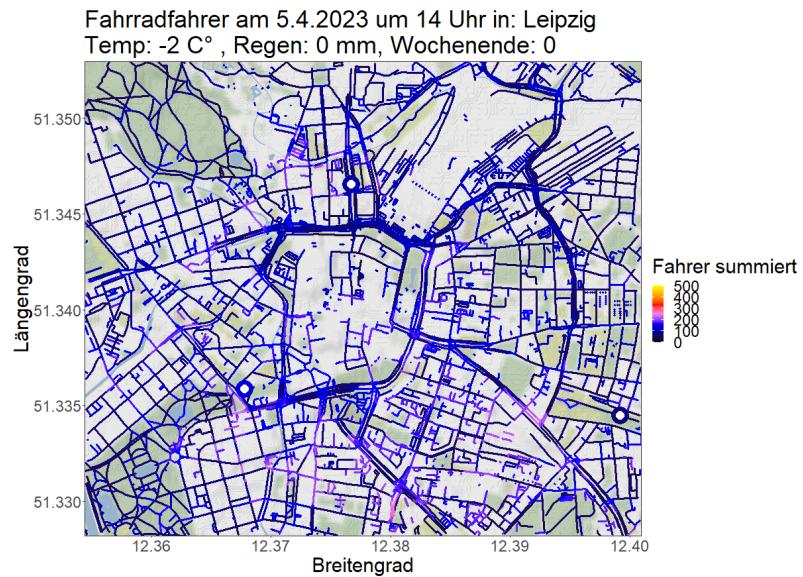


(a) Berlin großer Zoom

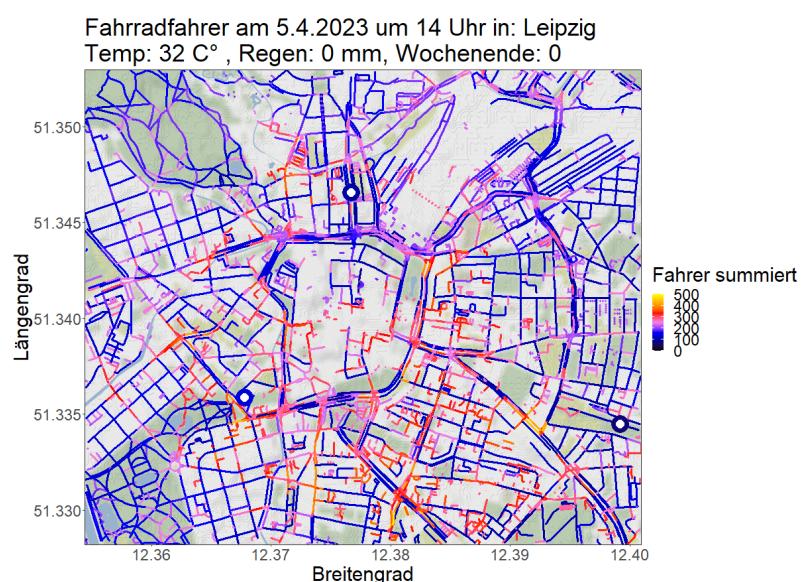


(b) Berlin kleiner Zoom

Fig. 7.11: Kartendarstellung mit unterschiedlichen Details



(a) Leipzig bei kaltem Wetter



(b) Leipzig bei heißem Wetter

Fig. 7.12: Kartendarstellung mit unterschiedlichem Wetter

## LITERATURVERZEICHNIS

- Alattar, M. A., Cottrill, C. D., and Beecroft, M. (2021). Modelling cyclists' route choice using strava and osmnx: A case study of the city of glasgow. *Transportation Research Interdisciplinary Perspectives*.
- Awad, M. and Khanna, R. (2015). *Support Vector Regression*, pages 67–80. Apress, Berkeley, CA.
- BMDV (2020). Verwaltungsvereinbarung sonderprogramm „stadt und land“. *über die Gewährung von Finanzhilfen des Bundes an die Länder nach Artikel 104b des Grundgesetzes und aufgrund des Haushaltsgesetzes 2020 für Investitionen in den Radverkehr durch das Sonderprogramm „Stadt und Land“*.
- Broach, J., Dill, J., and Gliebe, J. P. (2012). Where do cyclists ride? a route choice model developed with revealed preference gps data. *Transportation Research Part A-policy and Practice*, 46:1730–1740.
- Broucke, S. V., Piña, L. M. V., Do, T. H., and Deligiannis, N. (2019). Brubike: A dataset of bicycle traffic and weather conditions for predicting cycling flow. In *2019 IEEE International Smart Cities Conference (ISC2)*, pages 432–437.
- Carl, K. and Dror, M. (2015). Construction of a topographical road graph for bicycle tour routes. *Sports Technology*.
- Colace, F., De Santo, M., Lombardi, M., Pascale, F., Santaniello, D., and Tucker, A. (2020). A multilevel graph approach for predicting bicycle usage in london area. In Yang, X.-S., Sherratt, S., Dey, N., and Joshi, A., edi-

- tors, *Fourth International Congress on Information and Communication Technology*, pages 353–362, Singapore. Springer Singapore.
- Destatis (2022a). Ausländer: Kreise, stichtag, geschlecht: Tabelle 12521-0040. *Statistisches Bundesamt*.
- Destatis (2022b). Bevölkerung: Kreise, stichtag, altersgruppen: Tabelle 12411-0017. *Statistisches Bundesamt*.
- Destatis (2022c). Kraftfahrzeugbestand: Kreise, stichtag, kraftfahrzeugarten: Tabelle 46251-0020. *Statistisches Bundesamt*.
- Eisenberger, D. (2015). Pressemitteilung: Zahlen – daten – fakten zum deutschen fahrradmarkt 2015sehr gutes jahr für die deutsche fahrradindustrie. *Zweirad-Industrie-Verband e.V.*
- Gao, C. and Chen, Y. (2022). Using machine learning methods to predict demand for bike sharing. In Stienmetz, J. L., Ferrer-Rosell, B., and Massimo, D., editors, *Information and Communication Technologies in Tourism 2022*, pages 282–296, Cham. Springer International Publishing.
- Goldmann, K. and Wessel, J. (2021). Some people feel the rain, others just get wet: An analysis of regional differences in the effects of weather on cycling. *Research in Transportation Business and Management*, 40:100541. Active Travel and Mobility Management.
- Hankey, S., Zhang, W., Le, H. T., Hystad, P., and James, P. (2021). Predicting bicycling and walking traffic using street view imagery and destination data. *Transportation Research Part D: Transport and Environment*, 90:102651.
- Harvey, F. and Krizek, K. (2007). Commuter bicyclist behavior and facility disruption. *Minnesota Department of Transportation, Research Services Section*.
- Hausman, J., Hall, B. H., and Griliches, Z. (1984). Econometric models for count data with an application to the patents-r and d relationship. *Econometrica*, 52(4):909–938.

- Heinen, E., van Wee, B., and Maat, K. (2010). Commuting by bicycle: An overview of the literature. *Transport Reviews*, 30(1):59–96.
- Hijmans, R. J. (2021). *geosphere: Spherical Trigonometry*. R package version 1.5-14.
- Holmgren, J., Aspegren, S., and Dahlströma, J. (2017). Prediction of bicycle counter data using regression. *Procedia Computer Science*, 113:502–507. The 8th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2017) / The 7th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH-2017) / Affiliated Workshops.
- Hong J, McArthur DP, S. J. K. C. (2022). Did air pollution continue to affect bike share usage in seoul during the covid-19 pandemic? *J Transp Health*.
- James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013a). *An Introduction to Statistical Learning*, chapter Support Vector Machines, pages 337–372. Springer New York, New York, NY.
- James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013b). *An Introduction to Statistical Learning*, chapter Tree-Based Methods, pages 303–335. Springer New York, New York, NY.
- James, G., Witten, D., Hastie, T., and Tibshirani, R. (2021). *An Introduction to Statistical Learning*, chapter Deep Learning, pages 403–458. Springer New York, New York, NY.
- Kaltenbrunner, A., Meza, R., Grivolla, J., Codina, J., and Banchs, R. (2010). Urban cycles and mobility patterns: Exploring and predicting trends in a bicycle-based public transport system. *Pervasive and Mobile Computing*, 6(4):455–466. Human Behavior in Ubiquitous Environments: Modeling of Human Mobility Patterns.
- Kodzo, J. (2022). Wie verlief der erste lockdown in deutschland? *Wirtschafts Woche*.

- Kodzo, J. and Imöhl, S. (2022). So ist der zweite lockdown in deutschland verlaufen. *Wirtschafts Woche*.
- Kondo, M. C., Morrison, C., Guerra, E., Kaufman, E. J., and Wiebe, D. J. (2018). Where do bike lanes work best? a bayesian spatial model of bicycle lanes and bicycle crashes. *Safety Science*, 103:225–233.
- Kursa, M. B. and Rudnicki, W. R. (2010). Feature selection with the boruta package. *Journal of Statistical Software*, 36(11):1–13.
- Lee, K. and Sener, I. N. (2020). Emerging data for pedestrian and bicycle monitoring: Sources and applications. *Transportation Interdisciplinary Perspectives*, 4:100095.
- Li, X., Xu, Y., Chen, Q., Wang, L., Zhang, X., and Shi, W. (2022). Short-term forecast of bicycle usage in bike sharing systems: A spatial-temporal memory network. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):10923–10934.
- Li, Y., Zheng, Y., Zhang, H., and Chen, L. (2015). Traffic prediction in a bike-sharing system. In *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*, SIGSPATIAL ’15, New York, NY, USA. Association for Computing Machinery.
- Liaw, A. and Wiener, M. (2002). Classification and regression by randomforest. *R News*, 2(3):18–22.
- Liu, X., Kounadi, O., and Zurita-Milla, R. (2022). Incorporating spatial autocorrelation in machine learning models using spatial lag and eigenvector spatial filtering features. *ISPRS International Journal of Geo-Information*, 11(4).
- Meng, M., Zhang, J., Wong, Y. D., and Au, P. H. (2016). Effect of weather conditions and weather forecast on cycling travel behavior in singapore. *International Journal of Sustainable Transportation*, 10(9):773–780.

- Menghini, G., Carrasco, N., Schüssler, N., and Axhausen, K. (2010). Route choice of cyclists in zurich. *Transportation Research Part A: Policy and Practice*, 44(9):754–765.
- Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., and Leisch, F. (2021). *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071)*, TU Wien. R package version 1.7-9.
- Microsoft and Weston, S. (2022). *doParallel: Foreach Parallel Adaptor for the 'parallel' Package*. R package version 1.0.17.
- Microsoft and Weston, S. (2022). *foreach: Provides Foreach Looping Construct*. R package version 1.5.2.
- Mitchell, P. P. (2018). Predicting bike-sharing traffic flow using machine learning.
- Musakwa, W. and Selala, K. M. (2016). Mapping cycling patterns and trends using strava metro data in the city of johannesburg, south africa. *Data in Brief*, 9:898–905.
- Möllers, A., Specht, S., and Wessel, J. (2021). The impact of the covid-19 pandemic and government intervention on active mobility. Technical Report 34. Publication status: Published.
- Nankervis, M. (1999). The effect of weather and climate on bicycle commuting. *Transportation Research Part A: Policy and Practice*, 33(6):417–431.
- Padgham, M., Rudis, B., Lovelace, R., and Salmon, M. (2017). osmdata. *The Journal of Open Source Software*, 2(14).
- Pebesma, E. (2018). Simple Features for R: Standardized Support for Spatial Vector Data. *The R Journal*, 10(1):439–446.
- Pisner, D. A. and Schnyer, D. M. (2020). Chapter 6 - support vector machine. In Mechelli, A. and Vieira, S., editors, *Machine Learning*, pages 101–121. Academic Press.

- Prati, G., Pietrantoni, L., and Fraboni, F. (2017). Using data mining techniques to predict the severity of bicycle crashes. *Accident Analysis and Prevention*, 101:44–54.
- Pritchard, R. (2018). Revealed preference methods for studying bicycle route choice—a systematic review. *International Journal of Environmental Research and Public Health*, 15:470.
- Pucher, J., Dill, J., and Handy, S. (2010). Infrastructure, programs, and policies to increase bicycling: An international review. *Preventive Medicine*, 50:S106–S125.
- Reddy, S., Shilton, K., Denisov, G., Cenizal, C., Estrin, D., and Srivastava, M. (2010). Biketastic: Sensing and mapping for better biking. volume 3, pages 1817–1820.
- Rietveld, P. and Daniel, V. (2004). Determinants of bicycle use: do municipal policies matter? *Transportation Research Part A: Policy and Practice*, 38(7):531–550.
- Romanillos, G., Austwick, M. Z., Ettema, D., and Kruijf, J. D. (2016). Big Data and Cycling. *Transport Reviews*, 36(1):114–133.
- Saha, D., Alluri, P., Gan, A., and Wu, W. (2018). Spatial analysis of macro-level bicycle crashes using the class of conditional autoregressive models. *Accident Analysis and Prevention*, 118:166–177.
- Stock, J. H. and Watson, M. W. (2015a). *Introduction to Econometrics*, chapter Estimation of Dynamic Causal Effects, pages 635–683. PEARSON, 3 edition.
- Stock, J. H. and Watson, M. W. (2015b). *Introduction to Econometrics*, chapter Introduction to Time Series Regression and Forecasting, pages 568–634. PEARSON, 3 edition.
- Thomas, B. and DeRobertis, M. (2013). The safety of urban cycle tracks: A review of the literature. *Accident Analysis and Prevention*, 52:219–227.

- Vandenbulcke, G., Thomas, I., and Int Panis, L. (2014). Predicting cycling accident risk in brussels: A spatial case–control approach. *Accident Analysis and Prevention*, 62:341–357.
- Wessel, J. (2020). Using weather forecasts to forecast whether bikes are used. *Transportation Research Part A: Policy and Practice*, 138:537–559. Publication status: Published.
- Winters, M., Davidson, G., Kao, D., and Teschke, K. (2010). Motivators and deterrents of bicycling: Comparing influences on decisions to ride. *Transportation*, 38:153–168.
- Xu, H., Ying, J., Wu, H., and Lin, F. (2013). Public bicycle traffic flow prediction based on a hybrid model. *Applied Mathematics and Information Sciences*, 7:667–674.
- Zhao, P., Li, S., Li, P., Liu, J., and Long, K. (2018). How does air pollution influence cycling behaviour? evidence from beijing. *Transportation Research Part D: Transport and Environment*, 63:826–838.
- ZIV (2022). Pressemitteilung: „freiheitsmobilität“ bleibt stark nachgefragt. *Der Zweirad-Industrie-Verband e.V.*

## EHRENWÖRTLICHE ERKLÄRUNG

„Ich versichere, dass ich die vorliegende Arbeit ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt und die den benutzten Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.  
Ich bin damit einverstanden, dass meine Arbeit zum Zwecke eines Plagiatsabgleichs in elektronischer Form anonymisiert versendet und gespeichert werden kann.“

Ort, Datum: \_\_\_\_\_ Unterschrift: \_\_\_\_\_