PROFESSUR FÜR WIRTSCHAFTSINFORMATIK

DER FREIEN UNIVERSITÄT BERLIN

Hausarbeit

Business Intelligence

Fachbereich Wirtschaftswissenschaft
der Freien Universität Berlin

Crime Classification in San Francisco

Gutachter(in):	Prof. Dr. Bastian Amberg
Verfasser:	Chris Steden, Maximilian Kerner Emanuele Maurer, Max Schubert
Matrikel-Nr.:	5281167, 5281075, 4683195, 5279608
Abgabetermin:	05. August 2019

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Project Understanding

Von 1934 bis in das Jahr 1963 galt das Hochsicherheitsgefängnis auf der Insel Alcatraz in San Francisco als eines zu der damaligen Zeit berühmt-berüchtigtsten Gefängnisses in den USA. Heute wird mit der Stadt eher eine fortschreitende Tech-Szene assoziiert. Trotzdem lassen sich auch heute in San Francisco täglich Diebställe und Verbrechen beobachten und die Stadt kämpft gegen ihre Kriminalitätsrate. Für dieses Projekt liegt ein Datensatz vor, der eine Aufzeichnung der Verbrechen in San Francisco zwischen den Jahren 2003 bis 2015 beinhaltet. Diese Daten stammen aus einem Wettbewerb der Plattform Kaggle.com, welcher im Zeitraum 2. bis 06.06.2016 durchgeführt wurde. Ziel des Projektes ist eine möglichst genaue Vorhersage der Kategorie eines Verbrechens zu prognostizieren. Durch eine möglichst genaue Vorhersage der Verbrechen können beispielsweise Personaleinsatzpläne der Polizei optimiert oder Präventivmaßnahmen zur Minderung der Verbrechensrate eingeleitet werden.

Der Projektbericht gliedert sich nach der Vorgehensweise zur Modellentwicklung an das Cross Industry Standard Process for Data Mining (CRISP – DM) Modell, welches sich in sechs Phasen unterteilt: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, Deployment. Die Durchführung des letzten Schritts „Deployment“ ist für dieses Projekt nicht notwendig. Das CRISP-DM Modell ist anwendungsneutral und bietet eine Schritt-für-Schritt Anleitung für Data Mining Projekte (Chapman et al. 2000).

Die Genauigkeit des entwickelten Modells kann mit Hilfe der Accuracy gemessen werden. Der Wert stellt dar wie hoch die Wahrscheinlichkeit ist, dass die richtige Straftat vorausgesagt wird. Dies bietet allerdings nur eine bedingte Aussagekraft, da sie keinen Unterschied zwischen falsch positiven und falsch negativen Fehlern ermittelt. Aus diesem Grund wird zusätzlich der Multi Class Logarithmic Loss (MLL) herangezogen. Der MLL ist eine metrische Skala, die eine Einschätzung zur Vorhersagekraft des entwickelten Modells darstellt und gleichzeitig eine Vergleichbarkeit mit anderen Teilnehmern des Wettbewerbs bietet. Der MLL-Bestwert des Wettbewerbs liegt bei 1,95936 und der Medianwert bei 2,59. Zusätzlich werden Confusion Matrizen je Kategorie verwendet, um eine Vergleichbarkeit der Ergebnisse sicherzustellen.

Neben der Qualität des Modells wird die Qualität der Daten anhand folgender Parameter bewertet: Repräsentativität, Informativität, Zuverlässigkeit und Vollständigkeit, externe Faktoren und zusätzliche Anforderungen. Der Datensatz beinhaltet Angaben zu polizeilich

verfolgten Verbrechen im Zeitraum vom 06.01.2003 bis 13.05.2015 in San Francisco. Die Dunkelziffer der tatsächlichen Verbrechen wird wahrscheinlich deutlich höher liegen. Eine hundertprozentige Erfassung aller Verbrechen in der Realität ist unmöglich. Aus diesem Grund ist eine Repräsentativität der Daten gegeben. Zur Überprüfung der Informativität der Daten wurde eine Cognitive Map erstellt (Vgl. Anhang 1). Mit dieser wurden die Einflussfaktoren auf die Kriminalität in San Francisco untersucht und mit dem Datensatz verglichen. Ein erster Plausibilitätscheck hat ergeben, dass die Daten durch die Polizei zuverlässig und vollständig erfasst wurden. Eine detaillierte Überprüfung wird im Kapitel Data Preparation näher beschrieben. Für die externen Faktoren wird die Annahme getroffen, dass diese konstant bleiben. Zusätzlich wird in der Modellentwicklung berücksichtigt, dass es zu keinen ethischen, politischen oder gesetzlichen Konflikten kommt. Hierfür dürfen beispielweise keine diskriminierenden Attribute, wie die Hautfarbe oder das Geschlecht, verwendet werden.

Die Entwicklung der Cognitive Map lässt zusätzlich Rückschlüsse über das Domänenwissen zu. Die Stadt San Francisco gliedert sich in 11 Supervisorial Districts. Um eine möglichst gleichmäßige Auslastung der Polizeistationen zu gewährleisten, unterscheidet sich die polizeiliche Bezirksaufteilung (Vgl. Anhang 2). Die Kriminalitätsraten und Häufigkeiten über die Kategorie der Verbrechen unterscheiden sich in den einzelnen Bezirken. So gelten bspw. die beiden Distrikte “SFPD Mission Station” und “SFPD Tenderloin Station” als die gefährlichsten Distrikte mit den höchsten Aufkommen von Verbrechen (San Francisco Police Department 2019).

2 Data Understanding

In diesem Kapitel werden die zur Verfügung stehenden Daten näher untersucht. Insgesamt sind diese in drei verschiedenen CSV-Dateien dargestellt. Zur Untersuchung der Datenqualität wird diese mit Hilfe verschiedener Methoden und Techniken analysiert.

Der Datensatz beinhaltet neun Attribute: Dates, Category, Descript, DaysOfWeek, PdDistrict, Resolution, Address sowie die X- und Y-Koordinate. Eine detaillierte Auflistung kann dem Anhang 3 entnommen werden. Das Ziel ist eine Vorhersage der Variable „Category“, welche mit 39 verschiedenen Verbrechenskategorien eine hohe Granularität aufweist. Gleichzeitig wird angenommen, dass die Variable nicht-dynamisch und sich damit im Laufe der Zeit nicht revidieren lässt. Die Variablen Dates und DaysOfWeek ermöglichen Rückschlüsse über die Zeit. Mit Hilfe der Variablen PdDistrict, Address und die X- und Y-Koordinate lässt sich der Ort des Verbrechens näher spezifizieren.

Durch die Visualisierung der Daten soll ein besseres Verständnis geschaffen werden (Vgl. Anhang 4-8). Zudem lassen sich räumliche und zeitliche Zusammenhänge erkennen. Zunächst wurde die räumliche Verteilung der Verbrechen untersucht:

Abbildung 1: Verteilung Straftaten über San Francisco (eigene Darstellung)

Aus der Abbildung lässt sich erkennen, dass im Stadtzentrum eine höhere Anzahl von Verbrechen stattfindet als in den Randgebieten. Zudem konnten mit der Visualisierung erste Datenausreißer gefunden werden (Vgl. Anhang 4), die ausgefiltert werden sollten. In der zeitlichen Darstellung über den gesamten Zeitraum lassen sich keine Auffälligkeiten feststellen. Lediglich in der Darstellung der Tageszeit lässt sich erkennen, dass die meisten Verbrechen tagsüber stattfinden.

Die Quantität und Qualität der verfügbaren Daten entscheiden über die Modell- und Ergebnisqualität. Die Dateien weisen jeweils über 800.000 Datensätze auf. Damit ist eine ausreichende Quantität gegeben, da jedes gängige Modell mit dieser Anzahl an Datensätzen arbeiten kann. Aus der Visualisierung lassen sich bereits erste Rückschlüsse zur Datenqualität treffen. Der Datensatz kann hinsichtlich der Attribute optimiert werden, da für die Modelle nicht jedes gegebene Attribut benötigt wird. Dies wird im folgenden Kapitel "Data Preparation" näher erläutert. Wie eingangs erwähnt, wurden die Daten und die Zuordnung der Verbrechens-Kategorien von der Polizei durchgeführt. Um eine höhere Gewissheit zur richtigen Erfassung und Zuordnung zu erlangen, wäre ein tieferes Domänenwissen erforderlich. In diesem Fall wird allerdings davon ausgegangen, dass die Verbrechen richtig erfasst und zugeordnet wurden. Zudem wurden in einer ersten Untersuchung der Datensätze Duplikate und ungültige Angaben erkannt, die rausgefiltert werden sollten. Dieser Prozess wird im folgenden Kapitel näher erläutert.

3 Data Preparation

Auf Basis des CRISP-DM Prozesses soll im Folgenden die Data Preparation näher beschrieben werden. Zu Beginn wird die Auswahl der Attribute, die für das trainieren der Modelle verwendet werden, näher erläutert. Darauffolgend werden die Daten um Ausreißer und fehlende Daten bereinigt, um sie zum Abschluss des Kapitels so aufzubereiten, dass sie für die Modelling Phase genutzt werden können.

3.1 Data Selection

Nach ausführlicher Sichtung der Daten wurde die Spalte des Datums so aufgeteilt, das jedes Attribut mit einer eigenen Spalte versehen ist. Durch die Aufspaltung des Datums kann beispielsweise der Zusammenhang zwischen dem Tag und dem Verbrechen hergestellt werden. Es kann davon ausgegangen werden, dass der Tag einen signifikanten Einfluss auf das verübte Verbrechen haben wird. Außerdem kann aufgrund der Extraktion des Monats beispielsweise überprüft werden, ob im Winter andere Delikte verübt werden als im Sommer. Des Weiteren kann die Uhrzeit mit in die Beobachtung einbezogen werden, da mit einer Änderung der Tageszeit auch eine Änderung der Kriminalität einhergeht.

3.2 Feature Selection

Im Folgenden wird der Pearson Chi-Quadrat-Test angewendet, um zu überprüfen, ob ein signifikanter Zusammenhang oder eine Unabhängigkeit zwischen einem Attribut und der Zielvariable (Kategorie) besteht. Das Ziel des Tests ist zu bewerten, wie hoch der Einfluss eines Attributes auf die Vorhersage der Verbrechenskategorie ist. Dabei überprüft der Chi-Quadrat-Test, ob die Nullhypothese zwischen einer erwarteten Verteilung und der tatsächlichen Beobachtung übereinstimmt. Kann die Nullhypothese nicht bestätigt werden, wird von einer statistischen signifikanten Abweichung ausgegangen. Das Ergebnis des Chi-Quadrat-Tests steigt mit Abweichung der Verteilung (Han et al. 2011). Durch die Größe des Ergebnisses kann daher erkannt werden, welcher Wert den Größten Einfluss auf die Vorhersage haben wird

Attribut:	Pearson Chi-Square
x	1301158
y	1301158
Address	881220
Description	33364

Minutes	2243
Day	1140
Hour	874
Resolution	608
Year	456
Month	418
Seconds	0

Tabelle 1: Chi-Quadrat-Test

Wie aus Tabelle 1 zu erkennen, nehmen die X- sowie die Y-Koordinate den größten Einfluss auf die Zielvariable. Nach den beiden Koordinaten besitzt die Adresse den nächstgrößten Einfluss, diese kann nachrangig behandelt werden, da sie sich ähnlich zu den Koordinaten verhält. Des Weiteren wird die Description nicht weiterverwendet, da diese zum Tatzeitpunkt nicht bekannt ist.

Neben der Durchführung des Chi-Quadrat Tests wurden die Daten auf Aktualität, Repräsentativität sowie auf außerordentliche Ereignisse überprüft. Durch das plotten der Daten konnte kein signifikanter Unterschied in der Anzahl von Verbrechen in den einzelnen Jahren entdeckt werden. Darüber hinaus ist kein Rückgang oder eine Veränderung der Kriminalität festzustellen. Außerordentlichen Ereignisse, die durch Domainwissen aufgedeckt wurden, konnten aufgrund der Größe des Datensatzes ignoriert werden, da kein signifikanter Einfluss nachgewiesen werden konnte. Aus diesem Grund kann der Datensatz als repräsentativ für die Implementierung der Modelle angesehen werden.

3.3 Data Cleaning

Im vorherigen Abschnitt "Data Understanding" wurden Unstimmigkeiten in den Datensätzen festgestellt. Diese wurden wie folgt identifiziert und bereinigt. Der Datensatz enthält 67 ungültige Koordinaten, welche durch die Visualisierung identifiziert wurden. Diese wurden nochmals mittels Z-Score ermittelt (Han et al. 2011). Nach Austausch mit Domain-Experten kann davon ausgegangen werden, dass es sich bei diesen Koordinaten um Default-Einstellungen gehandelt hat, welche immer in Kraft getreten sind, sobald das GPS-Signal nicht funktionsfähig war. Durch die geringe Anzahl ungültiger Koordinaten wurde sich gegen eine Aufbereitung entschieden und die Daten wurden aus dem Datensatz entfernt.

Der Datensatz wurde außerdem noch auf Missing Values untersucht. Die Untersuchung zeigt, dass im Datensatz keine Missing Values bestehen. Darüber hinaus wurden alle Leerzeichen

entfernt und um weitere Fehler in der Modelling Phase zu vermeiden, wurden alle Buchstaben in Großbuchstaben umgewandelt. Abschließend konnten 2.323 Duplikate über die numerischen Werte identifiziert und entfernt werden. Weitere Maßnahmen zur Steigerung der Datenqualität, wie die Verwendung eines Spell-Checker, waren nicht erforderlich.

3.4 Data Transformation

Nach der Bereinigung des Datensatzes wird dieser im Folgenden final bearbeitet, um die bestmöglichen Ergebnisse in der Modelling Phase zu erzielen. Zu Beginn werden alle Spalten von „Strings“ in numerische Werte umgewandelt, um Problemen in der Modelling Phase entgegenzuwirken. Nachfolgend wurde für das Attribut DayOfWeek sowie für PdDistrict eine Dummy-Variable eingeführt. Die durch die Dummy-Variable ersetzen Spalten DayofWeek sowie PdDistrict wurden im Nachhinein gelöscht. Damit das Modell die einzelnen Kategorien erkennt, wurden die einzelnen Kategorien mittels eines Label Encoders in numerische Werte umgewandelt.

Um die Performance des Modells zusätzlich zu verbessern, wurden die vorher identifizierten und nicht mehr benötigten Attribute Resolution, Address sowie Seconds und Minutes aufgrund ihrer geringen Einflussnahme auf die Vorhersage entfernt. Der finalisierte Datensatz besteht nach der durchgeführten Transformation, aus X sowie Y-Koordinate, aus den Kategorien Year, Month, Hour sowie Day und den Dummy-Variablen PdDistrict sowie DayofWeek.

4 Modeling

Zu Beginn wird die Auswahl der Modelle näher erläutert. Die initial gewählten Modelle sollten verschiedene Vorgehensweisen vorstellen, um die Problemstellung zu lösen. Im Verlauf des CRISP-DM Prozesses wurde sich auf die vielversprechendsten Modelle konzentriert, welche zum Abschluss des Kapitels gesondert dargestellt werden.

4.1 Model Selection

Die vorliegende Problemstellung besteht in der Vorhersage einer Verbrechenskategorie. Die einzelnen Kategorien lassen sich in keine natürliche Reihenfolge bringen und sind als unabhängig voneinander anzusehen. Der Datensatz beinhaltet bereits markierte Daten, die für ein Training des Modells genutzt werden können. Aus diesem Grund wird ein Lösungsansatz aus dem Bereich Supervised Learning gesucht.

Modelle im Bereich Supervised Learning lassen sich in zwei Kategorien unterteilen. Die erste Kategorie wird durch Klassifikationsanalysen beschrieben. Das Ziel dieser Analysen ist es, den Input einzelnen und vorher festgelegten Kategorien zuzuordnen (Liu et al. 2012). Bei dieser Zuordnung kann es zu Problemen kommen, wenn ein Ungleichgewicht in den Daten vorliegt. Der Naives Bayes Klassifikator ordnet beispielsweise die Daten den Klassen zu, zu denen sie mit größter Wahrscheinlichkeit gehören. Der Kategorie „Theft“ liegt ein Anteil von über 30% im Datensatz zugrunde. Aufgrund dieses großen Anteils kann es bei der Verwendung dieser Modelle zum Majority-Class-Problem kommen und somit zu einer Ungenauigkeit in der Vorhersage führen. Um herauszufinden wie diese Methoden auf den Datensatz reagieren, wurde der Naive Bayes Klassifikator sowie die Variante des k-Nearest Neighbour initial implementiert (Chapelle et al. 2009).

Die zweite Kategorie des Supervised Learning wird durch Regressionsanalysen beschrieben. Beide Kategorien wirken sehr ähnlich zueinander, unterscheiden sich allerdings in ihren Grundsätzen. Regressionsanalysen beschreiben wie hoch die Wahrscheinlichkeit ist, dass der Input in eine der vorher festgelegten Kategorien fällt oder nutzen lineare Funktionen, um die Daten einzelnen Klassen zuzuordnen (ebd.). Aus diesem Grund wurde aus diesem Bereich die logistische Regression, eine Support Vector Machine sowie ein Random Forest implementiert.

Eine weitere Alternative stellt die Implementierung eines neuronalen Netzes dar. Die Stärke neuronaler Netze liegt in der Fähigkeit auf Basis des Outputs zu lernen (Kalchbrenner et al. 2014). Für das vorliegende Projekt eignet sich dieses Modell aufgrund der flexiblen Anpassungsmöglichkeiten während des Trainings.

In einer Literaturrecherche wurden erste Annahmen über die Genauigkeiten der einzelnen Modelle analysiert. Um diese innerhalb des Projektrahmens zu überprüfen, wurden aus den verschiedenen Lösungswegen jeweils eine Methode implementiert. Diese Aufgabe wurde auf die einzelnen Projektmitglieder gleichmäßig aufgeteilt. Folgende Modelle wurden während des Projektes initial implementiert:

- Naive Bayes Klassifikation
- k-Nearest Neighbour
- Logistische Regression
- Random Forest
- Support Vector Machines
- Neuronales Netz

4.2 Model Implementation

Die einzelnen Modelle wurden mithilfe der Python Bibliotheken Scikit-Learn sowie Keras implementiert. Als Backend wurde TensorFlow verwendet. Im Prozess der Implementierung wurden die einzelnen Modelle iterativ verbessert. Während dieser Entwicklung wurde ein Validierungsdatensatz von den Daten extrahiert. Dieser wies abschließend ein Verhältnis von 80% Trainingsdatensatz zu 20% Validierungsdatensatz auf.

Nach der Literaturrecherche und den ersten Implementierungsansätzen stellte sich heraus, dass der Random Forest sowie das Neuronale Netz die vielversprechendsten Ergebnisse liefern werden. Aus diesem Grund wird die Implementierung der beiden Modelle im Folgenden näher erläutert.

4.2.1 Random Forest

Ein Random-Forest-Klassifikator basiert auf einer Vielzahl von Decision Trees. Bei der Anwendung der Methode wird der Datensatz zu Beginn in verschiedene Samples aufgeteilt. Für jeden dieser Teile wird ein Decision-Tree-Klassifikator erzeugt. Die Fehlerrate eines Forest nimmt mit zunehmender Anzahl an Decision Trees ab. Soll die Klasse eines neuen Datensatzes ermittelt werden, stimmen alle Decision Trees des Random Forest über die Klasse ab (Breiman

2001). Dies bedeutet, dass die Leistung eines Forests von der Güte jedes einzelnen Trees abhängig ist. Das beste Feature des Sub-Datensatzes wird verwendet, um die Knoten des Trees zu teilen (Breiman 2001).

Durch die Verwendung des Random-Forest-Klassifikators können alle Vorteile eines Decision Trees behalten werden. Zusätzlich bietet die Verwendung dieser Methode Vorteile im Bereich des Overfittings durch das sogenannte Bagging. Bagging beschreibt den Vorgang, dass einzelne Decision Trees nur einen Teil des gesamten Datensatzes sowie nur einen geringen Teil aller Features zum Trainieren verwenden (Breiman 2001).

Der für das Projekt verwendete Random Forest wurde mit 50 Decision Trees implementiert. Die Anzahl der Decision Trees wurde während des Prozesses immer wieder angepasst, um die besten Vorhersageergebnisse zu erhalten. Jeder Decision Tree erhält als Input 6 Features. Die Tiefe der einzelnen Decision Trees wird zusätzlich auf 14 beschränkt. Als Split-Kriterium wurde die Gini-Impurity verwendet.

4.2.2 Neuronales Netz

Im Rahmen dieser Arbeit wurden zwei Neuronale Netze implementiert. Die Erarbeitung der einzelnen Komponenten erfolgte dabei iterativ. Es wurden verschiedene Aktivierungsfunktionen, verschiedene Arten von künstlichen Neuronen und verschiedene Optimizer während des Prozesses verwendet. Im folgenden Abschnitt werden die verwendeten Komponenten der Netzwerke vorgestellt.

Zu Beginn der Implementation wurde ein einfaches Neuronales Netz mit Dense-Layern implementiert. Bei dieser Art von Neuronalem Netz sind die einzelnen Layer vollständig miteinander verbunden. Durch den simplen Aufbau und den geringen Ressourcenbedarf des Netzes bietet sich dieses für eine schnelle Prototyping Phase an. Das Netzwerk wurde mit 30 Schichten sowie 39 Knoten aufgebaut. Die an das Modell gestellte Erwartung einer möglichst genauen Vorhersage konnte jedoch nicht erreicht werden. Aus dieser Implementierung konnten für das zweite Netzwerk wichtig Erkenntnisse gewonnen werden.

Im Laufe der Bearbeitung stellte sich der Optimizer Adadelta als besonders geeignet dar. Das Besondere an diesem Optimizer liegt an dem Umgang mit dem Gradientenabstiegsverfahren. Wenn dieses in der Nähe eines lokalen Minimums liegt, könnten hohe Lernraten zu Sprüngen im Log-Loss Ergebnise führen. Adadelta löst dieses Problem durch eine Absenkung der Lernrate in der Nähe eines Minimums. Die Dimensionen können trotzdem einen starken

Einfluss auf das Training haben. Adadelta legt im Gegensatz zu anderen Optimizern für jede Dimension eine spezifische flexible Lernrate fest, wodurch eine Grundrobustheit gegenüber Änderungen der Hyperparameter gegeben ist. Bekannte Optimizer, wie beispielsweise der SGD, reagieren sensitiv gegenüber Änderungen der Hyperparameter. Auch bei großen Gradienten, verschiedenen Architekturen und bei einem Rauschen der Daten erweist sich Adadelta als robust (Zeiler 2012).

Als Aktivierungsfunktion für die künstlichen Neuronen wurde die Parametric Rectified Linear Unit (PReLU) verwendet. Diese bietet den Vorteil, dass sie im Gegensatz zu Softmax oder Sigmoid keine obere Schranke beinhaltet. Außerdem wird durch den Einsatz von PReLU eine sehr geringe Belastung der Recheneinheit gewährleistet und das Risiko des Overfittings verringert. Die Funktion erreichte in vielen Projekten eine sehr gute empirisch überprüfbare Leistung (He et al. 2015). Um zusätzlich ein Overfitting zu vermeiden, wurde der Parameter „Use_Bias“ verwendet. Der Bias stellt ein zusätzliches Neuron dar, welches in jedem Layer enthalten ist und den Integer-Wert eins speichert. Durch den Wert des Bias b und dem Gewicht der Verbindung w wird die Aktivierungsfunktion um den konstanten Wert $b * w$ verschoben. Durch diese Verschiebung wird eine schnellere Erreichung der Aktivierungsschwelle gewährleistet, was zu einer verbesserten Generalisierung beiträgt. Das Neuronale Netz verfügt zusätzlich über eine Dropout-Funktion, welche die Verbindungen zwischen den einzelnen Layern mit den geringsten Gewichten entfernt (Srivastava et al. 2014).

Des Weiteren wurde ein Validierungsdatensatz verwendet, um während des Trainings des Neuronalen Netzes zu prüfen, ob ein Overfitting stattfindet. Dies wäre der Fall, wenn das Ergebnis der Kostenfunktion des Trainingssatzes sinkt und gleichzeitig das Ergebnis des Validierungsdatensatzes konstant bleibt.

Aufgrund dieser Erkenntnisse wurde ein Neuronales Netz mit Long-Short-Term-Memory (LSTM) Schichten aufgebaut. Diese sind auf das Verarbeiten von sequenziellen Daten spezialisiert (Goodfellow et al. 2016). Das letztendlich implementierte Neuronale Netz wurde mit 51 LSTM-Schichten mit jeweils 70 Knoten implementiert.

Das Training der Neuronalen Netze wurde auf Google Colab durchgeführt. Google stellt für die Verarbeitung der Python-Scripts eine Tesla K80 GPU von Nvidia bereit. Die GPU hat ungefähr 12 Gigabyte GDDR5 VRAM und 2496 CUDA Kerne. Ein schnelles Training der Modelle konnte somit sichergestellt werden (Google Colab 2019).

5 Evaluation

Im folgenden Kapitel erfolgt die Evaluation des Projektes. Im ersten Abschnitt, der Modellevaluation, wird das Modell mit Hilfe der Accuracy und Logloss-Funktion (MLL) evaluiert. Im zweiten Abschnitt wird ein Fazit gezogen und ein Ausblick gegeben.

5.1 Modellevaluation

Zur Evaluation der Modelle wird zunächst die Berechnung der Accuracy herangezogen. Diese wird häufig zur Evaluation von Data-Mining Projekte genutzt, da sie eine schnelle und einfache Vergleichbarkeit der entwickelten Modelle ermöglicht (Hossing & Sulaimann 2015). Die Grundlage zur Berechnung der Accuracy und anderen Evaluationsmetriken bilden die Output-Ergebnisse der Modelle, welche in einer Confusion Matrix festgehalten werden (Provost et al. 2013). Eine weitere Evaluationsmetrik, die in diesem Projekt betrachtet wird, ist der Kaggle-Score. Dieser wird mittels einer Vorlage auf der Kaggle-Plattform hochgeladen und durch die Logloss-Funktion (MLL) bewertet.

Wie in Anlage 9 ersichtlich, wurde zunächst eine Confusion Matrix für den Random Forest erstellt. Dies bildet die Grundlage für eine weitere Evaluation durch die Accuracy-Kennzahl. Das Mapping der Klassifizierungen kann der Anlage 11 entnommen werden. Die Werte der Y-Achse stellen die tatsächlichen Werte dar, die aus dem Trainingsdatensatz entnommen werden. Die X-Achse stellt die Vorhersage auf Grundlage der tatsächlichen Werte dar.

Die Accuracy des Neuronalen Netzes wurde mit Hilfe der Funktion aus der Keras Bibliothek ermittelt. Der Verlauf im Trainingszustand und die Verbesserung der Accuracy kann aus der Anlage 10 entnommen werden. Die berechnete Accuracy für die Modelle sind aus der Tabelle 2 ersichtlich und werden zwischen Trainingsset und Testset untergliedert:

	Trainingsset	Testset
Random Forest	32,14%	27,81%
Neuronales Netzwerk	19,83%	38,76%

Tabelle 2: Accuracy Random Forest & Neuronales Netz

Der Kaggle-Score wurde mit Hilfe einer Vorlage auf der Kaggle-Plattform hochgeladen. Die Ergebnisse sind in Tabelle 3 dargestellt. Der Score wird mit folgender Formel berechnet:

$$logloss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} * \log(P_{iJ})$$

$$yi\ j \ldots 0\ or\ 1\ Label\ \forall i\ \in\ N, j\ \in\ M$$

$$pi\ j \ldots Predicted\ probability\ \forall i\ \in\ N\ , j\ \in\ M$$

$$N \ldots Number\ of\ cases\ in\ the\ test\ set$$

	Logloss – Score
Random Forest	6,09684
Neuronales Netzwerk	2,61572

Tabelle 3: Logloss-Score Random Forest & Neuronales Netz

Werden anhand dieser Werte die einzelnen implementieren Modelle aus Kapitel 4.1 verglichen, so stellen das Random Forest und das Neuronale Netzwerk die besten Accuracy Werte und Logloss-Score bereit und bilden somit die erfolgversprechendste Lösung. Im Vergleich zu anderen implementierten Modellen im Kaggle Scoreboard mit dem Logloss-Score landet der implementierte Random-Forest auf Platz 1700 und bildet somit eines der schwächeren Modelle, auch in Bezug auf den MLL-Durchschnittswert. Das Neuronale Netzwerk hingegen erreicht mit dem Logloss einen Platz im Bereich 1000 und liegt im MLL-Durchschnitt und kann somit als das beste Model für die Vorhersage einer Verbrechenskategorie bewertet werden.

5.2 Fazit und Ausblick

Für dieses Projekt wurde das CRISP-DM Modell verwendet, welches sich als geeignet herausstellte. Insbesondere mit den vorgesehenen Iterationsschritten konnte die Ergebnisqualität deutlich gesteigert werden. In der Projektplanung sollte ausreichend Zeit für Iterationen eingeplant werden, da es sonst zu zeitlichen Engpässen kommen kann. Wichtig ist ebenfalls, ausreichend Zeit für die Phase des Project Understandings einzuplanen, um ein gemeinsames Verständnis über das Projektziel zu schaffen.

Eine Verbesserung der vorgestellten Modelle kann mithilfe verschiedener Ansätze erreicht werden. Durch die große Anzahl an Diebstählen im Datensatz liegt ein Ungleichgewicht vor. Dieses Ungleichgewicht kann mitunter zu verzerrten Ergebnissen beitragen. Es existieren eine Vielzahl von Möglichkeiten, die eingesetzt werden können, um diese Problematik entgegenzuwirken. Eine Möglichkeit besteht beispielsweise darin, die Kategorie aufgrund ihrer

Beschreibung in verschiedene Kategorien aufzuteilen, um die Dominanz einer Majority Class abzubauen.

Neben der Manipulation von bereits verwendeten Daten, kann eine Anreicherung der Datenbasis zu besseren Vorhersagen beitragen. Durch das Hinzufügen weiterer Attribute, wie Daten über die Wettervorhersage, können langfristig weitere Verbesserungen erwartet werden. Durch die Einbeziehung eines Domänenexperten könnten die Aussagekraft der Ergebnisse erhöht werden. Dieser könnte beispielsweise feststellen, ob die Erfassung und Zuordnung der Verbrechen repräsentativ sind.

Literaturverzeichnis

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Chapelle, O., Scholkopf, B., & Zien, A. (2009). Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. IEEE Transactions on Neural Networks, 20(3), 542-542.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). CRISP-DM 1.0: Step-by-step data mining guide. SPSS inc, 16.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning| The MIT Press. Cambridge, Massachusetts.

Google Colab (2019). Backend specs. Abgerufen am 01.08.2019 von: https://colab.research.google.com/drive/1_x67fw9y5aBW72a8aGePFLlkPvKLpnBl

Han, J., Pei, J., & Kamber, M. (2011). Data mining: concepts and techniques. Elsevier.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE international conference on computer vision (pp. 1026-1034).

Hossin, M., & Sulaiman, M. N. (2015). A review on evaluation metrics for data classification evaluations. International Journal of Data Mining & Knowledge Management Process, 5(2), 1.

Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A convolutional neural network for modelling sentences. arXiv preprint arXiv:1404.2188.

Liu Q., Wu Y. (2012) Supervised Learning. In Seel N.M. (eds) Encyclopedia of the Sciences of Learning. Springer, Boston, MA, 331 – 350.

Provost, F., & Fawcett, T. (2013). Data science for business. Beijing: O'Reilly.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). Imagenet large scale visual recognition challenge. International journal of computer vision, 115(3), 211-252.

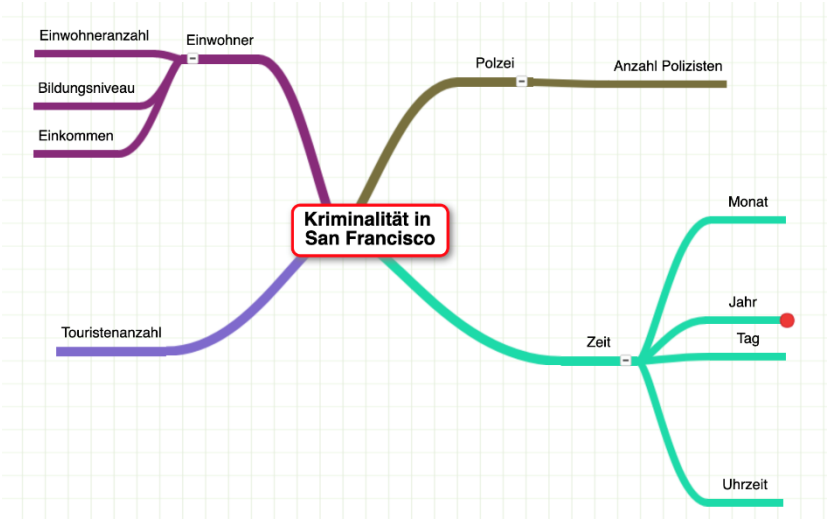
San Francisco Police Department (2019). Police Station Finder. Abgerufen am 01.08.2019 von: https://www.sanfranciscopolice.org/station-finder

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958.

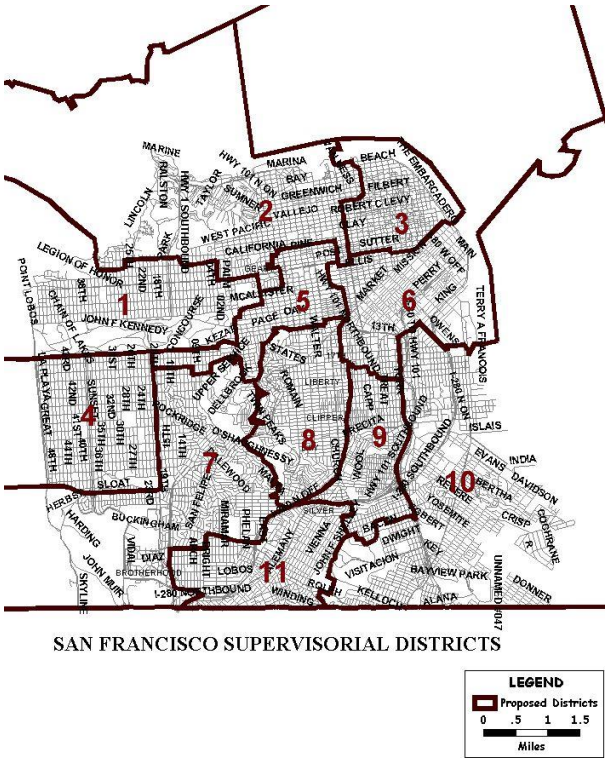
Zeiler, M. D. (2012). ADADELTA: an adaptive learning rate method. arXiv preprint arXiv:1212.5701

Anhang

Anhang 1: Cognitive Map (eigene Darstellung)



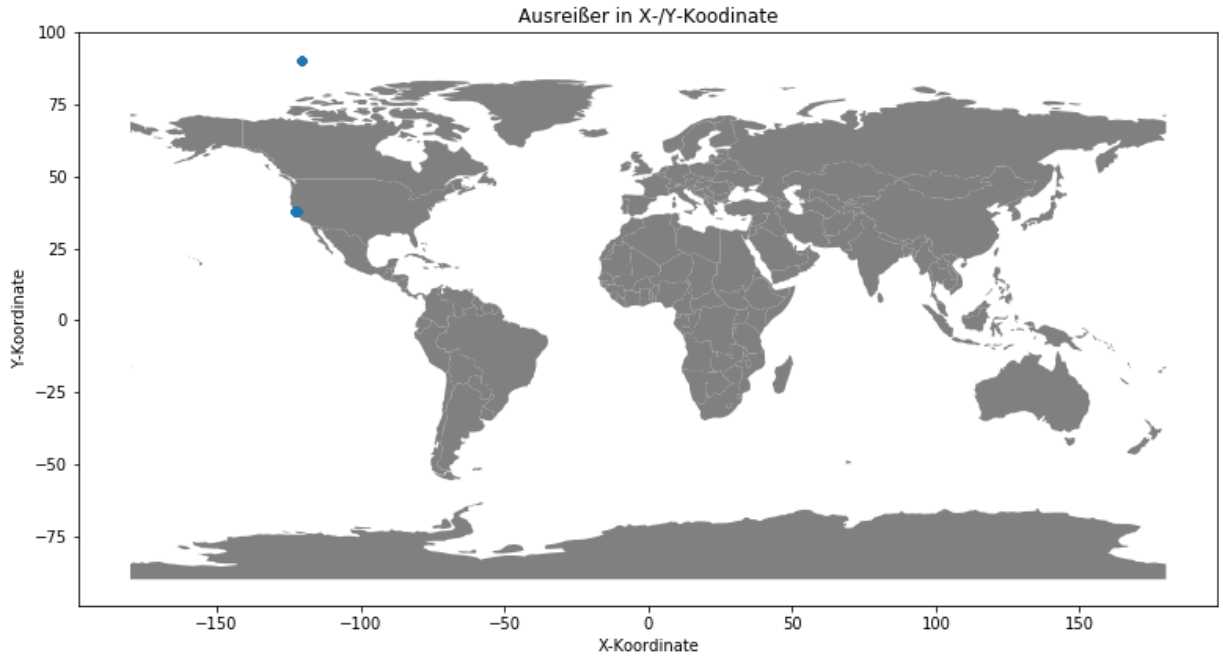
Anhang 2: Gegenüberstellung Supervisorial Districts und Police Districts (San Francisco Police Department 2019)



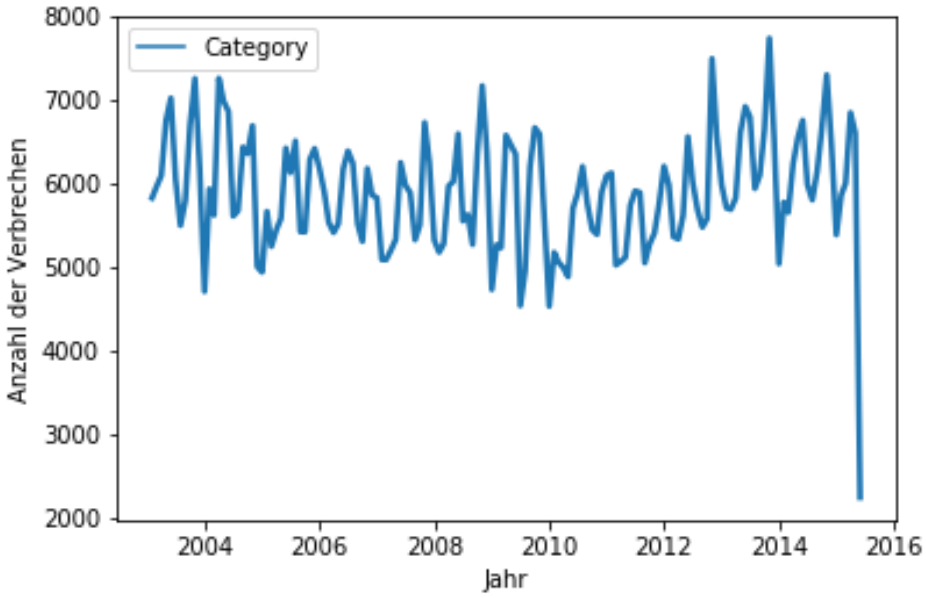
Anhang 3: Variablen im Datensatz

Data	Beschreibung	Type
Dates	Zeitstempel des Verbrechens (YYYY-MM-DD HH:MM:00)	String / Numerisch, Intervall Scala
Category	Kategorie des Verbrechens	Nominal
Descript	Beschreibung des Verbrechens	String
DayOfWeek	Wochentag (Monday bis Sunday)	Oridnal
PdDistrict	Name des zuständigen Polizeipräsidiums	String
Resolution	Lösung des Verbrechens	String
Address	Adresse des Verbrechens	String
X	Längengrad	Float
Y	Breitengrad	Float

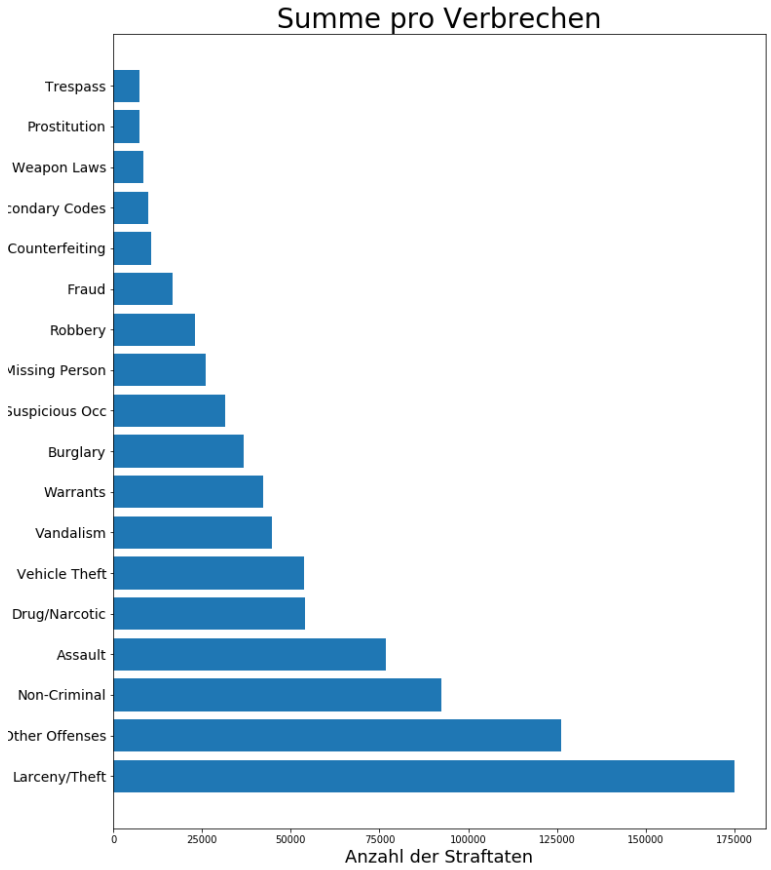
Anhang 4: Datenvisualisierung 1) Ausreißer Koordinatensystem



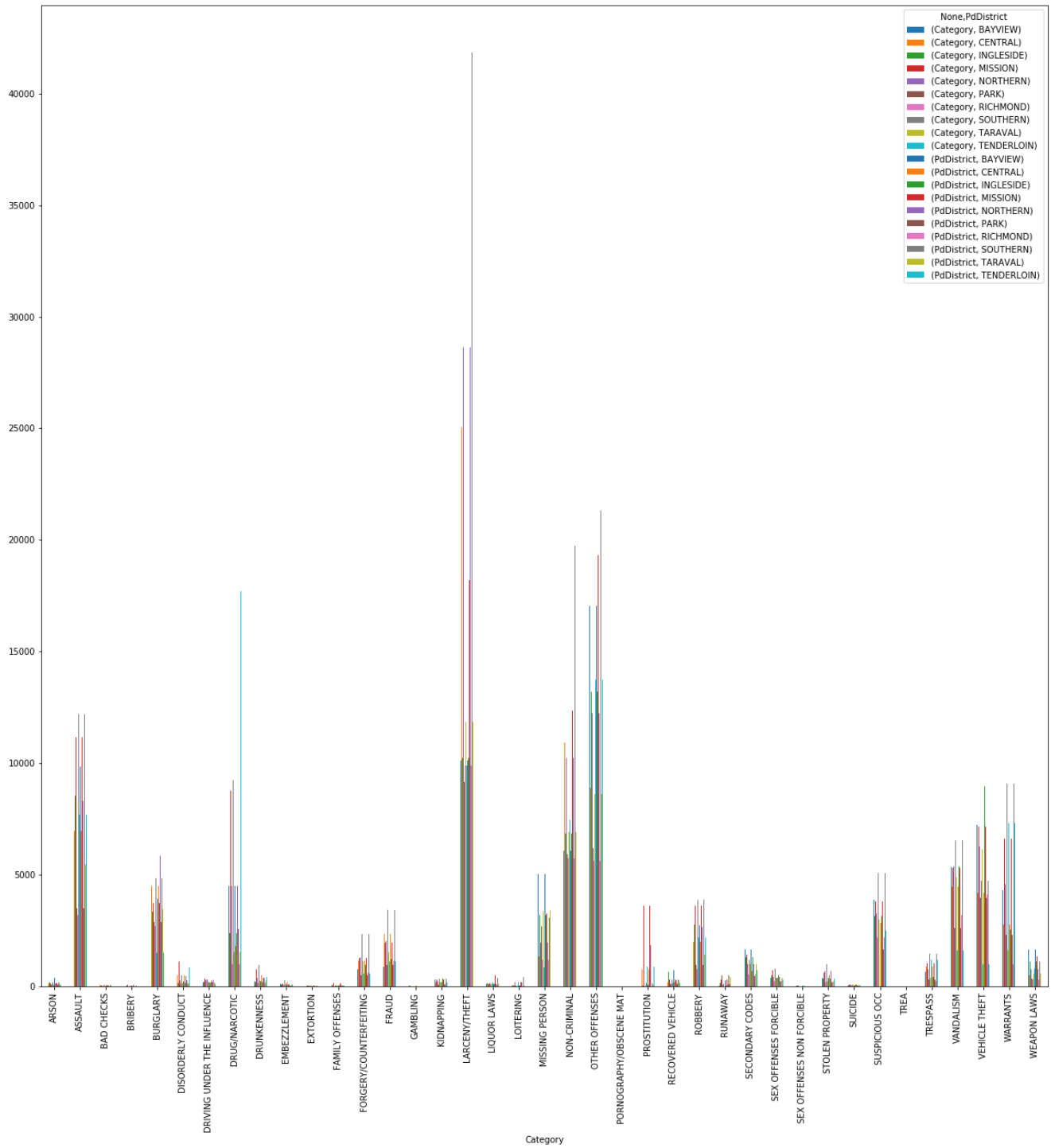
Anhang 5: Datenvisualisierung 2) Verteilung Verbrechen über Jahre



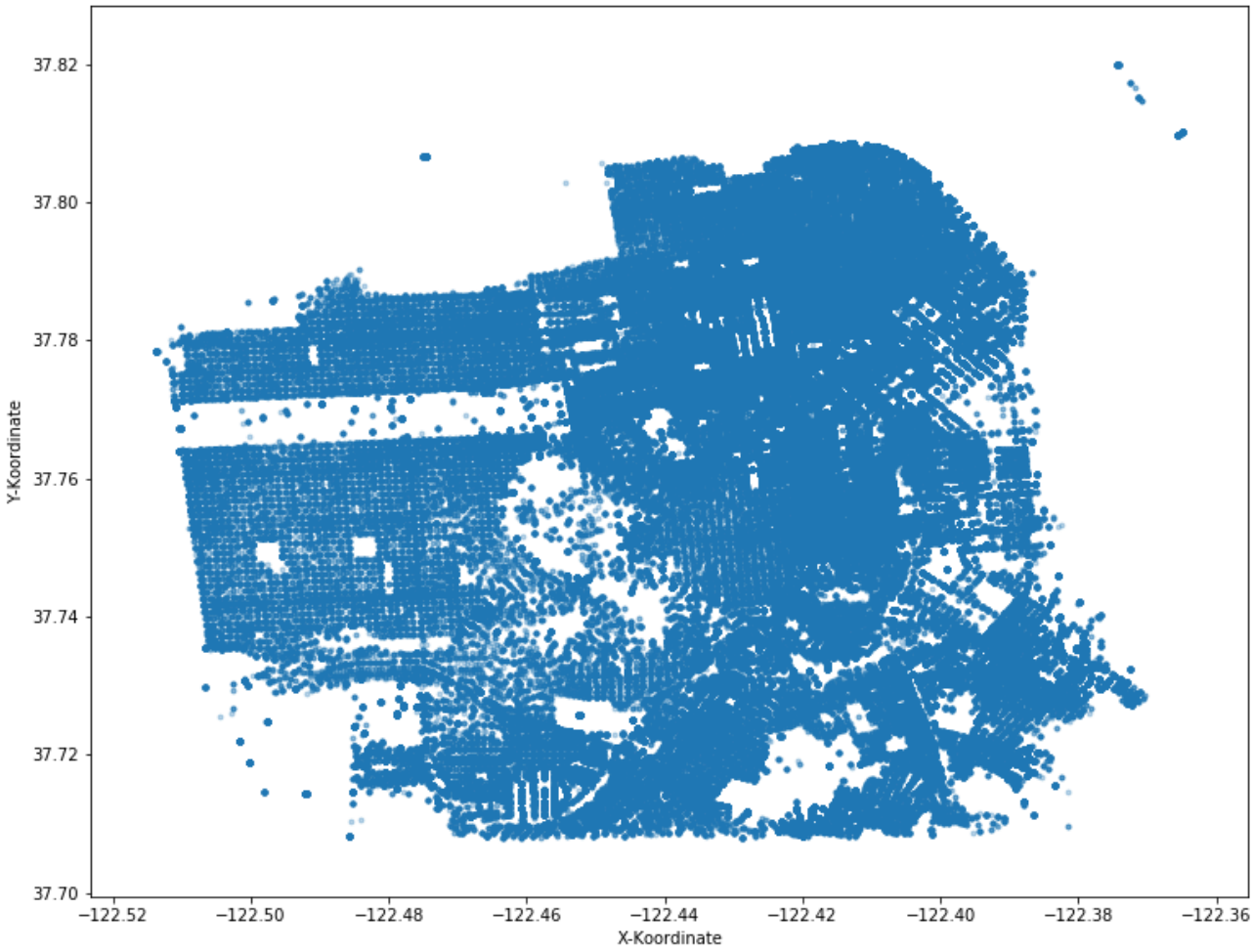
Anhang 6: Datenvisualisierung 3) Verbrechen nach Kategorie (Kurzübersicht)



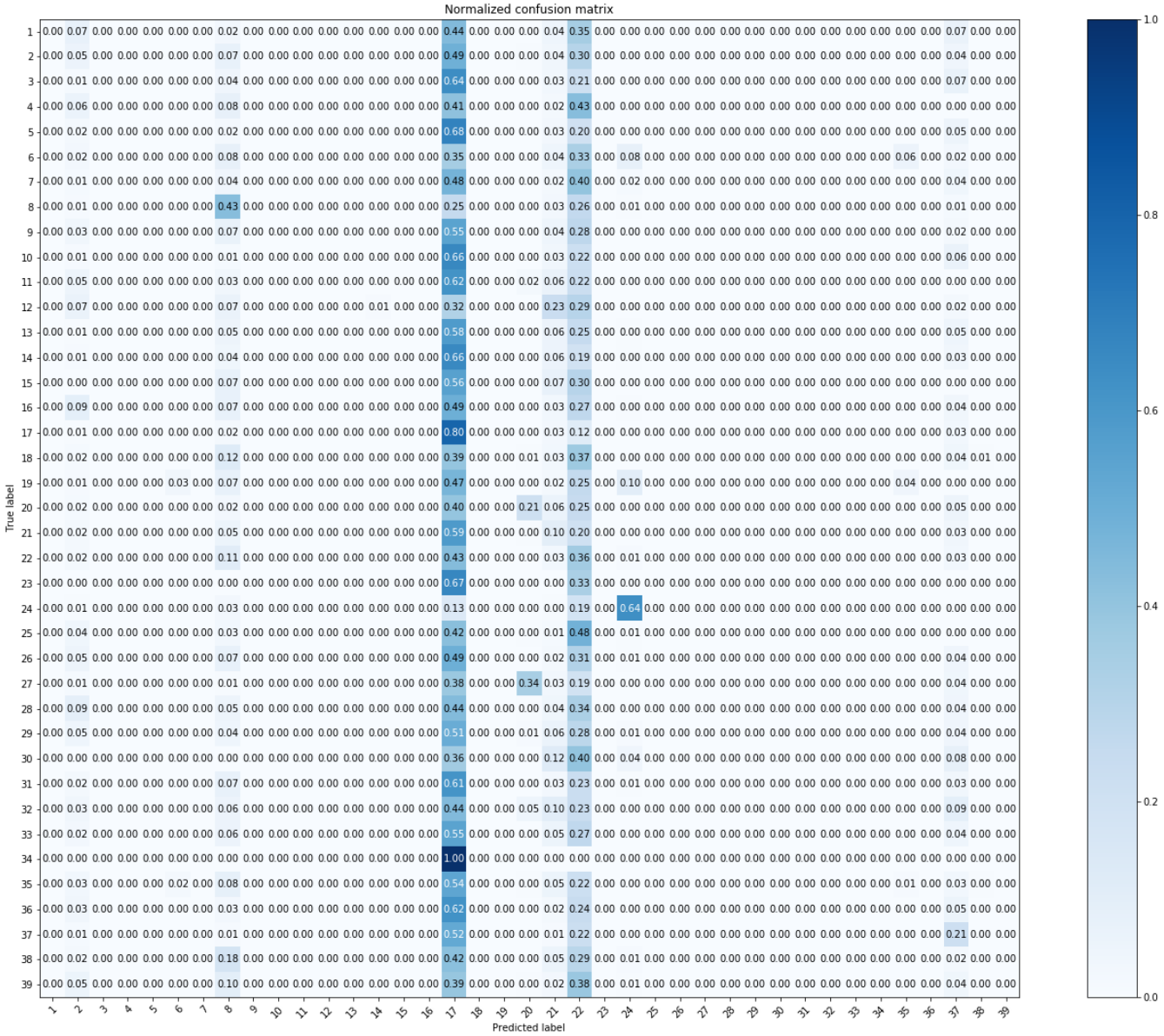
Anhang 7: Datenvisualisierung 3) Verbrechen nach Kategorie und District

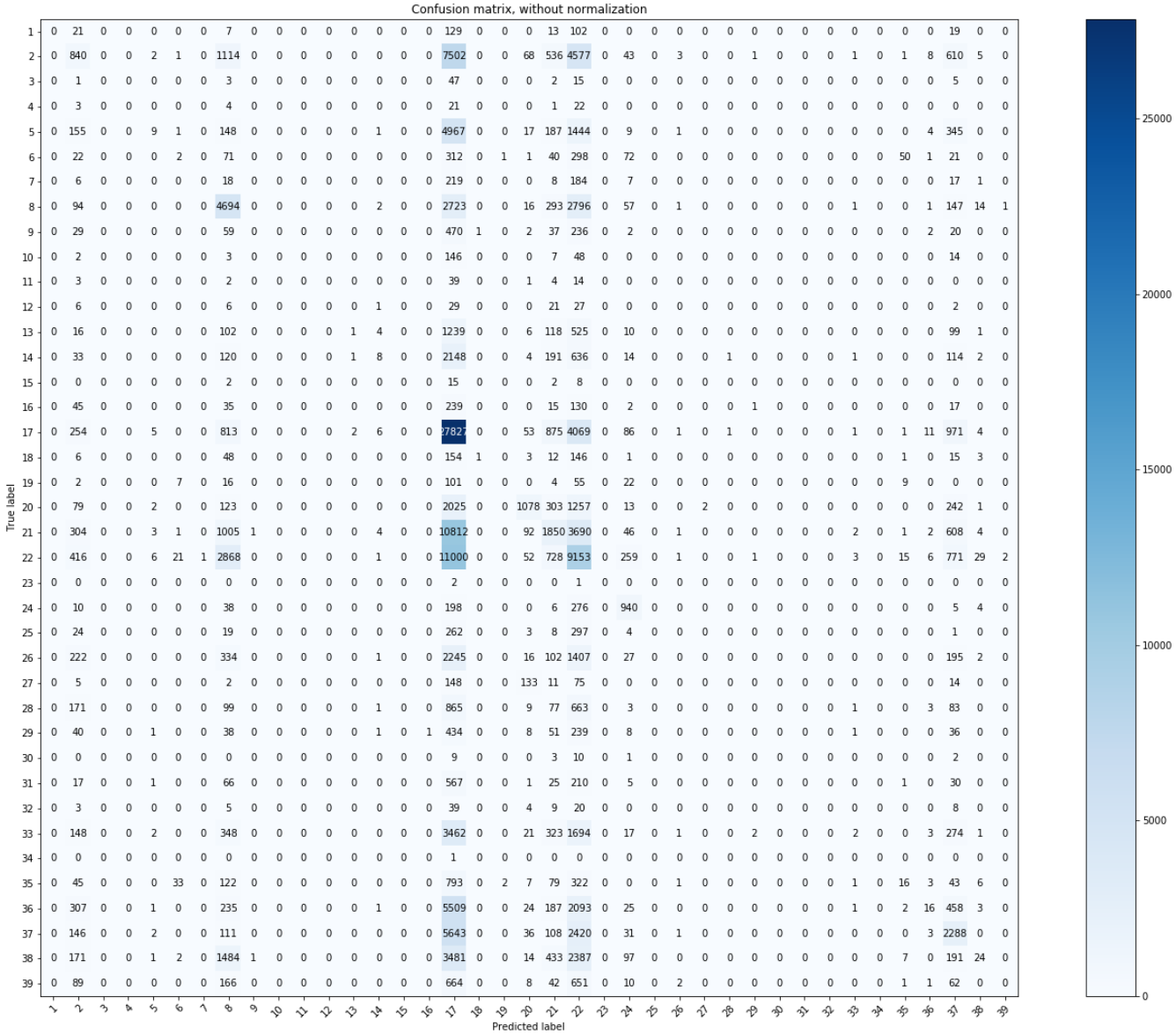


Anhang 8: Datenvisualisierung 3) Verbrechen nach Bezirk

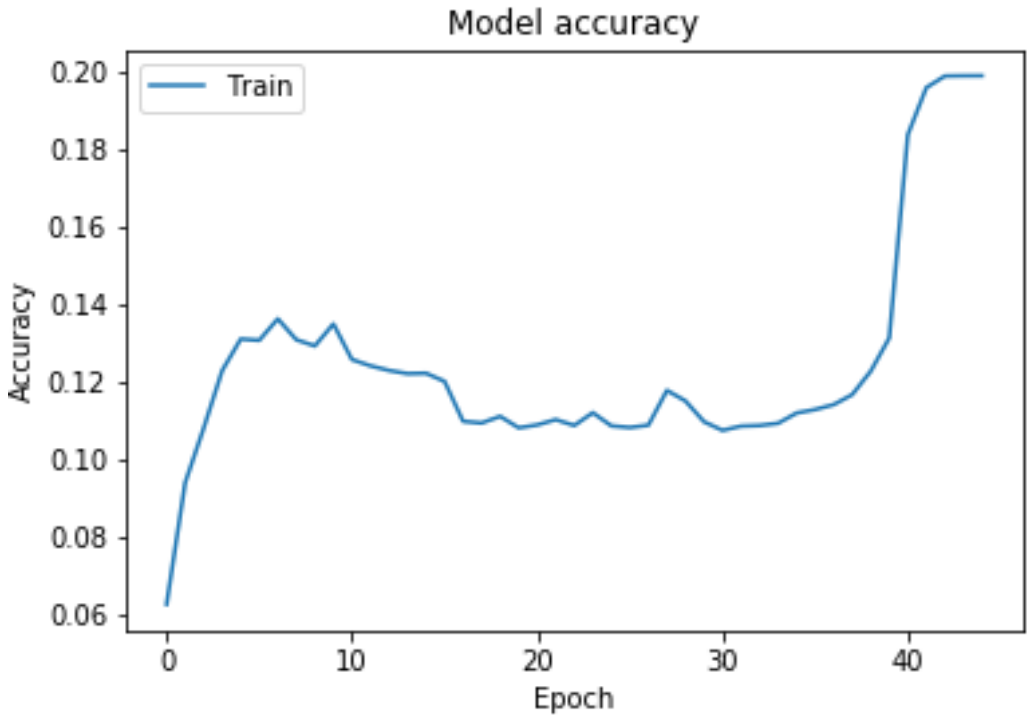


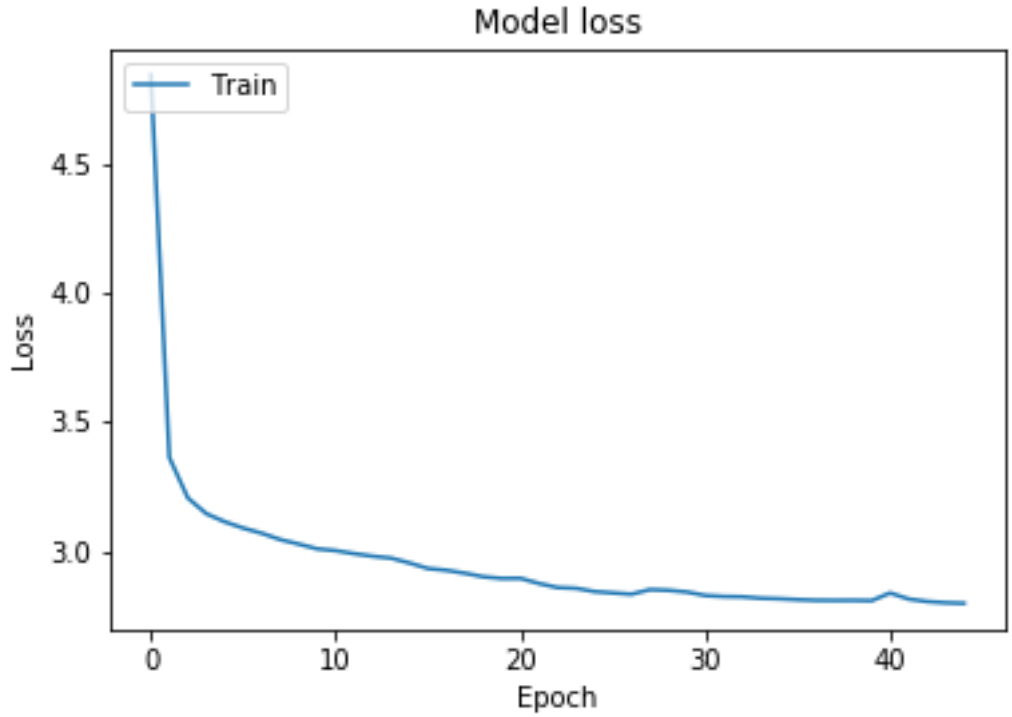
Anhang 9: Confusion Matrix Random Forest




Confusion matrix, without normalization
True label
Predicted label
25000
20000
15000
10000
5000
0

Anhang 10: Accuracy und Loss des Neuronalen Netzes





Anhang 11: Mapping der Klassen zur Verbrechenskategorie

1	ARSON
2	ASSAULT
3	BAD CHECKS
4	BRIBERY
5	BURGLARY
6	DISORDERLY CONDUCT
7	DRIVING UNDER THE INFLUENCE
8	DRUG/NARCOTIC
9	DRUNKENNESS
10	EMBEZZLEMENT
11	EXTORTION
12	FAMILY OFFENSES
13	FORGERY/COUNTERFEITING
14	FRAUD
15	GAMBLING
16	KIDNAPPING
17	LARCENY/THEFT
18	LIQUOR LAWS
19	LOITERING
20	MISSING PERSON
21	NON-CRIMINAL
22	OTHER OFFENSES
23	PORNOGRAPHY/OBSCENE MAT
24	PROSTITUTION
25	RECOVERED VEHICLE
26	ROBBERY
27	RUNAWAY
28	SECONDARY CODES
29	SEX OFFENSES FORCIBLE
30	SEX OFFENSES NON FORCIBLE
31	STOLEN PROPERTY
32	SUICIDE
33	SUSPICIOUS OCC
34	TREA
35	TRESPASS
36	VANDALISM
37	VEHICLE THEFT
38	WARRANTS
39	WEAPON LAWS