MODELING TRUCK SAFETY CRITICAL EVENTS: EFFICIENT BAYESIAN HIERARCHICAL STATISTICAL AND RELIABILITY MODELS

Miao Cai, M.S.

Draft on June 19, 2020

Dissertation Presented to the Graduate Faculty of
Saint Louis University in Partial Fulfillment
of the Requirements for the Degree of
Public Health Studies, Ph.D.

2020

COMMITTEE IN CHARGE OF CANDIDACY:

Professor Steven E. Rigdon, Ph.D.
Chairperson and Advisor

Professor Hong Xian, Ph.D.

Assistant Professor Fadel Megahed, Ph.D.

DEDICATION

To my parents, Zhimin Cai (蔡致民) and Guizhen Xu (徐桂珍), and my girlfriend Ziqi Peng. This work could not have been done without any of you.

ACKNOWLEDGEMENT

I want to thank my dissertation committee Drs. Steven E. Rigdon, Hong Xian, and Fadel Megahed. I want to express my special gratitude to my mentor and committee chair Dr. Rigdon for his continuous guidance, support, and encouragement in my doctoral program. Your knowledge, understanding, and teaching style inspired me to think and solve problems as a statistician. I would like to thank Dr. Xian for providing me insightful feedbacks on my hierarchical modeling, dissertation topics, and journal submission. I also want to thank Dr. Megahed for coordinating the data collecting, statistical modeling, and journal article writing. I enjoyed working with you both individually and in our group as a team.

I also want to thank my colleagues at the the Clinical Epidemiology Center in Veteran Affairs Saint Louis Health Care System: Dr. Ziyad Al-Aly, Benjamin (Charlie) Bowe, Yan Xie, Andrew Gibson, and Daniel Eaton. Thank you for giving me the opportunity to work with you, sharing your insights to world-class clinical epidemiology studies, and growing to be a better researcher in the field of clinical epidemiology. I can feel my exponential growth in data analytical skills, set up a good study framework for scientific research, and most importantly, understanding all the necessary steps to think as a clinical epidemiologist.

To my friends and classmates in the public health doctoral program, thank you (Asabe, Charlie, Eric, Longwen, Steve, Thembie, Xue, and Ucheoma) for creating a encouraging and mutually beneficial atmosphere for studying and research. To my colleagues in our college Drs. Echu Liu, Zhengmin Qian, Qiang Fu, Jen-jen Chang, Rhonda BeLue, and Travis Loux for your help. With all your contribution and collaboration, I was able to get several first-author publications during my three years of study here as a doctoral student. To my teammates in Auburn, Miami, and Johns Hopkins University, Amir, Mohammad, Qiong, Drs. Vinel and Davis, I enjoyed working on trucking data and solving problems with you.

I want to thank my family back in Wuhan, China: my parents Zhimin Cai and Guizhen Xu, for encouraging and supporting me to travel across the planet for higher education, as well as strictly staying at home in Wuhan to help preventing a potentially even worse COVID-19 outbreak; Ziqi, who has been extremely supportive and and always there when I feel lonely; my aunt and cousins Xiang Li and Yan Li, who have been taking care of my

parents while my absence during the 76-day draconian quarantine in Wuhan.

I want to thank China Scholarship Council, National Science Foundation of USA, and College for Public Health and Social Justice, Saint Louis University for sponsoring my traveling, tuition fees, and living expenses while studying in America. I also want to thank the DarkSky for providing five million free API calls to query weather information for this dissertation, the Ohio Supercomputer Center for providing free access to their resource that empowers the large data cleaning and simulation in this dissertation.

In this hard time of COVID-19 pandemic, I also want to thank all the doctors, nurses, epidemiologists, volunteers, relevant workers, as well as those residents who strictly stay at home, for standing up to sustain the transmission of the virus and saving lives across the world. Not only are we witnessing one of the most important turning point in human history, we also are changing the history with our brave move.

Finally, I want to note that this is not an exhaustive list of all the people I want to thank. A countless number of people consciously or unconsciously contributed to my growth but may not be listed here in this short section. I am grateful to to all of you!

TABLE OF CONTENTS

List of Figures

List of Tables

CHAPTER 1 INTRODUCTION

Transportation safety

Traffic safety is a pressing public health issue that involves huge losses of lives and financial burden across the world. As reported by the World Health Organization (WHO, 2018b), road injury was the eighth cause of death globally in 2016, killing approximately 1.4 million people, which consisted of about 2.5% of all deaths in the world. If no sustained action is taken, road injuries are predicted to be the seventh leading cause of death across the world by 2030 (WHO, 2018a). Compared to the victims who were claimed lives by diseases, people killed in traffic are mostly early- or middle-aged, particularly those aged 4 to 44 years old (Litman, 2013; Evans, 2014). Without traffic accidents, these victims could have much longer lives with normal quality of life

Apart from fatal deaths, road traffic injuries were also reported to be the cause of 50 million non-fatal life injuries and approximately 75.5 million disability-adjusted life years globally (Staton et al., 2016). In high-income countries, most of non-death costs were attributable to non-fatal crashes, with 2% of non-fatal events leading to over 40% of life-time medical costs (Ameratunga et al., 2006). Besides non-fatal injuries, traffic safety is a major economic burden. The global economic losses attributable to transportation safety were estimated to be 518 billion United States Dollars (USD), which accounted for 1% the gross domestic product (GDP) in low-income countries, 1.5% in middle-income countries, and 2% in high-income countries (Peden et al., 2004; Dalal et al., 2013).

The United States, Bureau of Labor Statistics (2017) reported that transportation contributed to the highest number of fatal occupational injuries, leading to 2,077 deaths and accounting for over 40% of all fatal occupational injuries in 2017. The National Safety Council reported that the number of deaths attributable to car crashes was at least 40,000 in 2018, which was the third straight year that this number was over 40,000 (The National Safety Council, 2018). Despite large amounts of investments in roads, improved vehicle

protection and traffic policy implementation, and advanced emergency and trauma care, the reduction in traffic associated fatality rates is nominal (Litman, 2013). If the change of traffic fatality rates in the US match those in other unremarkable countries, 20,000 traffic deaths could have been prevented each year (Evans, 2014).

Trucking safety

In the US, the large commercial truck industry is the backbone of the economy. Approximately 70% of freight is delivered via a truck at some point of their transportation, which account for 73.1% of value and 71.3% of volume of the domestic goods (Olson et al., 2016; Anderson et al., 2017). However, large trucks are the primary concern of traffic safety as they are associated with more catastrophic accidents. The FMCSA (2018a) reported that 27% fatal crashes in work zones involved large trucks: among all 4,079 crashes involving large trucks or buses, 4,564 people (1.12 people per crash) were killed. Large truck crashes approximately claim 5,000 lives and cause 120,000 injuries each year, but only 15% of these fatalities occur in the trucks, with a predominate 78% occurred in the other vehicles (Neeley and Richardson Jr, 2009). Besides, the economic losses associated with large truck crashes are also higher than those with passenger vehicles, with an estimated average cost of 91,000 USD per crash (Zaloshnja et al., 2008).

The high risk of large trucks is attributed to two aspects of reasons (Huang et al., 2013). First, large truck drivers need to drive alone for long routes, under on-time demands, and challenging weather and traffic conditions. Professional truck drivers usually need to work in shifts, and sometimes unavoidable late-night or early-morning shifts (Pylkkönen et al., 2015), which have been reported to be associated with sleep deprivation and disorders (Åkerstedt, 1988; Mitler et al., 1997; Solomon et al., 2004; Sallinen et al., 2005). Besides, the long route, constant concentration, and overtime work, intertwine with sleep deprivation and disorder and induce the fatigue symptoms among truck drivers. It is estimated that fatigue among long distance truck drivers caused up to 31% of single vehicle fatal truck crashes (National Transportation Safety Board, 1990; Mitler et al., 1997).

On the other hand, trucks have huge weights, large physical dimensions, and potentially carry hazardous cargoes. Large trucks can weight up to 80,000 pounds by federal law, which are twenty times as heavy as a normal-weight passenger vehicle (Department of Transportation, Utah, 2019). If these trucks travel at the speed of 65 miles per hour on the highway, it will take them around 525 feet to stop, which is about two times the length of a football field (Department of Transportation, Utah, 2019). The large physical size also creates large blind spots on both sides of the truck, which poses more threat on smaller-sized vehicles. When a crash occurs between a large truck and a smaller vehicle, the sheer size and weight of the truck result in the tragedy that the victims are from the smaller vehicle in around 80% of the cases (Neeley and Richardson Jr, 2009). Besides, commercial trucks crashes can cause massive casualties and regional public health emergency when carried hazardous materials are leaked (such as gasoline and sulfuric acid). The importance of truck industry and the potential catastrophic consequences underscore the need to reduce crash risk and improve the safety of truck transportation.

Modern trucking safety studies

To reduce the lives and economic losses associated with trucks, numerous studies attempted to accurately identify the risk factors for truck-related traffic crashes and make accurate prediction. However, there are several limitations of the studies using crash data. First, traffic crashes are characterized by rare events (dozens to thousands of times fewer crashes than non-crashes) (Theofilatos et al., 2016,0). To tackle this rare-event issue, the most common study design is a case-control study, which matches a crash with one to up to ten non-crashes, and then use statistical models such as logistic regressions to explain the causes or predict the crashes (Braver et al., 1997; Chen and Xie, 2014; Meuleners et al., 2015; Née et al., 2019). Unfortunately, a case-control study is limited in estimating incidence rates or overall average treatment effect. It may be contentious in selecting the ratio of controls to cases and how to select these controls (Grimes and Schulz, 2005; Sedgwick, 2014). Second, due to the retrospective nature of crash data, it is unrealistic to trace back

to the real-time traffic, weather, and other environmental factors that were associated with the crashes. Crash data reported by police and associated drivers were subject to recall and misinformation bias (Girotto et al., 2016). Third, crashes are underreported, especially for no- or minor-injuries crashes (Ye and Lord, 2011). It is estimated that 25% of minor-injury crashes and 50% of no-injury crashes were not reported, compared to nearly 100% reporting rate for fatal crashes (Savolainen et al., 2011).

Past truck safety literature almost exclusively focused on crashes, while ignoring precursors to crashes. A precursor to crashes, also known as safety critical events (SCEs), adverse events, or near-miss crashes, is an emerging pattern or signature associated with an increasing chance of crashes (Saleh et al., 2013; Janakiraman et al., 2016). Truck SCEs deserve more attention as they occur more frequently than crashes, potentially suggest fatigue, a lapse in performance, and potential catastrophic crashes (Dingus et al., 2006b).

Dissertation Overview

With the rapid development of modern technology, more real-time naturalistic driving data are available, which provide a unique opportunity to study real-world driving performance and potential consequences (Dingus et al., 2016). The overarching goal of this dissertation is to construct a generalizable analysis framework for trucking NDS data and understand how different factors impact truck driver performance. This work will:

(a) provide insights into the association between crashes and SCEs using commercial truck driver NDS data sets;

(b) serve as pattern to collect NDS and third-party API data, merging data sets from different sources, and aggregate into analyzable units;

(c) innovate Bayesian hierarchical statistical and reliability models specifically in transportation NDS setting.

The proposed analyzing framework and statistical models can be further used to optimize trucking routes and promote a safe commercial truck driving environment.

CHAPTER 2 LITERATURE REVIEW

This section reviews previous literature that is relevant to the key components of this dissertation: naturalistic driving studies, SCEs, the association between crashes and SCEs, risk factors for traffic safety, statistical models, and Bayesian statistics. I then propose the conceptual framework and specific aims of this dissertation.

Naturalistic Driving Studies (NDS)

Traditional truck crash prediction studies almost exclusively use data that ultimately trace back to post hoc vehicle inspection, interviews with survived drivers and witnesses, and police reports, which inherently have several limitations (Hickman et al., 2018; Stern et al., 2019). Firstly, truck crashes are extremely rare compared with non-crashes. According to the FMCSA (2018b), large truck and bus fatalities in 2017 were 0.156 per million traveled vehicle miles, which was a 6.8 percent increase from 2016. This rareness poses a challenge to statistical inference (Guo et al., 2010; Theofilatos et al., 2018). Secondly, truck crash data almost exclusively rely on post hoc police reports. Although these data are generally accurate and detailed, they are limited in determining the information of the driver in a meaningful time period leading up to the crash (Dingus et al., 2011). Critical factors, such as distraction, are not reported or cannot be determined due to a variety of reasons (Dingus et al., 2011), and these data were subject to recall bias even if they were reported. Thirdly, truck crashes are under-reported, particularly for no-injury and minor-injury crashes (Ye and Lord, 2011; Stern et al., 2019). It is estimated that 25% of minor-injury and 50% of non-injury crashes were not reported, while 100% of fatal crashes were reported (Savolainen et al., 2011).

Considering these limitations, a growing number of naturalistic driving studies have been initiated worldwide to improve data quality in traffic safety research (Klauer et al., 2009; Hickman et al., 2018; Guo, 2019). NDS use unobtrusive devices, sensors, and cameras to proactively collect frequent naturalistic driving behavior and performance data under real-

world driving conditions (Hickman et al., 2018; Guo, 2019). Compared with traditional reported crash data that are road segment-based, NDS collect driver-based data that are more useful in comparing the rates of SCEs since the traveling distance and time are collected. In addition, NDS data provide high-resolution driver behavior and performance data, which enable researcher to understand the data shortly prior to crashes or SCE without information bias or selection bias (Guo et al., 2010). Third, collecting naturalistic data is considerably less costly and difficult per observation compared to traditional crash data that involve often length interviews. Therefore, NDSs collect a large amount of data, which is both an opportunity and a challenge to researchers.

The first large-scale NDS was the 100-Car Naturalistic Driving study conducted by the Virginia Tech Transportation Institute (Neale et al., 2005; Dingus et al., 2006a). Other well-known NDS projects include the second Strategic Highway Research Program (Ghasemzadeh and Ahmed, 2017) and the UDRIVE NDS in Europe (Eenink et al., 2014; Barnard et al., 2016). There are also a few other NDS that target at specific populations, such as the 40-Teen NDS (Alden et al., 2016), the Older Driver Fitness-to-Drive NDS (Guo et al., 2015), and the Commercial Truck Driver NDS (Sparrow et al., 2016).

Safety-Critical Events (SCEs)

Instead of collecting extremely rare vehicle crash data, NDS focus on safety-critical events (SCEs), defined as events that used last-second successful evasive maneuver that avoided crashes (Dingus et al., 2011). Although near crashes or SCEs were not real crashes, a few studies suggested that they were correlated with crashes among cars (Dingus et al., 2006a; Guo et al., 2010; Dingus et al., 2011). The most commonly studied SCE is hard brakes (also knowns as hard-braking events or harsh braking), defined as a deceleration force higher than a pre-specified threshold, such as 0.3 g (Jansen and Simone Wesseling, 2018; Mollicone et al., 2019).

The rationale for using SCEs as surrogates for crashes is Heinrich's Triangle, which as-

sumes that less severe events are more frequent than severe events, and the frequency of severe events can diminish as that of less severe events decreases (Guo, 2019). The latter assumption can be quantitatively tested using crash and naturalistic driving data, but verifying the former assumption is challenging since the causal mechanism is complex and unknown (Guo et al., 2010). Applying SCEs in traffic safety studies to this Heinrich's Triangle can substantially increase the study sample size and may potentially enable the estimation of driving risk. However, a crucial question prior to the usage of SCEs in naturalistic studies is whether they are good surrogates of traffic crashes.

Guo et al. (2010) proposed two critical principles for using SCEs as surrogates for crashes: 1) similar or the same causal mechanisms between crashes and surrogates, 2) a strong association between the frequency of surrogates and crashes. Based on the 100-car database, they investigated the two principles using a sequential factor analysis, a Poisson regression, and a sensitivity analysis. The study concluded that using near crashes as surrogates for crashes will lead to conservative risk estimates but significantly reduce the variance of estimation. They suggested that using near crashes as surrogates in small-scale studies will be informative for evaluating the risk of crashes.

Association between crashes and SCEs

There has been a considerable number of studies evaluating the association between crashes and surrogate measures since the 1980s, with the general approach being estimating the conversion factor between the two types of events (Evans and Wasielewski, 1982,9; Cooper, 1984; Risser, 1985; Hydén, 1987). This topic has become a crucial issue as more naturalistic driving data sets are available to researchers in the recent decade. Table 2.1 below provides an overview of these studies on the relationship between SCEs and crashes, highlighting the sample size, driver types, driving locations, number of observed crash and surrogate events for the participating vehicles/drivers, statistical approach used, and statistically significant effects. Although the studies we collected in Table 2.1 use different surrogate measures of SCEs, they reached similar conclusions that their crash surrogates had

positive or zero association with crashes, except for speed alert events in Gitelman et al. (2018). Based on Table 2.1, there are four main gaps in the literature:

1. No studies examined the association between crashes and surrogates using NDS data sets that specifically target commercial truck drivers.
2. The sample sizes reported in those studies are limited, with the largest studies (Dingus et al., 2006a; Guo et al., 2010) examining 241 drivers. Thus, the number of reported crashes < 100.
3. The studies investigating the association between surrogates and crashes were confined to small geographic areas, which may limit the generalizability of the conclusions (Tsai et al., 2015).
4. The individual sample sizes in these studies were small and the surrogate measures were very different, which made it difficult to synthesize the evidence and to reach strong conclusions.

Table 2.1: *Characteristics of existing literature and this dissertation regarding the association between crashes and SCEs.*

Ref.	**Data set**	**Study size**	**Region**	**Statistical model**	**Crash surrogates (effect direction)**
Dingus et al. (2006a)	100-car	Drivers: 241 commuters Hours driven: 43,000 Miles driven: ∼2M Time frame: 1 year Crashes: 69 Surrogates: 761	Northern Virginia & Washington D.C., USA	95% confidence limits modeled using a Poisson distribution	Braking (↑) Steering (↑) Accelerating (↑)
Guo et al. (2010)	100-car	Drivers: 241 commuters Hours driven: 43,000 Miles driven: ∼2M Time frame: 1 year Crashes: 69 Surrogates: 761	Northern Virginia & Washington D.C., USA	Sequential factor analysis, Poisson regression	Near crashes (↑)

Continued on next page

Table 2.1 – continued from previous page

Ref.	Data set	Study size	Region	Statistical model	Crash surrogates (effect direction)
Gordon et al. (2011)	-	Drivers: 78 commuters Hours driven: NR Miles driven: ∼0.08M Time frame: 10 months Crashes: NR Surrogates: NR	Michigan, USA	Seemingly unrelated regression, Poisson regression	Lateral deviation (-) Lane-departure warning (↑) Time to edge crossing (↑)
Simons-Morton et al. (2012)	-	Drivers: 42 newly licensed teenagers Hours driven: NR Miles driven: NR Time frame: 18 months Crashes: 37 Surrogates: NR	Virginia, USA	Logistic regression using generalized estimating equations	Elevated gravitational-force (↑)
Wu and Jovanis (2012)	100-car	Drivers: 241 commuters Hours driven: 43,000 Miles driven: ∼2M Time frame: 1 year Crashes: 13 Surrogates: 38	Northern Virginia & Washington D.C., USA	Logistic regression	Yaw rate (-) Lateral acceleration (-)
Guo and Fang (2013)	100-car	Drivers: 102 young and high mileage Hours driven: 43,000 Miles driven: ∼2M Time frame: 1 year Crashes: 60 Surrogates: 7,394	Northern Virginia & Washington D.C., USA	Negative binomial regression	Critical-incident events (↑)
Wu et al. (2014)	100-car	Drivers: 90 commuters Hours driven: NR Miles driven: ∼1.1M Time frame: 1 year Crashes: 14 Surrogates: 182	Northern Virginia & Washington D.C., USA	Poisson regression	Run-off-road events (↑)
Pande et al. (2017)	-	Drivers: 33 commuters Hours driven: NR Miles driven: NR Time frame: 10 days Crashes: NA Surrogates: NA	California, USA	Negative binomial regression	High magnitude jerks while decelerating (↑)

Continued on next page

Table 2.1 – continued from previous page

Ref.	Data set	Study size	Region	Statistical model	Crash surrogates (effect direction)
Gitelman et al. (2018)	-	Drivers: 64 commuters Hours driven: NR Miles driven: NR Time frame: 1 year Crashes: NA Surrogates: NA	Israel	Negative binomial regression	Braking (↑) Speed alerts (↓)

Abbreviations: NR indicates that the parameter was not reported (or reported in combination with another parameter and hence cannot be inferred). M and B denote that the reported numbers are in millions and billions, respectively.

Risk factors for traffic safety

Fatigue

Fatigue has been the most pressing risk factor for truck crashes and SCEs. It is estimated that approximately 32% of drivers drive with fatigue over twice a month (National Sleep Foundation, 2008). American Automobile Association Foundation for Traffic Safety (2010) estimated that 16.5% of fatal traffic accidents and 12.5% of injuries-related collisions were associated with driving with fatigue in 2010 . The National Highway Traffic Safety Administration (NHTSA) estimated that 60% of fatal truck crashes were attributable to the driver falling asleep while driving (Craye et al., 2016; Cavuoto and Megahed, 2017). The FMCSA estimated that the causal role of fatigue is around five times higher in fatal than in property damage truck crashes (Knipling, 2017).

Although fatigue is the primary threat for traffic safety, preventing drowsy driving has not been effective since there is no simple way to measure fatigue driving (Dement, 1997). Fatigue is often defined as a multidimensional process that leads to diminished worker performance, which may be a result of prolonged work, psychological, and environment factors (Yung, 2016; Cavuoto and Megahed, 2017). However, this definition has low specificity since there are other factors associated with a decreased worker performance, and there is no uniform and succinct definition on fatigue. In view of the difficulty of measuring fa-

tigue, researchers attempted to use different proxies of fatigue, such as cumulative driving time, ocular and physiological metrics, sleep patterns, and night driving, but none of them has shown significant superiority.

Cumulative driving time has also been a measure of driver fatigue level, especially among NDSs. For example, Nakayama (2002) found that there was a significant increase in the fatigue of drivers after 12 hours of continuous driving. Jovanis et al. (2011) used cumulative hours of driving, time of the day, driving patterns over multiple days, rests after driving, and the 34-hour recovery policy as measures of driver fatigue. They found that more driving time was associated with increased odds of crashes among 686 less-than-truck-load drivers, with the highest odds in the 11-th hour. From the fifth hour to the 11th hour, the odds of crashes were consistently increasing. In contrast, Soccolich et al. (2013) found no significant difference in safety outcomes between the 11-th driving hour and 8-, 9-, or 10-th driving hours among truck drivers, but suggested a working day that starts with several hours of non-driving work and then followed by 14 hours of driving was significantly associated with SCEs. Mollicone et al. (2019) reported a significant association between predicted fatigue (estimated using cumulative driving time and time of the day) and the rate of hard-braking events (relative risk 1.078, 95% CI: 1.013-1.146).

Lack of sleep or specific sleep patterns have also been used as proxies of fatigue. For example, Chen et al. (2016b) used negative binomial regression to identify the association between four sleep patterns and driving performance based on the Naturalistic Truck Driving Study data. They revealed that shorter sleep, early-stage sleep in a non-work period, and insufficient sleep between 1 am and 5 am were associated with increased safety-critical event rates. Sparrow et al. (2016) reported that truck drivers with a restart break of only one nighttime period (defined as 1 am to 5 am) experienced more lapses of attention, elevated lane deviation at night, and higher sleepiness measured by subjective questionnaires.

Other studies emphasize the association between time of the day (such as night driving)

and fatigue development (Cavuoto and Megahed, 2017). Night driving is often accompanied by changes in shift scheduling, inadequate sleep, sleep apnea and disorder. For example, Pack et al. (1995) reported that the crashes caused by drivers falling asleep occurred primarily from mid-night to 7 am and from 2 pm to 4 pm. Mitler et al. (1997) found that drivers had an average of 5.18 hours of sleep in bed and 4.78 hours of electrophysiologically validated sleep per day, which were significantly less than needed to stay alert on job. Pahukula et al. (2015) investigated the association between five working time periods (early morning [12 am to 4 am], morning [5 am to 9 am], mid-day [10 am to 3 pm], afternoon [4 pm to 8 pm], and evening [9 pm to 11 pm]) and crashes in Texas, and found that different time periods contributed differently to the severity of injuries.

Driver characteristics

Young and old drivers have been reported to have higher risk of crashes. A review of age-related safety issues among professional heavy vehicle drivers suggested a U-shaped relationship: the chance of driving safety issues declines before 27 years old, plateau until the age of 63, and starts to grow up again after 63 (Duke et al., 2010).

Young drivers are much better in physical health and resistance to fatigue (Otmani et al., 2005). However, inexperience and reckless driving contribute to higher risk of accidents among young drivers. Clarke et al. (2006) suggested that young drivers (17–19 years old), especially males, have significantly more accidents than other drivers during the hours of darkness, on rural curves, and rear-end shunts compared to male drivers aged 20 -25 years. Campbell (1991) found that truck drivers under the age of 19 were over-involved in fatal accidents by a factor of 4, and those aged between 19 and 20 were over-involved by a factor of 6. Pack et al. (1995) revealed that the drivers under the age of 25 accounted for 55% of the 4,333 crashes in which the drivers were judged to be asleep whiling driving.

In contrast, older drivers have an increased risk of accidents for three reasons: impaired eyesight, prolonged reaction time to exogenous stimuli, and vulnerability to fatigue (Di Milia

et al., 2011). Aged drivers often have eyesight diseases or functionality impairment, such as cataracts, narrowed peripheral vision and decreasing visual acuity (Di Milia et al., 2011). Besides, working for truck companies often means irregular shifts and taking night schedules, which disrupt the circadian time-keeping systems, especially for aged workers (Moneta et al., 1996). It is indicated that the "critical age" of shiftwork intolerance is about 45 to 50 years, at which sleep disorder, persisting fatigue and digestive problems become the most prominent (Di Milia et al., 2011).

Traffic

Traffic is also viewed as an important risk factor for traffic safety. For the sake of availability and low cost, most prior studies used aggregated traffic data as proxies of traffic, such as annual average daily traffic. More recently, an increasing number of studies start to use real-time traffic data as a high-resolution proxy of traffic characteristics. Three published papers reviewed the impact of traffic variables on traffic safety issues (Wang et al., 2013; Theofilatos and Yannis, 2014; Roshandel et al., 2015).

Traffic variables include flow (traffic volume), occupancy/density, and speed (Wang et al., 2013; Theofilatos and Yannis, 2014). Traffic flow is defined as the number of vehicles passing through a specific road segment in a given unit time. Traffic occupancy or density is defined as the number of vehicles in a unit area of road at a moment. Speed can be computed from the road perspective as the mean speed of vehicles passing that road segment, or from the vehicle perspective as the vehicle speed. Compared to traffic flow and speed, traffic density is relatively less investigated due to a lack of relevant data.

For example, Dong et al. (2017) used vehicle and driver characteristics, traffic, environment, and road geometry to predict the frequency and severity of large truck-involved crashes. They found that the percent of large trucks, annual average daily traffic, driver condition, and weather characteristics were significantly associated with both crash frequency and severity. Theofilatos et al. (2018) used hourly aggregated traffic data, includ-

ing flow, occupancy, mean time speed, and percentage of trucks to predict crash occurrence with a bias-correction logistic regression. They found that the main risk factor average speed had a negative effect on crashes. Kamla et al. (2019) found that hard braking incidents were influenced by traffic and geometric variables in a similar fashion as crashes among large trucks at roundabouts.

Weather

Weather variables, including precipitation, visibility, wind speed, and temperature, have reported to have both direct and indirect effects on traffic safety events Theofilatos and Yannis (2014). Real-time extreme weather conditions such as heavy rain, fog, storm, and snow can either impair the driver's visual capability or reduce the safety of driving on the road (Baker and Reynolds, 1992; Chang and Chen, 2005; Al-Ghamdi, 2007). A positive linear relationship between precipitation and traffic accidents can be observed in both driver accidents and pedestrian accidents (Al-Ghamdi, 2007; Graham and Glaister, 2003). Naik et al. (2016) used ordinal and multinomial regression models with random-effects to investigate crash severity under various weather conditions. They found that wind speed, rain, humidity, and temperature were associated with single-vehicle truck crashes. Abdel-Aty et al. (2012) used detector and sensor data to successfully predict more than 70% of accidents with low visibility conditions.

In addition, the increase of ambient temperature places risks on occupational safety, and possibly leads to cognition loss, heat stroke, and impairment of wakefulness. Previous evidence showed that the risk of mistakes and SCEs are elevated in hot weather (Kjellstrom et al., 2009; Basagaña et al., 2015). Leard et al. (2015) found that for a day with temperature above 80 °F, there is a 9.5% increase in fatality rates compared with a day at 50-60 °F. A review by im Kampe et al. (2016) found that 11 out of the 13 included studies indicated an increase in unintentional injuries associated with high temperatures. In contrast, when low temperature is present, drivers are faced with snowy, foggy, and icy conditions, which substantially increase the risk of driving (Lemp et al., 2011). For example, Ahmed

et al. (2018) reported that truck-involved crashes were 19% more likely to occur than no truck-involved crashes when snow or strong wind were present.

Road characteristics

Commonly used road characteristics in traffic safety studies are the number of lanes, lane with, speed limits, horizontal curves, road curvature, and lighting conditions. The effect of the number of lanes and lane with on traffic safety is inconsistent in previous literature. Some studies suggest that the number of lanes is negatively associated with the risk of traffic accidents. For example, Zhu and Srinivasan (2011) found that crashes on roadways with more lanes tended to be less severe, which may result from the fact that more lanes give more space and separation between vehicles. They also reported that crashes in higher speed limit segments were more likely to be severe crashes. In contrast, Noland and Oh (2004); Kononov et al. (2008); Zhu and Srinivasan (2011); Islam et al. (2014) suggested that an increase in the number of lanes and lane width were positively associated with traffic fatalities, which could possibly be explained by an increased chance of lane-change-related conflict opportunities (Wang et al., 2013).

It is generally believed that lower speed limits can reduce the chances of traffic crashes, as well as the severity of crashes. For example, Neeley and Richardson Jr (2009) used state-level data from 1991 to 2005 to examine the association between truck-specific restrictions and fatality rates. They found that higher speed limits were associated with increased fatality rates, although different speed limits across vehicle types had no significant effect. Another study by Davis et al. (2015) also provided evidence that both overall and truck-involved fatalities were positively associated with maximum speed limits.

Road geometric features, such as road curvature and terrain type, were also reported to be risk factor of traffic safety events. Based on 1,787 truck-involved crashes from 1,310 highway segments in four years, Dong et al. (2015) found that AADT, segment length, degree of horizontal curvature, terrain type, land use and width, median type, right side shoulder

width, lighting condition, rutting depth, and posted speed limits were significantly associated with the likelihood of truck-involved crash frequency. Islam et al. (2014) found that crashes on roadway curved were associated with higher likelihood of major and possible injuries in urban single-vehicle large truck at-fault accidents, but this association is not statistically significant in multi-vehicle accidents.

Statistical models

Explanatory and predictive models

There are two cultures in current statistical or data science field: explanation and prediction (Shmueli et al., 2010; Breiman et al., 2001). The pro-explanation culture has long been adopted by most social science disciplines, such as epidemiology, economics, and psychology, in which researchers use generalized linear models, such as logistic regression and Poisson regression, to explain the association between the outcome and predictor variables. In contrast, the pro-prediction culture has recently been adopted in engineering and data science disciplines, in which they use black box algorithms such as neural networks to achieve high prediction accuracy. Pro-explanation models excel at explaining the association between predictors and the outcome variable and is unlikely to overfit the data. However, compared with black-box machine learning and deep learning algorithms, pro-explanation models are less likely to capture potential interaction between predictor variables, and therefore have less prediction accuracy. Traffic safety field has a pro-explanation culture, although it is shifting towards a pro-prediction given an emerging trend of innovative deep learning algorithms.

Binary classification models

The most commonly used statistical models in transportation safety research are logistic regression and Poisson regression. Logistic regression is commonly applied to predict crash likelihood (probability) using predictors such as driver features, weather, and traffic (Wang

et al., 2017). The parameterization of a binary logistic regression is shown in Model (2.1).

$$\begin{aligned} Y_i &\sim \text{Bernoulli}(p_i) \\ \log(\frac{p_i}{1-p_i}) &= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k, \end{aligned} \tag{2.1}$$

where Y_i is a binary variable that indicates whether an event occurred or not for the i-th observation. p_i is the mean parameter of a Bernoulli distribution, which is constrained on $[0, 1]$. The logit transformation of p_i then has the range from $-\infty$ to $+\infty$, which equals a linear combination of the predictors $x_1, x_2, \cdots, x_k$ and associated parameters $\beta_0, \beta_1, \cdots, \beta_k$.

The most commonly used outcomes for binary logistic regressions are injury versus non-injury crashes or fatal versus non-fatal crashes (Savolainen et al., 2011). For example, Chen et al. (2016a); Theofilatos et al. (2016); Ahmed et al. (2018). Roshandel et al. (2015) and Xu et al. (2015) provide excellent systematic reviews on traffic crash likelihood predictions. Other variants of binary logistic regression are binary probit models (Yu and Abdel-Aty, 2014), ordered logistic or probit models (Xie et al., 2009; Zhu and Srinivasan, 2011), multinomial logit models (Ye and Lord, 2011). Ordered logistic or probit regressions model an ordered multi-category outcome variable. The most common scenario is investigating different severity of crashes, such as no-injury crashes, minor-injury crashes, and fatal-injury crashes (Zhu and Srinivasan, 2011). These ordered models account for the ranked nature of different severity levels but make the proportional odds assumption (Rifaat et al., 2012). When the proportional odds assumption is violated, researchers often switch to multinomial logit or probit models, in which the outcome variable is considered as nominal.

Count outcome models

When researchers have crash data that are aggregated over a long time-period such as one year, it often makes sense to study the number of crashes instead of whether a crash occurred or not since they are often more than one crash. The most commonly used statisti-

cal model is therefore Poisson model, as it handles count data that are right-skewed, long tailed, and only have non-negative integer values. The parameterization of a Poisson regression is shown in model(2.2).

$$\begin{aligned} Y_i^\star &\sim \text{Poisson}(\mu_i) \\ \log \mu_i &= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k, \end{aligned} \tag{2.2}$$

where $Y_i^\star$ is the number of events for the i-th observation, which must be a non-negative integer. μ_i is both the mean and the variance parameter of the Poisson distribution, and it must be a non-negative numeric value. The logarithm of μ_i transforms μ_i into the range of $(-\infty, +\infty)$, which equals a linear combination of the predictors $x_1, x_2, \cdots, x_k$ and associated parameters $\beta_0, \beta_1, \cdots, \beta_k$. Poisson distribution assumes the mean parameter equals the variance, which is often violated in real-life data. When the variance of the data is greater than expected, it is called over-dispersion. Otherwise, it is called under-dispersion. Over-dispersion is much more common than under-dispersion in practice.

Other variants of a Poisson model include negative binomial models, quasi-Poisson models, and zero-inflated Poisson or negative binomial models (Lord, 2006; Mohammadi et al., 2014). Negative binomial or quasi-Poisson models are developed to account for the over-dispersion and under-dispersion issues in count data. Zero-inflated Poisson or negative binomial models are developed to account for the rare-event nature of traffic crash data (Lord et al., 2005,0; Washington et al., 2010; Dong et al., 2014). There is an excellent review paper on statistical models for crash frequency data by Lord and Mannering (2010).

Recurrent event models

Recently, recurrent event models have also been increasing applied to model the change in intensity of SCEs in the traffic safety field. For example, Liu et al. (2019) proposed to use a mixed-effects Poisson process to model unintentional lane deviation events, with the baseline intensity and time-varying coefficients modeled by penalized B-splines. They first conducted a simulation study to assess the performance of the proposed model with

different curvature of time-varying coefficients and the magnitude of event rate. Simulated 500 data sets with 500 shifts per set suggested satisfactory estimates for the true Gamma fragility parameter ϕ as estimated by an expectation–maximization algorithm, where larger values of ϕ indicated greater heterogeneity between shifts and more intense events. The bias ϕ in the simulation ranged from -0.01 to -0.09, which was around 2% smaller and 0.6% smaller than the true value in low and high event rate settings respectively. They applied the proposed model to 96 commercial truck drivers including 1,880 shifts. The study found that shifts with normal sleep time (7-9 hours) had a lower intensity compared with insufficient ($<$ 7 hours) and abundant ($\geq$ 9 hours) sleep time shifts.

Bayesian Inference

Traditional frequentist models that view parameters as unknown but fixed values. In the Bayesian perspective, parameters are viewed as random variables that have probability distributions (Gelman et al., 2013). Bayesian researchers need to specify subjective prior distributions on the parameters $p(\theta)$ before they collect data. After observing the data $\mathbf{X}$, the posterior distribution $p(\theta|\mathbf{X})$, which balances the prior beliefs and the observed data, can be calculated by the Bayes Theorem:

$$\begin{aligned} p(\theta|\mathbf{X}) &= \frac{p(\theta)p(\mathbf{X}|\theta)}{p(\mathbf{X})} \\ &= \frac{p(\theta)p(\mathbf{X}|\theta)}{\int p(\theta)p(\mathbf{X}|\theta)d\theta}, \end{aligned} \tag{2.3}$$

where the $p(\mathbf{X}|\theta)$ is the likelihood function, which reflects the data generating process that gives rise to the observed data. The denominator $\int p(\theta)p(\mathbf{X}|\theta)d\theta$ is a normalizing constant that force the posterior distribution to be integrated to one. The prior and likelihood function are specified by researchers, while normalizing constant in the denominator is the trickiest part of Bayesian estimation (Gelman et al., 2013; Kruschke, 2014). Modern Bayesian inference often uses numerical methods such as Markov chain Monte Carlo (MCMC) to directly sample from the posterior distribution. However, MCMC often fail or

take an inhibitively long time to reach a steady state in the case of high-dimensional data or tall data.

There are several strengths of Bayesian models over traditional Frequentist models. First, the probabilistic distribution of parameters, posterior credible intervals, and posterior predictive distributions account for the uncertainty in parameters and the data generating process, and they also have straightforward and intuitive interpretation. Second, Bayesian models incorporate prior information $p(\theta)$ into the statistical model, which can be useful when there is sufficient prior background information. This prior distribution is particularly useful for estimation in high-dimensional, sparse data, and complex model settings as these priors can solve convergence issues in traditional maximum likelihood estimation (MLE) (Betancourt and Girolami, 2015; McElreath, 2018). Lastly, Bayesian models are useful when researchers assume a complex data generating process: researchers only need to specify the priors and likelihood function, and MCMC will sample from the posterior parameter space (Lambert, 2018).

Hierarchical models

In Bayesian perspective, a hierarchical model is a statistical model with the probability distribution of one parameter depends on another parameter (Kruschke and Vanpaemel, 2015). Suppose we have a model with two parameters α, β and data D. The joint prior distribution of the two parameters is $p(\alpha, \beta)$. According to the Bayes Theorem, the posterior distribution is proportional to the product of the prior distribution and the likelihood function: $P(\alpha, \beta|D) \propto P(\alpha, \beta)P(D|\alpha, \beta)$. In a hierarchical model setting, the product can be factored as a chain of products among parameters, also known as conditional independence, such as $P(\alpha, \beta)P(D|\alpha, \beta) = P(D|\beta)P(\beta|\alpha)P(\alpha)$. In this parameterization, the parameter α is known as the hyperparameter because it gives rise to the parameter β (the parameter of a parameter) (Kruschke and Vanpaemel, 2015).

Hierarchical models have the advantage of partial pooling across different groups, which

shrinks group-level parameter estimates towards the group mean and shares information across groups (McElreath, 2018). Therefore, hierarchical model estimates are generally more robust to extreme observations and reasonably accurate for groups with sparse data (Gelman and Hill, 2006; Lambert, 2018). In the meanwhile, hierarchical models are particularly complex for Bayesian estimation using MCMC. It is notoriously difficult for MCMC to efficiently sample from the posterior distributions of hyperparameters due to the correlation between different levels of parameters and the sheer amount of parameters created by the hierarchical structure.

In the field of traffic safety predictive modeling, most studies assume that the sampling unit is a spatial-temporal road segment with relatively high rate of crashes during a period. However, it is not sufficient to only study high-risk occasions; we must also study non-crashes occasions and compare the two scenarios (Mehdizadeh et al., 2020). On the other hand, these studies that focus on road segments ignore driver-level unobserved effects, which is not ignorable in traffic safety studies. It is reported that the chance of having crashes for truck drivers with crash history in the past year is nearly twice as high as those without a crash history (Cantor et al., 2010). Most motor carrier insurance companies and employers also view historical safety events as an important measure of the driver's performance. Therefore, it is intuitive to use driver-focused hierarchical models to account for unobserved variation and characteristics (Huang and Abdel-Aty, 2010).

Markov chain Monte Carlo (MCMC)

In modern statistics, Bayesian inference relies on MCMC to overcome the intractable denominator issue in Bayes Theorem (Equation (2.3)). A *Monte Carlo simulation* is a technique to understand a target distribution by generating a large amount of random values from that distribution (Kruschke, 2014). A *Markov chain* has the property that the probability distribution of the observation i only depends on the previous observation $i-1$, not

on any one prior to observation $i-1$, as demonstrated in Equation (2.4).

$$P\left(X_i = x_i | X_1 = x_1, X_2 = x_2, \ldots, X_{i-1} = x_{i-1}\right) = P\left(X_i = x_i | X_{i-1} = x_{i-1}\right) \tag{2.4}$$

Integrating Markov chains and Monte Carlo simulations, the MCMC method can characterize an unknown unconditional distribution by sampling from the distribution, without knowing its all mathematical properties (Van Ravenzwaaij et al., 2018). In Bayesian estimation, MCMC is used to draw samples from the posterior distribution and calculate relevant statistics, such as posterior mean and credible intervals.

Metropolis-Hasting algorithm

The first MCMC algorithm is the Metropolis algorithm (Metropolis et al., 1953). It starts with a randomly defined initial value of the parameter θ. From a pre-defined symmetric proposal probability distribution $p(\theta|\mathbf{x})$, it then draw a proposal parameter value $\theta^{(\text{prop})}$, which only depends on the current parameter value $\theta^{(t)}$. This proposal value will be accepted with the probability of α defined in Equation (2.5). This proposal and acceptance with probability steps will be iterated for a pre-define number of times. When the Metropolis algorithm reaches a steady state, these proposal values are random values drawn from the posterior distribution of parameter θ, which can be used to describe and characterize the posterior distribution.

$$\alpha = \min\left(1, \frac{p(\theta^{(\text{prop})}|\mathbf{x})}{p(\theta^{(t)}|\mathbf{x})}\right) \tag{2.5}$$

After decades of successful empirical trials, Hastings (1970) proposed a more generalized form of the Metropolis algorithm, in which the proposal distribution can be arbitrary, but the acceptance probability $\alpha^{\star}$ is modified as shown in Equation (2.6). This Metropolis-Hasting (M-H) algorithm is the most widely-known MCMC algorithm used in different fields. Let $p(\ |\mathbf{X})$ be the posterior distribution we want to know, then the *Metropolis-Hasting algorithm* is:

1. Let $\theta^{(1)}$ denote an initial value for the continuous state Markov chain,
2. Set $t = 1$,
3. Let q be the proposal density which can depend on the current state $\theta^{(t)}$. Simulate one observation $\theta^{(\text{prop})}$ from $q(\theta^{(\text{prop})}|\theta^{(t)})$,
4. Compute the following probability:

$$\alpha^{\star} = \min\left(1, \frac{p\left(\theta^{(\text{prop})}|x\right)}{p\left(\theta^{(t)}|x\right)} \frac{q\left(\theta^{(t)}|\theta^{(\text{prop})}\right)}{q\left(\theta^{(\text{prop})}|\theta^{(t)}\right)}\right) \tag{2.6}$$

5. Set $\theta^{(t+1)} = \theta^{(\text{prop})}$ with the probability of $\alpha^{\star}$; otherwise set $\theta^{(t+1)} = \theta^{(t)}$. Set $t \leftarrow t+1$ and return to 3 until the desired number of iterations is reached.

Gibbs Sampler

Although the M-H algorithm is simple and powerful, its performance highly depends on the statistical structure and the proposal distribution. When there are a few parameters and the proposal distribution is not well-tuned, the M-H algorithm will have a very low acceptance rate, which can be extremely inefficient. In view of this issue, Gibbs sampler was proposed with the idea that the proposed values are always accepted and each parameter is updated one at a time by generating samples from the conditional distributions (Geman and Geman, 1987; Gelfand and Smith, 1990; Lambert, 2018). Suppose $= [\theta_1, \theta_2, \cdots, \theta_k]$ is a k-dimensional parameter. Let $\mathbf{X}$ denote the data. The *Gibbs sampling* algorithm is then:

1. Begin with an estimate $^{(0)} = [\theta_1^{(0)}, \theta_2^{(0)}, \cdots, \theta_k^{(0)}]$ in the parameter space,
2. Set $t = 1$,
3. Simulate $\theta_1^{(t)}$ from $p(\theta_1|\theta_2^{(t-1)}, \theta_3^{(t-1)}, \cdots, \theta_k^{(t-1)}, \mathbf{X})$,
4. Simulate $\theta_2^{(t)}$ from $p(\theta_2|\theta_1^{(t)}, \theta_3^{(t-1)}, \cdots, \theta_k^{(t-1)}, \mathbf{X})$,
5. $\cdots$,
6. Simulate $\theta_k^{(t)}$ from $p(\theta_k|\theta_1^{(t)}, \theta_3^{(t)}, \cdots, \theta_{k-1}^{(t)}, \mathbf{X})$,
7. Set $t \leftarrow t+1$ and repeat steps $3-6$ for a pre-specified number of iterations and make sure the Gibbs sampler reaches the steady state after sufficient iterations.

Hamiltonian Monte Carlo (HMC)

The M-H algorithm and Gibbs sampler gained popularity in the recent 30 years among applied researchers with the availability of open-source softwares such as `R` and `BUGS` . However, as the size and dimensionality of data are growing exponentially, the performance of the two algorithms is still not satisfactory: when there are high dimensional data where high-probability regions are concentrated on an very small sample space, it is extremely them to generate effective samples from these small posterior regions (Barp et al., 2018). Hierarchical models even complicate this issue by adding random parameters for each subgroup and correlation between parameters, which further grows the dimensionality of parameter space. All the problems motivate researchers to develop more efficient samplers or algorithms to estimate Bayesian models for big data.

Originally proposed by Duane et al. (1987) with the name of Hybrid Monte Carlo, the Hamiltonian Monte Carlo (HMC) modifies the random-walk behavior in M-H algorithm into a deterministic one by adding auxiliary momentum parameters m, by which it generates distant and effective proposals by taking advantage of the Hamiltonian system (Betancourt, 2017). Here is the general idea of the *HMC* (Lambert, 2018):

1. Let $\theta^{(0)}$ denote a random initial value from a proposal distribution,
2. Set $t = 1$,
3. Generate a random initial momentum m from a proposal distribution (typically a multivariate normal distribution),
4. Use the leapfrog algorithm to solve the trajectory moving over the high-density posterior parameter space under the Hamiltonian mechanism for a period,
5. Calculate the new momentum m' and new position $\theta^{(\text{prop})}$
6. Compute the following probability:

$$\alpha^H = \min\left(1, \frac{p\left(\theta^{(\text{prop})}|x\right) p(\theta^{(\text{prop})})}{p\left(\theta^{(t)}|x\right) p(\theta^{(t)})} \frac{q(m')}{q(m)}\right) \tag{2.7}$$

7. Set $\theta^{(t+1)} = \theta^{(\text{prop})}$ with the probability of α^H; otherwise set $\theta^{(t+1)} = \theta^{(t)}$. Set $t \leftarrow t+1$ and return to 3 until the desired number of iterations is reached.

Conceptual framework

Figure 2.1 below shows the conceptual framework used in this study, which is based on (a) *truck driver fatigue model* (Crum and Morrow, 2002), (b) 5×ST-level hierarchy theory (Huang and Abdel-Aty, 2010), and (c) Commercial motor vehicle driver fatigue framework (Stern et al., 2019). This frame in Figure 2.1 has a two-level hierarchy structure: driver level and trip level. Driver level factors include driver features and fatigue; trip level factors include traffic, road geometry, and weather. These factors are assumed to be directly associated with SCEs, which can be modeled by statistical and reliability models. Final, the SCEs are hypothesized to be directly associated with crashes.

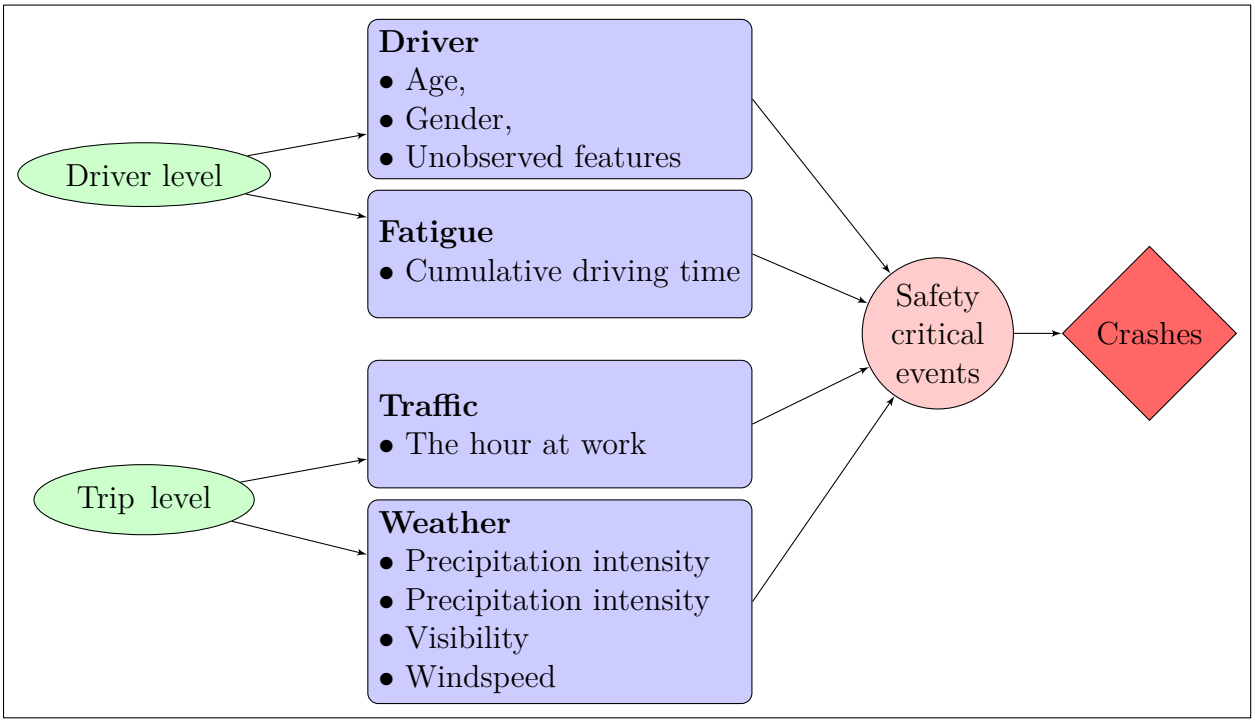


Figure 2.1: *Conceptual framework of driver- and trip-level risk factors on safety critical events and crashes*

Specific Aims

The overarching goal of this proposed dissertation is to construct a generalizeable framework of data collection, aggregation, and statistical modeling for NDS data and under-

stand how various risk factors impact the performance of truck drivers. In the literature section, several gaps in traffic safety studies have been identified: (a) an increasing number of studies are using SCEs as the outcome variable, but the association between crashes and SCEs has not been well studied among truck drivers; (b) there is no consistent framework to analyze the high-resolutional and high-dimensional data collected by NDS; (c) SCEs are much more common than crashes and there can be multiple SCEs within a short period, but recurrent events models were not widely applied in the field of transportation safety to understand these SCEs. Accordingly, the specific aims of this dissertation are:

1. **Aim 1: To examine the association between truck crashes and SCEs using Bayesian negative binomial regression models**. I hypothesize that the rate of crashes is positively associated with the rate of SCEs among truck drivers controlling for the miles driven and other covariates.
2. **Aim 2: To construct three scalable hierarchical models to identify potential risk factors for SCEs**. I hypothesize that the patterns of SCEs vary significantly from drivers to drivers and can be predicted using risk factors including cumulative driving time, weather, age, gender, speed, speed variation, and others.
    - **Sub-aim 2(a): to construct hierarchical logistic regressions to model the probability of SCEs in 30-minute intervals**. I hypothesize that the probability of SCEs is positively associated with the cumulative driving time and risk factors, and it varies significantly from drivers to drivers.
    - **Sub-aim 2(b): to construct hierarchical negative binomial regressions to model the rate of SCEs in 30-minute intervals**. I hypothesize that the rate of SCEs is positively associated with the cumulative driving time and risk factors, and it varies significantly from drivers to drivers.
3. **Aim 3: to study the pattern of SCEs within shifts using recurrent event models**. I hypothesize that SCEs are more frequent in later stages of shifts, and it varies from drivers to drivers.

- **Sub-aim 3(a): to construct a Bayesian hierarchical non-homogeneous Poisson process with the power law process intensity function to model the intensity change of SCEs within each shift**. I hypothesize that the intensity of SCEs increases in later stage of shifts, can be predicted by the risk factors, and varies from drivers to drivers.
- **Sub-aim 3(b): to propose an innovative reliability model that accounts for both within shift cumulative driving time and between-trip rest time**. I hypothesize that the intensity function can be recovered by some proportion or by some amounts during rests between trips, and intensity function varies significantly from drivers to drivers.

CHAPTER 3 METHODS

Data sources

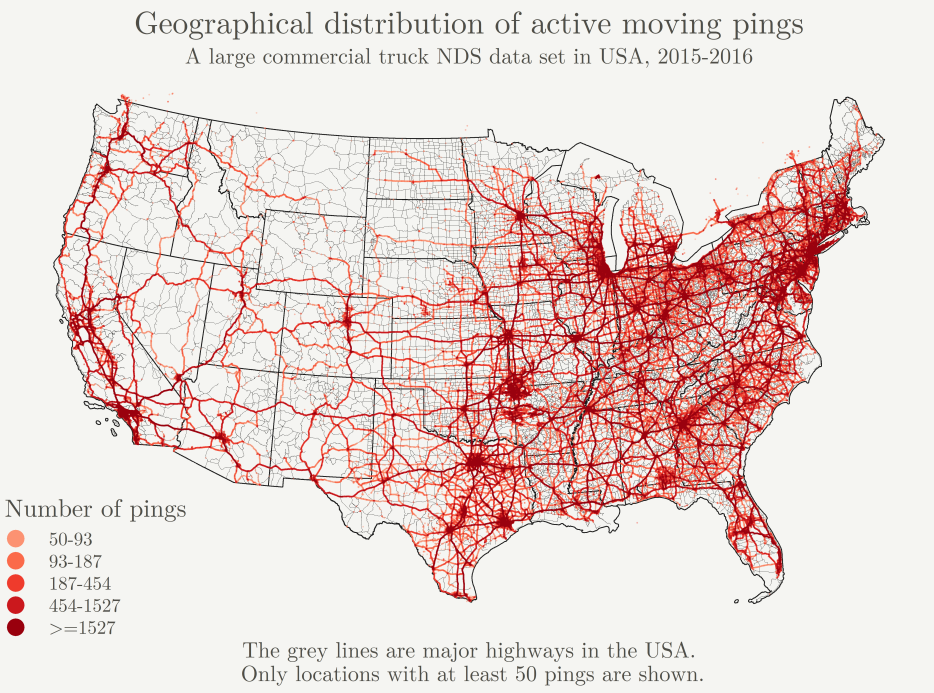
Real-time ping

A real-time *ping* data set was collected by a North American commercial trucking company between April 1st, 2015 and March 29th, 2016. A small device was installed in each of these trucks and intermittently collect real-time driving ping records. The time between two neighbor pings ranges between every couple of seconds to approximately 15 minutes. Over 50% of the time intervals between two pings were less than 5 minutes and over 95% of them were less than 15 minutes. Each ping (Table 3.1) will collect data on date and time, latitude, longitude, driver identification number (ID), and speed. In total, 1,494,678,173 pings were collected from 31,828 truck drivers, with 98.7% of the pings were "good quality" according to the GPS quality indicator.

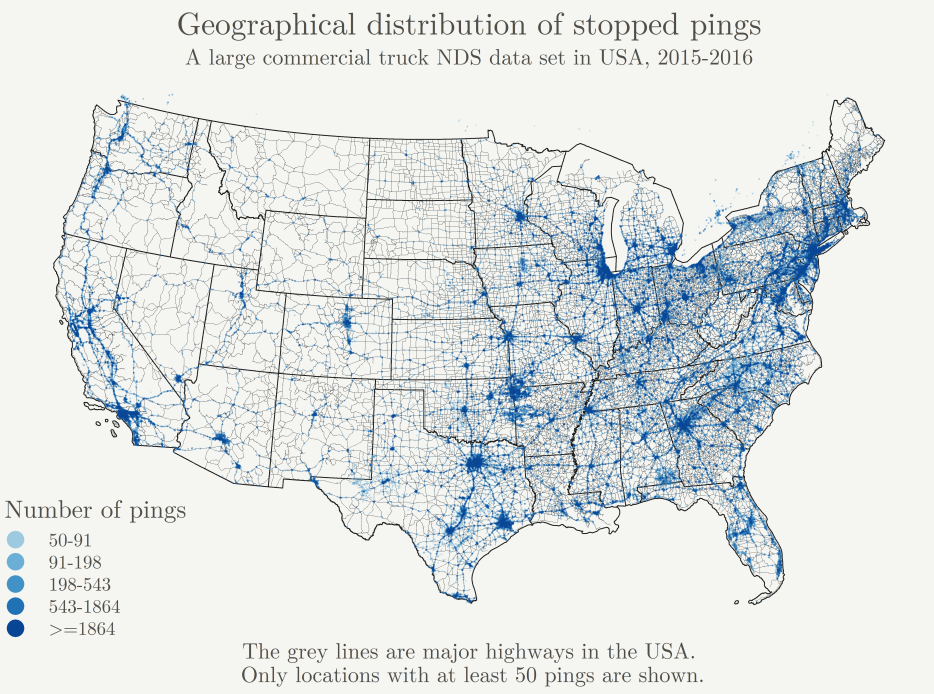
Table 3.1: *A sample of the ping data set*

ping time	speed	latitude	longitude	driver ID
2015-10-23 08:09:26	5	33.94288	-118.1681	driver1
2015-10-23 08:22:58	4	33.97146	-118.1677	driver1
2015-10-23 08:23:12	8	33.97178	-118.1677	driver1
2015-10-23 08:23:30	4	33.97233	-118.1678	driver1
2015-10-23 08:38:00	40	34.00708	-118.1798	driver1

Figure 3.1 below shows the geographical locations of these ∼1.5 billion pings, where the active (speed $>$ 0 MPH) and inactive (speed $=$ 0 MPH) pings are depicted in Figures 3.1a and 3.1b, respectively. Note only locations with at least 50 pings are displayed to make the maps look less overwhelming . Both Figures 3.1a and 3.1b utilize a sequential color scheme, where a darker color indicates a higher number of pings. The geographical point patterns suggest that most of the trucking transportation closely matches the U.S. population density distribution (i.e., it is more concentrated along the coasts). The active and inactive pings are generally consistent, but active pings are more concentrated in major midwest roads.



(a) *Active pings captured from the 31,828 truck drivers from April 1, 2015 to March 31, 2016.*



(b) *Inactive pings captured from the 31,828 truck drivers from April 1, 2015 to March 31, 2016.*

Figure 3.1: *Geographical point patterns of moving and stopped pings generated by the 497 sample drivers.*

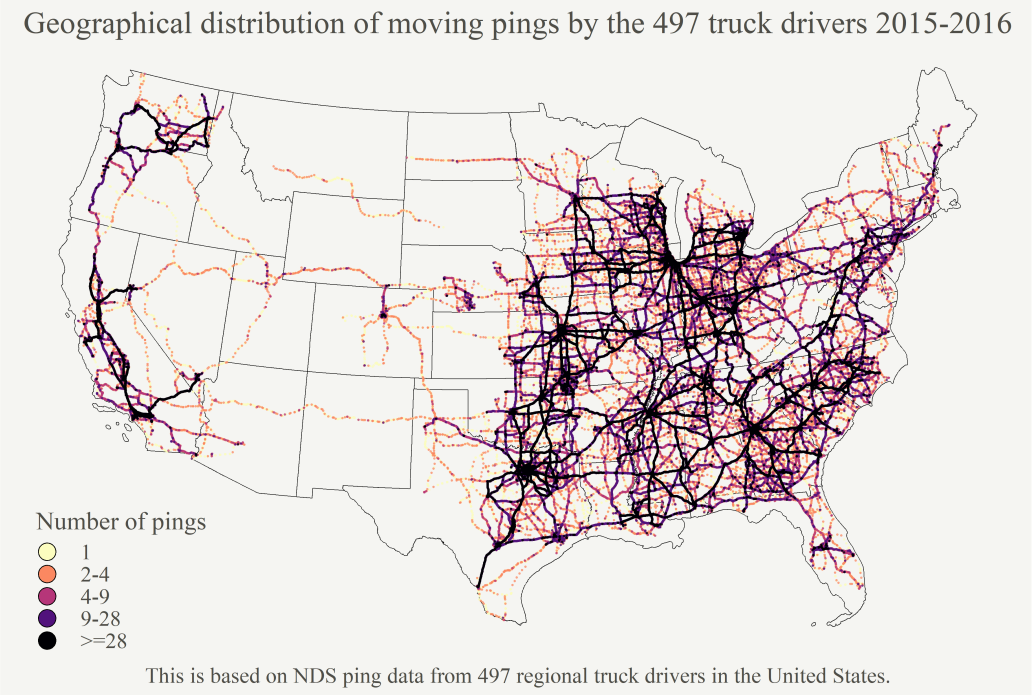
My aim 1 will use all the 1.5 billion ping data, while aim 2 and 3 will use a sample of 496 regional drivers, which generated around 13 million real-time ping data. Similarly, Figure 3.2 below demonstrates the geographical point patterns of the 13 million pings, where active and inactive pings are shown in Figure 3.2a and 3.2b. The two maps show a similar pattern with Figure 3.1: the majority of the transporting tasks was in the middle and east parts, with a few in the west (California and Seattle), while very few points were in the Midwest. The coverage of locations in almost all populous cities in the U.S. makes the sample in this study representative of the regional driving tasks in this country.

Truck crashes and SCEs

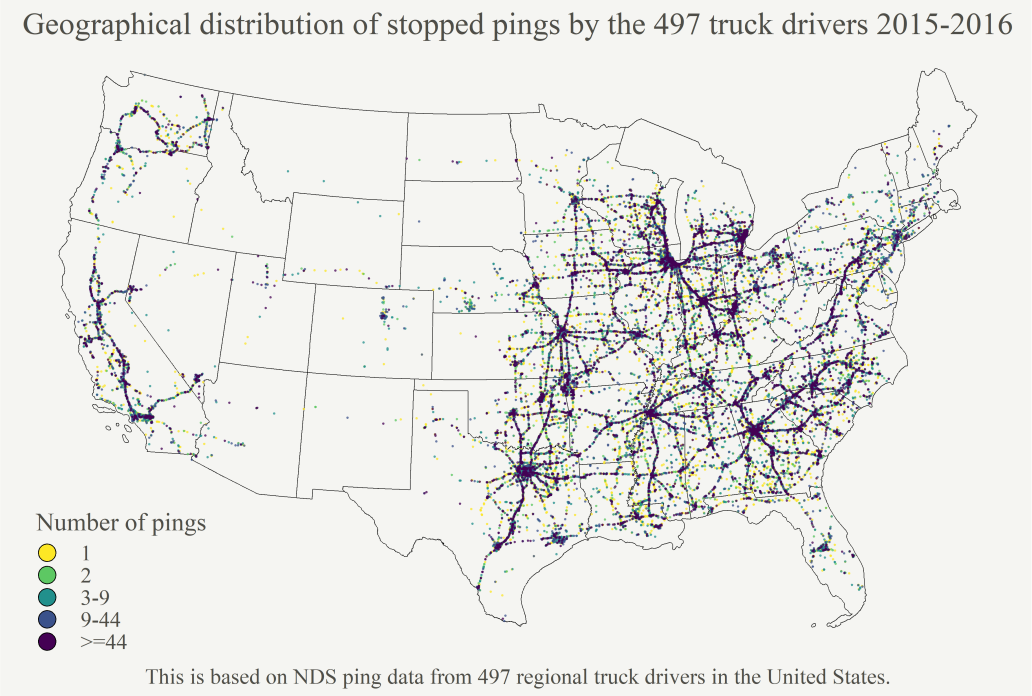
Real-time SCEs and associated GPS locations for all trucks in the same period were also collected by a sensor-based monitoring system. Four types of SCEs were recorded:

- *Headway*, which signals an instance of tailgating for at least 118 seconds at an unsafe gap time (a measure of distance between leading and trailing vehicles of 2.8 seconds or less (Grove et al., 2015).
- *Hard brakes*, which are defined as instances of deceleration rate 9.5 miles per hour per second or more.
- Activation of the *rolling stability* system, which intervenes by applying brake pressure (in addition to potentially applying trailer pressure) assisting the driver in aligning the vehicle when the system's critical thresholds are approached (Bendix®, 2007).
- Activation of the *forward collision mitigation system*.

Once the kinematic thresholds regarding the driving behavior were met, the sensor will be automatically triggered and the information of these SCEs (latitude, longitude, speed, driver ID) will be recorded. A sample of the SCEs data and crashes are demonstrated in Table 3.2 and Table 3.3.



(a) *Active pings*



(b) *Inactive pings*

Figure 3.2: *Geographical point patterns of moving and stopped pings generated by the 497 sample drivers.*

Table 3.2: *A sample of SCEs data set*

driver	event time	event type
driver1	2015-10-23 14:46:08	HB
driver1	2015-10-26 15:06:03	HB
driver1	2015-10-28 11:58:24	HB
driver1	2015-10-28 17:42:36	HB
driver1	2015-11-02 07:13:56	HB

Table 3.3: *A sample of the truck crashes table*

Open date	Open time	Driver	Type	Cause	# of injuries	# of Fatalities
2014-06-10	22:00:00	driver1	L13	99	0	0
2014-06-18	10:52:00	driver1	L13	1	0	0
2014-10-02	13:38:00	driver1	L13	1	0	0
2014-09-04	19:46:00	driver1	L13	1	0	0
2014-09-22	05:00:00	driver1	L13	1	0	0
2014-10-23	07:00:00	driver1	L25	1	0	0
2015-11-04	13:01:00	driver1	L70	3	0	0

Driver demographics

A demographic table including the birthday, gender, race of each driver, as well as driver type and business units is available. These demographic data can be merged back to the trips, shifts, and crashes tables using a common unique driver ID.

Driver types include: (a) local drivers who transport freight within a 200-mile radius and return home on the same day, (b) regional drivers moving freights in regional routes that may include several surrounding states, and (c) over-the-road drivers who specialize in hauling freight long distances, requiring them on the road for days/weeks.

Business units include: (a) dedicated contract carriage, in which trucks and drivers are assigned to a singular customer with familiar routes, task and work duties; (b) intermodal freight, in which the freight is transported in intermodal containers between shipping ports, rail terminals, and inland shipping docks; and (c) final-mile delivery, in which non-conveyable products are delivered to customers.

Weather data from the Dark Sky API

Weather data including *precipitation intensity*, *precipitation probability*, *wind speed*, and *visibility* for the 496 sample regional drivers, were retrieved from a third-party weather

Table 3.4: *A sample of weather data from the DarkSky API*

ping time	latitude	longitude	precipitation intensity	precipitation probability	wind speed	visibility
2015-10-23 08:09:26	33.94	-118.16	0	0	0.21	9.82
2015-10-23 08:22:58	33.97	-118.16	0	0	0.22	9.81
2015-10-23 08:23:12	33.97	-118.16	0	0	0.22	9.81
2015-10-23 08:23:30	33.97	-118.16	0	0	0.22	9.81
2015-10-23 08:38:00	34.00	-118.17	0	0	0.24	9.81

data provider *the Dark Sky API*. This API allows users to query historic minute-by-minute weather data anywhere on the globe (The Dark Sky API, 2019a). According to the official document, the `Dark Sky API` is supported by a wide range of weather data sources, which are aggregated together to provide the most precise weather data possible for a given location (The Dark Sky API, 2019b).

The latitude and longitude coordinates of the pings from these 496 sample drivers are rounded to two decimal places, which are worth up to 1.1 kilometers. In the meanwhile, the time of these pings are rounded the nearest hour and ignore those stopping pings. This rounding algorithm scales the original 13 million real-time ping data down to around five million unique latitude-longitude-date-time combinations, which significantly reduces the financial cost and querying time. I used the R package `darksky` to obtain weather variables for these reduced five million unique combinations (Rudis, 2018). The weather data for these combinations will then be merged back to the original ping data using the unique latitude-longitude-date-time combinations. A sample of the weather data is shown in Table 3.4

Data aggregation

To convert the original real-time ping data into analyzable units, I aggregated them into *shifts*, *trips*, and *30-minute intervals*, which are inspired by real world truck transporting practice and the hours-of-service policy by Federal Motor Carrier Safety Administration (2017). *Shifts* are on-duty periods with no breaks longer than eight hours (there can be short breaks less than 8 hours). *Trips* are continuous driving periods with no breaks less than half an hour. These trips are further divided into *30-minute fixed intervals*. This is

because the length of trips can vary from several minutes to several hours, which are not good homogeneous analyzable units for statistical modeling.

Figure 3.3 visually present the data aggregation process of ping → shifts → trips → 30-minute intervals, as well as the nested structure. The y-axis is speed and x-axis is global time. Each dot is a ping, and the color of that ping indicate the current speed. Grey dots indicate stopping pings with the current speed of zero. The arrows in the lower part represent the aggregated shifts (blue), trips (purple), and 30-minute intervals (green). The long blue arrows (shifts) are separated and defined by long grey dots (more than eight hours) in the middle of the figure. Similarly, the shorter purple arrows are separated and defined by shorter grey dots (greater than half an hour but less than eight hours). The shortest green line segments (30-minute intervals) are defined by the start and end time of the purple arrows, and these 30-minute intervals are much more homogeneous in length than shifts and trips.

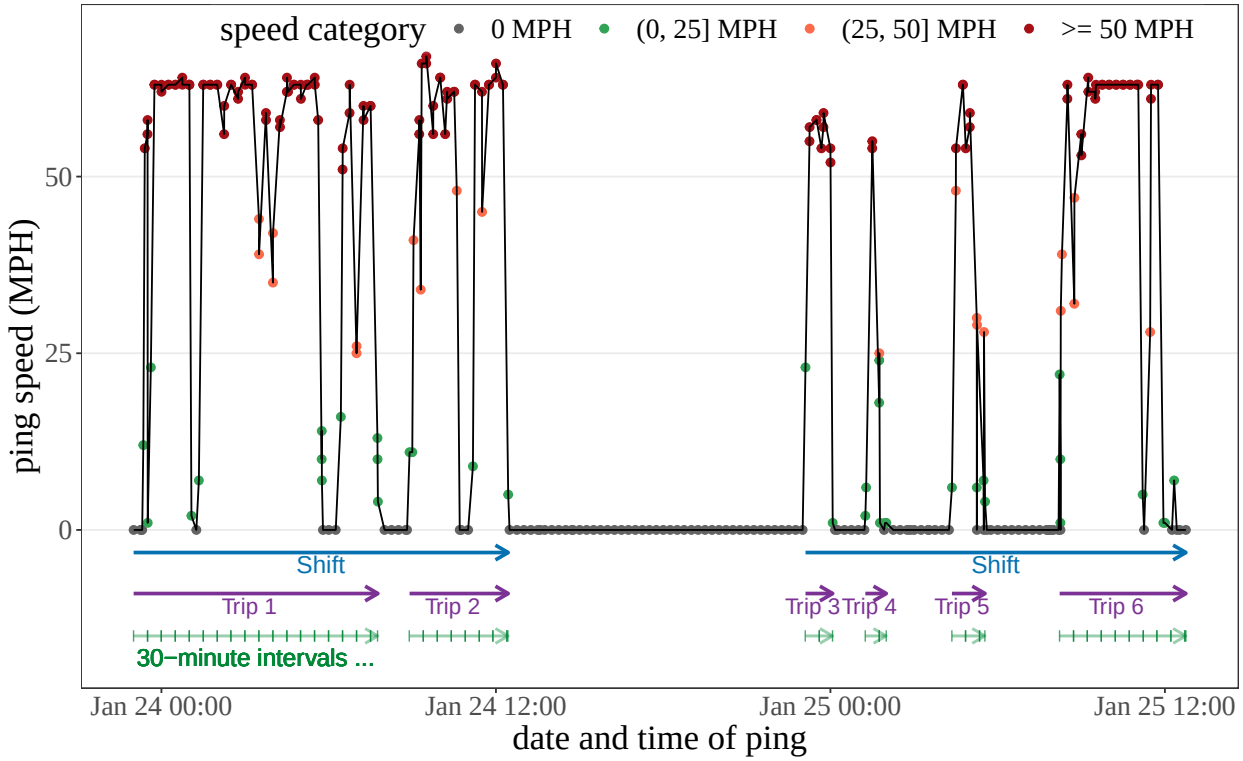


Figure 3.3: *Data aggregation process from pings to shifts, trips, and 30-minute intervals.*

The details of the aggregation algorithm and the aggregated data sets are shown in the following subsections.

Shifts

For each of the sample drivers, if the ping data showed that the truck was not moving (ping speed = 0) for more than eight hours, the pings were separated into two different *shifts* on the two sides of this long break. There could be several short breaks (less than eight hours) within each shift. A sample of the aggregated shifts data is shown in Table 3.5.

Table 3.5: *A sample of transformed shifts data*

driver	shift ID	shift start	shift end	shift length (minutes)
driver1	1	2015-10-23 08:09:26	2015-10-23 18:37:56	628
driver1	2	2015-10-26 07:49:04	2015-10-26 15:06:58	437
driver1	3	2015-10-27 01:59:48	2015-10-27 07:58:56	359
driver1	4	2015-10-28 08:05:08	2015-10-28 20:20:32	735
driver1	6	2015-10-30 09:27:12	2015-10-30 21:18:22	711

Trips

For each shift, if the ping data showed that the truck was not moving (ping speed = 0) for more than half an hour, the ping data were separated into different trips. These ping data were then aggregated into different *trips*. The drivers are assumed to be fully driving within each trip since there are not breaks longer than 30 minutes within each trip. The trips are nested within shifts. A sample of the trips data is shown in Table 3.6.

Table 3.6: *A demonstration of transformed trips data*

driver	trip id	start time	end time	trip time	distance (miles)
driver1	100160724	2015-10-23 08:09:26	2015-10-23 08:37:26	28	4.473
driver1	100160725	2015-10-23 09:04:24	2015-10-23 11:21:24	137	46.721
driver1	100160726	2015-10-23 12:00:36	2015-10-23 15:37:36	217	164.576
driver1	100160727	2015-10-23 16:38:10	2015-10-23 18:37:10	119	52.907
driver1	100160728	2015-10-26 07:49:04	2015-10-26 10:52:04	183	104.085

30-minute intervals

As the length of a trip can vary significantly from 5 minutes to more than 8 hours, each trip is further decomposed into *30-minute intervals* fixed intervals according to the start

and end time of the trip. The last interval of the trip is typically less than 30 minutes. The 30-minute interval data will dissect unnecessarily lengthy trips into small chunks and enable statistical analyses based on these small-interval data. The 30-minute intervals are nested within trips. A sample of the 30-minute interval data is shown in Table 3.7.

Table 3.7: *A sample of transformed 30-minute intervals*

driver	interval id	start time	end time	interval time	distance (miles)
driver1	197089	2015-10-23 08:09:26	2015-10-23 08:38:00	28	4.538
driver1	197090	2015-10-23 09:04:24	2015-10-23 09:34:24	30	2.645
driver1	197091	2015-10-23 09:34:24	2015-10-23 10:04:24	30	0.984
driver1	197092	2015-10-23 10:04:24	2015-10-23 10:34:24	30	5.928
driver1	197093	2015-10-23 10:34:24	2015-10-23 11:04:24	30	17.348

Data merging

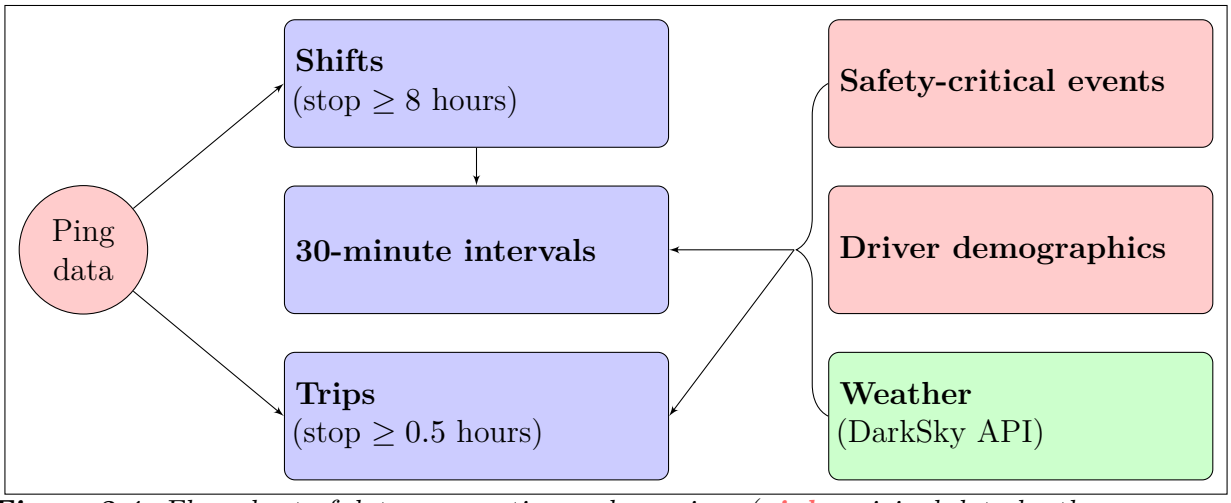


Figure 3.4: *Flow chart of data aggregation and merging. (**pink**: original data by the company; **blue**: aggregated data; **green**: third-party API data.)*

Figure 3.4 demonstrates the data aggregation and merging workflow. The left part shows the data aggregation from the original ping data to trips, 30-minute intervals, and shifts. The right part demonstrates the process of merging covariates table (SCEs, drivers, road geometry, and weather) back to the aggregated tables (trips, 30-minute intervals, and shifts tables). The specific details of the merging process and keys are shown below:

1. *SCEs*: the SCEs will be merged to the two aggregated tables by drivers and if the time of SCEs fall between the start and end time of the aggregated tables,

2. *Drivers*: the age of drivers are merged to the two aggregated tables using driver ID,

3. *Weather*: the weather variables will be merged to the original ping data by driver ID, latitude, longitude, date, and time. These weather variables will then be aggregated by taking the mean for each 30-minute interval and shift.

The resulting 30-minute intervals and shifts tables are demonstrated in Table 3.8 and Table 3.9. The predictor variables such as cumulative driving time, driver's age, weather and road geometry variables are truncated and not shown to fit in the page. Table 3.10 demonstrates the SCEs table, with time to events calculated as the time difference in hours between the time of the SCE and the starting time of the corresponding shift.

Table 3.8: *A sample of 30-minutes intervals data for hierarchical logistic and negative binomial regressions*

driver	start time	end time	interval time	distance (miles)
driver1	2015-10-23 08:09:26	2015-10-23 08:38:00	28	4.538
driver1	2015-10-23 09:04:24	2015-10-23 09:34:24	30	2.645
driver1	2015-10-23 09:34:24	2015-10-23 10:04:24	30	0.984
driver1	2015-10-23 10:04:24	2015-10-23 10:34:24	30	5.928
driver1	2015-10-23 10:34:24	2015-10-23 11:04:24	30	17.348

Table 3.9: *A sample of shifts data for hierarchical NHPP and JPLP*

driver	shift ID	# of SCE	time of SCEs	SCE type	start time	end time
driver1	1	1	2015-10-23 14:46:08	HB	2015-10-23 08:09:26	2015-10-23 18:37:56
driver1	2	1	2015-10-26 15:06:03	HB	2015-10-26 07:49:04	2015-10-26 15:06:58
driver1	3	0	NA	NA	2015-10-27 01:59:48	2015-10-27 07:58:56
driver1	4	2	2015-10-28 11:58:24; 2015-10-28 17:42:36	HB;HB	2015-10-28 08:05:08	2015-10-28 20:20:32
driver1	6	0	NA	NA	2015-10-30 09:27:12	2015-10-30 21:18:22

Aim 1

The first aim seeks to determine the association between the rate of crashes and the rate of SCEs at the level of drivers.

Table 3.10: *A demonstration of SCEs data for hierarchical non-homogeneous Poisson process*

driver	shift ID	start time	event time	shift length	time2event
driver1	1	2015-10-23 08:09:26	2015-10-23 14:46:08	10.467	6.600
driver1	2	2015-10-26 07:49:04	2015-10-26 15:06:03	7.283	7.267
driver1	4	2015-10-28 08:05:08	2015-10-28 11:58:24	12.250	3.883
driver1	4	2015-10-28 08:05:08	2015-10-28 17:42:36	12.250	9.617
driver1	7	2015-11-02 06:26:48	2015-11-02 07:13:56	13.667	0.783

Modeling strategy and its relation to the examined research questions

When making our modeling strategy, we considered two factors choosing an appropriate distribution for the outcome variable, and Bayesian vs. frequentist estimation. Our outcome (number of crashes, injuries, or fatalities) is a strictly non-negative integer. In the literature, Poisson regression or negative binomial models have been commonly applied for this type of outcome variable (e.g., see Table 2.1). Compared to Poisson regression models that by nature assumes the outcome distribution has equal mean and variance, negative binomial models can adjust for the variance independently from its mean, which allows for handling potential overdispersion or underdispersion issues in the data (Lord and Mannering, 2010). For the second factor, we adopted the Bayesian estimation approach since it provides more flexibility in specifying statistical models when compared to traditional maximum likelihood estimation methods (Dunson, 2001). Furthermore, in the case of rare-events, even relatively flat priors can increase the precision of parameter estimation.

Data

This aim uses the original ping data including 1,494,678,173 pings from 34,348 drivers, as demonstrated in Table 3.1. I excluded 2,520 drivers (7.4% of all the drivers) from our analysis if they met any of the following criteria: (a) driver inactivity, where the driver are required to have less than 100 active pings; (b) the unique identification code for the driver is not found in the demographics table; (c) the number of SCEs reported were identified as obvious outliers. Hereafter, the ping data used in aim 1 will correspond to only those generated by the remaining 31,828 drivers, with their characteristics summarized in Table 3.11

Outcome and predictor variables

The outcome variables are the number of crashes, injuries, and fatalities for each driver, respectively. The primary independent variable will be the number of SCEs per 10,000 miles. These SCEs will be further decomposed into the number of hard brakes, headways, and rolling stability per 10,000 miles in similar analysis. The covariates are age, mean

Table 3.11: *A summary of driver characteristics including their average age ± SD, number of drivers per gender, business unit and driver types (with their % in parentheses).*

Variable	Statistics
Age:	
Range	20 to 82 years
Mean age ± SD	44.48 ± 11.72
Gender: (%)	
Male	29,248 (91.9%)
Female	1,583 (5.0%)
Unknown	997 (3.1%)
Business unit: (%)	
Dedicated	16,152 (50.7%)
Final-mile	5,908 (18.6%)
Intermodal	9,768 (30.7%)
Driver type: (%)	
Local	13,381 (42.0%)
Regional	15,707 (49.3%)
Over-the-road	2,740 (8.6%)

speed, and gender. The offset variable is the total miles driven, which serves as a denominator in the negative binomial (NB) models.

Since the four SCEs are included in some models in the same time, I examined the correlation between different predictors. Note that the variables included both continuous (primary predictors, age and mean ping speed) and polytomous (gender, business unit, and driver type) variables. To account for the mixed variable types, the approach of Revelle et al. (2010,0) is adopted to compute the Pearson, polychoric and polyserial correlation coefficients for the pairwise evaluation of continuous, polytomous and mixed variables, respectively. Based on Figure 3.5, there are two observations to be highlighted. First, the Pearson correlation coefficient values between any two rates of SCEs were small (< 0.2). Second, none of the correlation coefficients exceeded 0.6 in magnitude. Accordingly, we concluded that the resulting statistical models will not have any serious multicollinearity issues (which will be examined in greater detail during the model assessment stage).

Bayesian negative binomial models

Let Y_i denote an outcome variable (i.e., the number of crashes, injuries or fatalities) over a distance of T_i miles for the ith driver. Each of the three outcomes were modeled in different Bayesian negative binomial models. I assume that Y_i has a negative binomial distribu-

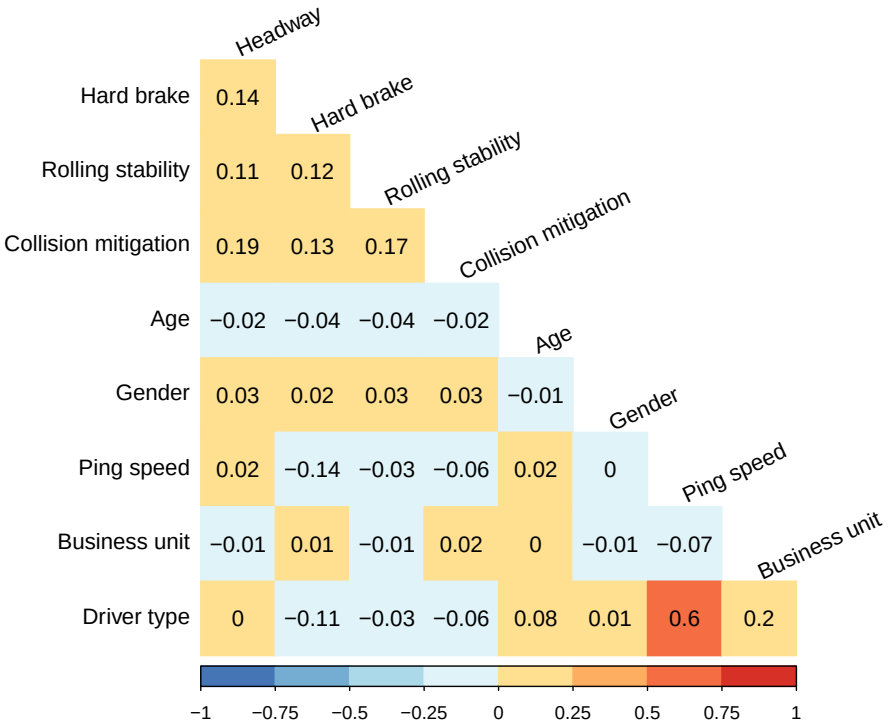


Figure 3.5: *A correlation plot of predictors and the covariates. The top four variables are the SCEs and the others are the covariates.*

tion with the mean parameter μ_i and a common auxiliary parameter ϕ. The corresponding probability mass function of Y_i is parameterized as

$$P(y_i|\mu,\phi) = \binom{y_i+\phi-1}{y_i}\left(\frac{\mu}{\mu+\phi}\right)^{y_i}\left(\frac{\phi}{\mu+\phi}\right)^{\phi}, \qquad y_i = 0,1,2,\ldots. \tag{3.1}$$

The mean and variance of Y_i are $E[Y_i] = \mu$ and $V(Y_i) = \mu + \frac{\mu^2}{\phi}$. The inverse of ϕ controls the overdispersion, which is scaled by μ^2. By assuming that the number of SCEs per 10,000 miles has a multiplicative effect on the logarithm of rate of crashes μ_i, I have the following log-linear Bayesian negative binomial regression:

$$\begin{aligned} Y_i &\sim \text{Negative Binomial}(T_i \times \mu_i,\ \phi) \\ \log \mu_i &= \alpha_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_J x_{iJ} + \theta_1 z_{i1} + \cdots + \theta_K z_{iK}, \end{aligned} \tag{3.2}$$

where the total miles driven T_i is considered as an offset to account for the mileage difference among drivers. α_0 is the intercept. $\beta_j,\ j = 1,2,\ldots,J$ is the coefficient of the j-th

primary predictor x_{ij}. $J = 1$ for pooled SCE rate models (x_{i1} represents the rate of any SCE) and one SCE rate at a time models (x_{i1} represents the rate of headway, hard brakes, rolling stability, or collision mitigation, with only one of them used at a time).

Note that the values of J and K depend on the application of the models. Specifically, $J = 4$ for four SCEs models ($x_{i1}, x_{i2}, x_{i3}, x_{i4}$ represent the rate of headway, hard brakes, rolling stability, and collision mitigation, respectively, with all four variables used in the same model at a time). $\theta_k, \ \ k = 1, 2, \ldots, K$ is the coefficient of the k-th covariate z_{ik}. $K = 4$ (age, mean speed, gender) for models stratified by business units and driver types, while $K = 8$ (age, mean speed, gender, business unit, and driver types) for non-stratified models. Gender, business units, and driver types were coded using dummy variables: (a) gender included female (reference group), male, and unknown; (b) business units included dedicated (reference group), intermodal, and final-mile; and (c) driver types included local (reference group), regional, and over-the-road. I took relatively noninformative priors for β_k and ϕ. Specifically, I assumed $\alpha_0, \ \ \beta_1, \ldots, \beta_J, \ \ \theta_1, \ldots, \theta_K \sim \text{Normal}(0, 10^2)$ and $\phi \sim \text{Exponential}(1)$.

Bayesian estimation and model validation

I applied the Hamiltonian Monte Carlo procedure to estimate the posterior distributions of all parameters. Compared to its two predecessor Markov chain Monte Carlo samplers, the Metropolis-Hastings algorithm and Gibbs sampling, Hamiltonian Monte Carlo is much more efficient in making valid proposal samples, and there is a well-developed statistical package to achieve this algorithm (McElreath, 2020). To make sure the Hamiltonian Monte Carlo converged to the true posterior distributions, I set 4,000 iterations for each of the four chains, with the first 2,000 being warm-up iterations. The Markov chains were considered as converged when the Gelman-Rubin diagnostic $\hat{R}$ was less than 1.1 for each variable (Gelman et al., 1992).

Since four different SCEs and/or multiple covariates were included in some of the mod-

els at the same time, it is important to check for the presence of multicollinearity. Here, I attempt to investigate whether a linear dependence exist among three or more of our variables through computing the variance inflation factors (VIFs). If the regressors are uncorrelated, VIF obtains its minimum value of 1. In statistical practice, a VIF less than 4 requires no additional investigation of linear dependence among the regressors and values greater than 10 indicate serious multicolinearity requiring model corrections.

Pareto smoothed importance-sampling leave-one-out (PSIS-LOO) cross-validation is used to check the goodness-of-fit of and compare different models (Vehtari et al., 2015,0). Instead of exact cross-validation that refits the model with different subsamples, the PSIS-LOO uses fast, efficient, and stable importance sampling weights to approximate leave-one-out cross-validation (Gelfand et al., 1992; Gilks et al., 1996). It estimates the expected log predicted density (ELPD), estimate number of parameters, and the LOO Information Criterion (LOOIC) for a new data set. Compared with other statistics such as Widely Applicable Information Criterion, Deviance information criterion, and other variants (Spiegelhalter et al., 2002; Watanabe, 2010), PSIS-LOO is both fast and stable in computing. Apart from PSIS-LOO, I also used posterior predictive checks to examine the prediction accuracy (Gelman et al., 2013, Chapter 6). The interpretation of these goodness-of-fit and model comparison statistics will be explained in Section 4.

Aim 2

The purpose of aim 2 is to develop two scalable hierarchical models (logistic and negative binomial regression) for the SCEs and identify potential risk factors. Both hierarchical models will account for both driver-level and trip-level variables. This aim uses the merged 30-minute interval data generated by the 496 sample regional drivers, as shown in Table 3.8.

Modeling strategy and relevance to the aim

Traditional statistical models assume that observations are independent from each other given their predictor variables. However, real data are almost never independent given the

predictor variables. In the example of truck driver's safety events, a truck drivers may exhibit similar driving patterns in multiple trips. Therefore, traffic accidents are naturally nested within drivers. Traditional statistical models that assume independence between observations are not appropriate in this case since objects tend to be similar within a group. Hierarchical models, also known as multilevel model, random-effects model or mixed model, have been developed to allow for the nested nature of data. Instead of assuming independence given predictor variables, hierarchical models assume conditional independence. Hierarchical models are advocated to be valuable since they can produce better prediction and more robust results than traditional models (Han et al., 2018; Pantangi et al., 2019).

Outcomes and predictors

The outcome variable of the hierarchical logistic regression is a binary variable of whether any SCEs occurred in the 30-minute interval, while the outcome variable of the hierarchical negative binomial regression is the count variable of the number of SCEs in the 30-minute interval.

The predictors in both models include driver demographics (age, gender, and race), mean weather (visibility, precipitation intensity and probability at trip level), and interval specific variables (mean and standard deviation of speed). Table 3.12 presents the summary statistics of driver characteristics and the aggregated 30-minute interval. Most of the driving tasks were performed in relatively high speed (mean speed is 43.01 miles per hour), good weather conditions, and working hours. The average age of the sample drivers was 45.83 (standard deviation: 12.03), with 36 (7.2%) female drivers. There were 247 (49.7%) white, 206 (41.4%) black, and 44 other-race drivers (8.9%).

Hierarchical logistic and negative binomial regression

Here we model the outcome variable SCEs using two set of hierarchical models: logistic and negative binomial (NB) regression models with driver-level intercepts and slopes. In the hierarchical logistic regression model, we categorized the number of SCEs during the

Table 3.12: *A summary of predictor variables and driver characteristics*

Category	Variables	Overall
Driver	Age (mean (SD))	45.83 (12.03)
	Gender = Female	36 (7.2%)
	Race	
	White	247 (49.7%)
	Black	206 (41.4%)
	Other	44 (8.9%)
Weather	Precipitation intensity (mean (SD))	0.00 (0.02)
	Precipitation probability (mean (SD))	0.06 (0.21)
	Wind speed (mean (SD))	3.74 (3.19)
	Visibility (mean (SD))	8.74 (2.23)
Traffic proxy	Weekend = Yes	150,409 (14.8%)
	Holiday = Yes	15,354 (1.5%)
	Hour of the day	
	21 p.m. - 5 a.m.	123,599 (12.1%)
	6 a.m. - 10 a.m.	273,382 (26.8%)
	11 a.m. - 14 p.m.	294,352 (28.9%)
	15 p.m. - 20 p.m.	328,149 (32.2%)
Interval	Cumulative driving hours (mean (SD))	4.47 (2.76)
	Speed mean (mean (SD))	43.01 (19.77)
	Speed standard deviation (mean (SD))	11.21 (10.37)
	Interval time in minutes (mean (SD))	27.40 (6.70)

i-th 30-minute interval into a binary variable Y_i with the value of either 0 or 1, where 0 indicated that no SCE occurred during that trip while 1 indicated that at least 1 SCEs occurred during the trip. The hierarchical logistic regression model is parameterized as:

$$\begin{aligned} Y_i &\sim \text{Bernoulli}(p_i) \\ \log\frac{p_i}{1-p_i} &= \beta_{0,d(i)} + \beta_{1,d(i)} \cdot \text{CT}_i + \beta_2 x_2 + \cdots + \beta_k x_k \\ \beta_{0,d(i)} &\sim N(\mu_0, \sigma_0^2) \\ \beta_{1,d(i)} &\sim N(\mu_1, \sigma_1^2), \end{aligned} \tag{3.3}$$

where $d(i)$ is the index of the driver for the i-th interval. $\beta_{0,d(i)}$ is the random intercept for driver $d(i)$; $\beta_{1,d(i)}$ is the random slope for the cumulative driving time (CTi) in the shift (the sum of driving time for all previous intervals within that shift) for driver $d(i)$. These random intercepts and random slopes are assumed to have a hyper-distribution with hyperparameters $\mu_0, \sigma_0, \mu_1, \sigma_1$. $x_2, \ldots, x_k$ are other fixed-effect variables including driver demographics (age, gender, and race), weather (visibility, precipitation intensity and prob-

ability), interval specific variables (mean and standard deviation of speed), and $\beta_2, \ldots, \beta_k$ are the associated parameters.

Although logistic regression is more robust to outliers of the outcome variable in each 30-interval, it does not fully use the information in the outcome variable since only a binary variable is used. Here we present a hierarchical NB model, with the number of SCEs $Y_i^\star$ within the i-th interval as the outcome variable. The hierarchical NB regression model is parameterized as:

$$\begin{aligned} Y_i^\star &\sim \text{Negative Binomial}(T_i \times \mu_i,\ \mu_i + \frac{\mu_i^2}{\theta}) \\ \log \mu_i &= \beta^\star_{0,d(i)} + \beta^\star_{1,d(i)} \cdot \text{CT}_i + \beta_2^\star x_2 + \cdots + \beta_k^\star x_k \\ \beta^\star_{0,d(i)} &\sim N(\mu_0^\star, \sigma_0^{\star 2}) \\ \beta^\star_{1,d(i)} &\sim N(\mu_1^\star, \sigma_1^{\star 2}), \end{aligned} \tag{3.4}$$

where T_i is the length of the i-th interval, μ_i is the expected number of SCEs per hour, θ is a fixed over-dispersion parameter. Since there is no good solution to estimate the θ parameter here, it was set as a fixed value estimated from a Poisson regression using maximum likelihood estimation. Other parameters are similar and explained in the previous hierarchical logistic regression model, and we put a $\star$ on the parameter to note the difference between the parameters of the two models.

To compare hierarchical models with non-hierarchical models, we also estimated logistic and NB regression models without any drive-level random effects. After comparing the performance of the four models in predicting SCEs, we choose the best model to predict different SCEs (hard brake, headway, collision mitigation, and rolling stability, respectively) among the sample drivers.

Aim 3

Aim 3 seeks to investigate the pattern of SCEs using recurrent-event models. A non-homogeneous Poisson process (NHPP) using a PLP intensity function and an innovative jump power law

process (JPLP) are proposed. This aim 3 uses the merged SCEs data within shifts by the 496 sample regional drivers, as shown in Table 3.9.

Modeling strategy and relevance to the aim

The hierarchical negative binomial model in aim 2 assumes that the intensity of SCEs in an interval is a constant, which may not hold in transportation practice. Figure 3.6 shows a sample of SCEs distributions in different shifts. Each arrow represents a shift while each red cross shows a SCE. The figure shows that SCEs are not uniformly distributed within a shift, but seem to cluster in the early stages of shifts. These recurrent events data fit into the analysis framework of point process and reliability models (Rigdon and Basu, 2000).

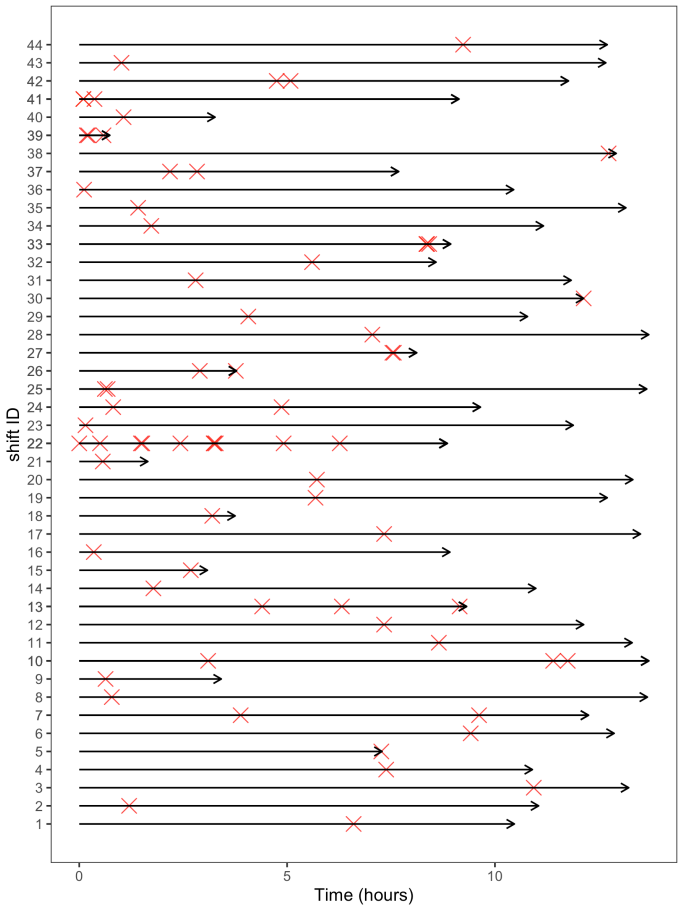


Figure 3.6: *An arrow plot of time to SCEs in each shift*

Therefore, a hierarchical NHPP with a PLP intensity function is proposed, which will answer whether SCEs occurred more frequently at early stages, towards the end, or does not show significant patterns within shifts. Afterwards, the rest time within shifts is accounted for by adding one more parameter κ in the hierarchical NHPP (Figure 3.7). The intensity function of the proposed jump-point PLP will be recovered for a certain percent κ every time the driver took a short break (less than eight hours) within shifts. This new reliability model (jump-point PLP, JPLP) will be a balance between a NHPP where the intensity function is not

influenced by within-shift rests ("as bad as old") and a renewal process where the intensity function is fully recovered by within-shift rests ("as good as new").

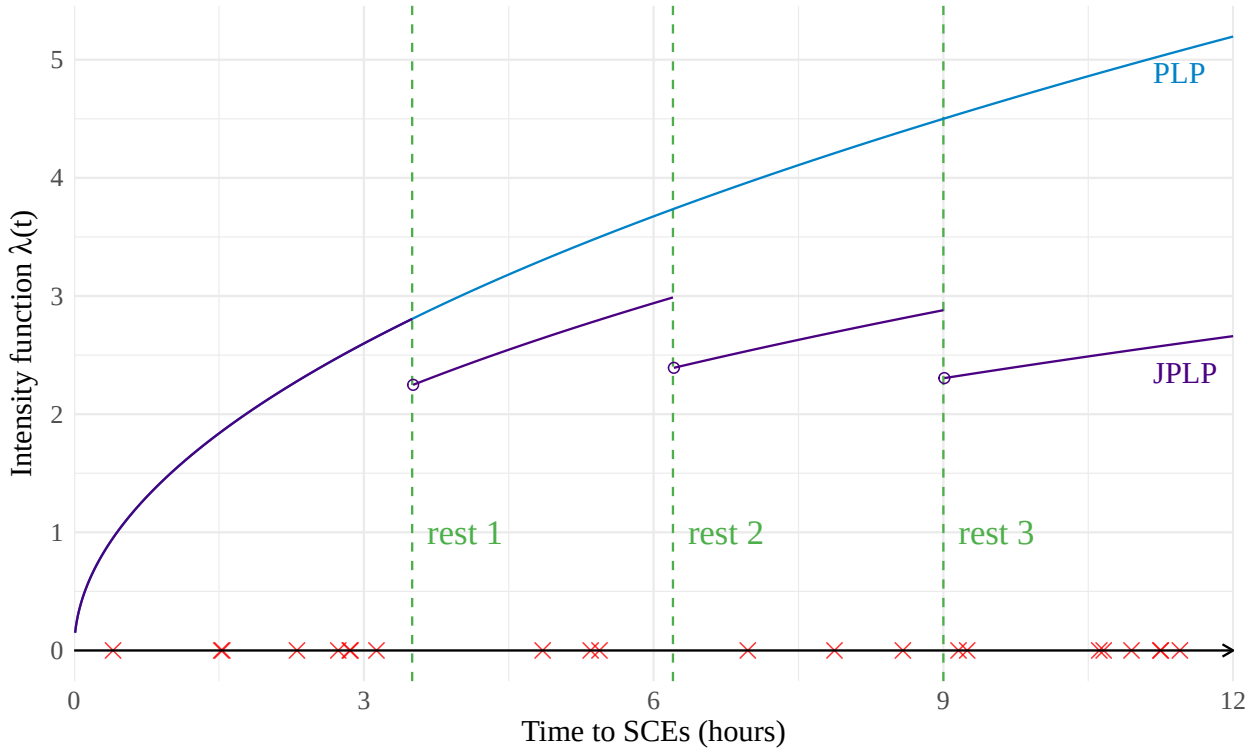


Figure 3.7: *Simulated intensity function of PLP and JPLP. The x-axis shows time in hours since start and y-axis shows the intensity of SCEs. The red crosses mark the time to SCEs and the green vertical lines indicates the time of the rests. Parameter values for simulation: shape parameters* $\beta = 1.2$*, rate parameter* $\theta = 2$*, jump parameter* $\kappa = 0.8$*.*

Outcomes and predictors

The outcomes in this aim are the time to the SCEs since the start of shifts. The predictors include driver demographics (age, gender, and race), weather (mean visibility, precipitation intensity and probability at shift level), and shift specific variables (mean and standard deviation of speed). The driver-, shift-, trip-, and SCE-level notations are:

- Driver $d : 1, 2, \ldots, D$,
- Shift $s, 1, 2, \ldots, S_d$,
- Trip $r : 1, 2, \ldots, R_{d,s}$,
- SCE $i : 1, 2, \ldots, I_{d,s}$.

The data used in this aim are the same with those in Aim 2, but aggregated on trip- and

shift-level for PLP and JPLP estimation. The notations for data in this aim are:

- $t_{d,s,i}$: time to the i-th SCE for driver d measured from the beginning of the s-shift,
- $n_{d,s,r}$: the number of SCEs for trip r within shift s for driver d,
- $a_{d,s,r}$: the end time of trip r within shift s for driver d.

Non-homogeneous Poisson Process and Power Law Process (PLP)

We assume the time to SCEs $t_{d,s,i}$ follows a non-homogeneous Poisson process, whose intensity function $\lambda(t)$ is non-constant. The intensity function is assumed to have the following function form:

$$\lambda_{\text{PLP}}(t) = \beta\theta^{-\beta}t^{\beta-1}, \tag{3.5}$$

where the shape parameter β indicates reliability improvement ($\beta < 1$), constant ($\beta = 1$), or deterioration ($\beta > 1$), and the scale parameter θ determines the rate of events. Here we assume the intensity function of a PLP because it has a flexible functional form, relatively simple statistical inference, and is a well-established model (Rigdon and Basu, 1989, 2000).

Bayesian Hierarchical Power Law Process (PLP)

The Bayesian hierarchical power law process is parameterized as:

$$\begin{aligned}
t_{d,s,1}, t_{d,s,2}, \cdots, t_{d,s,n_{d,s}} &\sim \text{PLP}(\beta, \theta_{d,s}, \tau_{d,s}) \\
\beta &\sim \text{Gamma}(1,1) \\
\log\theta_{d,s} &= \gamma_{0d} + \gamma_1 x_{d,s,1} + \gamma_2 x_{d,s,2} + \cdots + \gamma_k x_{d,s,k} \\
\gamma_{01}, \gamma_{02}, \cdots, \gamma_{0D} &\sim \text{i.i.d. } N(\mu_0, \sigma_0^2) \\
\gamma_1, \gamma_2, \cdots, \gamma_k &\sim \text{i.i.d. } N(0, 10^2) \\
\mu_0 &\sim N(0, 5^2) \\
\sigma_0 &\sim \text{Gamma}(1,1),
\end{aligned} \tag{3.6}$$

where $t_{d,s,i}$ is the time to the i-th event for driver d in shift s, $\tau_{d,s} = a_{d,s,R_{d,s}}$ is the length of time of shift s (truncation time) for driver d, and $n_{d,s} = \sum_{r=1}^{n_{d,s}}$ is the number of SCEs in shift s for driver d. The likelihood function of event times generated from a PLP for

driver d in shift s is given in Rigdon and Basu (2000, Section 2.3.2, Page 60):

$$L_{d,s}(\beta, \gamma_{0d}, \ |\mathbf{X}_d, \mathbf{W}_s) = \Big(\prod_{i=1}^{n_{d,s}} \lambda_{\text{PLP}}(t_{d,s,i}) \Big) \exp(- \int_0^{\tau_{d,s}} \lambda(u) du)$$

$$= \begin{cases} \exp\Big(- (\tau_{d,s}/\theta_{d,s})^{\beta} \Big), & \text{if } n_{d,s} = 0, \\ \Big(\prod_{i=1}^{n_{d,s}} \beta \theta_{d,s}^{-\beta} t_{d,s,i}^{\beta-1} \Big) \exp\Big(- (\tau_{d,s}/\theta_{d,s})^{\beta} \Big), & \text{if } n_{d,s} > 0, \end{cases} \tag{3.7}$$

where $\mathbf{X}_d$ indicates driver specific variables (e.g. driver age and gender), $\mathbf{W}_s$ represents shift specific variables (e.g. precipitation and traffic), and $\theta_{d,s}$ is the function of parameters $\gamma_{0d}, \gamma_1, \gamma_2, \dots, \gamma_k$ and variables $x_{d,s,1}, x_{d,s,2}, \dots, x_{d,s,k}$ given in the third line of Equation 3.6. The full likelihood function for all drivers are:

$$L = \prod_{d=1}^{D} \prod_{s=1}^{S_d} L_{d,s}(\beta, \gamma_{0d}, \ |\mathbf{X}_d, \mathbf{W}_s) \tag{3.8}$$

where $L_{d,s}(\beta, \gamma_{0d}, \ |\mathbf{X}_d, \mathbf{W}_s)$ is given in Equation 3.7.

Bayesian Hierarchical Jump Power Law Process (JPLP)

Since the Bayesian hierarchical PLP in Subsection 3.6.4 does not account for the rests $(r : 1, 2, \dots, R_{d,s})$ within shifts and associated potential reliability improvement. In this subsection, we proposes a Bayesian hierarchical JPLP, with the following piecewise intensity function:

$$\lambda_{\text{JPLP}}(t|d,s,r,\beta,\gamma_{0,d},\ ,\mathbf{X}_d,\mathbf{W}_s) = \begin{cases} \kappa^0\lambda(t|\beta,\gamma_{0,d},\ ,\mathbf{X}_d,\mathbf{W}_s), & 0 < t \le a_{d,s,1}, \\ \kappa^1\lambda(t|\beta,\gamma_{0,d},\ ,\mathbf{X}_d,\mathbf{W}_s), & a_{d,s,1} < t \le a_{d,s,2}, \\ \ldots & \ldots \\ \kappa^{R-1}\lambda(t|\beta,\gamma_{0,d},\ ,\mathbf{X}_d,\mathbf{W}_s), & a_{d,s,R-1} < t \le a_{d,s,R}, \end{cases}$$
$$= \kappa^{r-1}\lambda(t|d,s,r,\kappa,\beta,\gamma_{0,d},\ ,\mathbf{X}_d,\mathbf{W}_s),\ a_{d,s,r-1} < t \le a_{d,s,r}, \tag{3.9}$$

where the introduced parameter κ is the percent of intensity function recovery once the driver takes a break, and $a_{d,s,r}$ is the end time of trip r within shift s for driver d. By definition, the end time of the 0-th trip $a_{d,s,0} = 0$, and the end time of the last trip for the d-driver within the s-th shift $a_{d,s,R_{d,s}}$ equals the shift end time $\tau_{d,s}$. We assume that this κ is constant across drivers and shifts.

The Bayesian hierarchical JPLP model is parameterized as

$$\begin{aligned} t_{d,s,1}, t_{d,s,2}, \cdots, t_{d,s,n_{d,s}} &\sim \text{JPLP}(\beta, \theta_{d,s}, \tau_{d,s}, \kappa) \\ \beta &\sim \text{Gamma}(1,1) \\ \log\theta_{d,s} &= \gamma_{0d} + \gamma_1 x_{d,s,1} + \gamma_2 x_{d,s,2} + \cdots + \gamma_k x_{d,s,k} \\ \kappa &\sim \text{Uniform}(0,1) \\ \gamma_{01}, \gamma_{02}, \cdots, \gamma_{0D} &\sim \text{i.i.d. } N(\mu_0, \sigma_0^2) \\ \gamma_1, \gamma_2, \cdots, \gamma_k &\sim \text{i.i.d. } N(0, 10^2) \\ \mu_0 &\sim N(0, 5^2) \\ \sigma_0 &\sim \text{Gamma}(1,1), \end{aligned} \tag{3.10}$$

The notations are identical with those in Equation 3.6 except for the extra κ parameter. Similarly, the likelihood function of event times generated from a JPLP for driver d on

shift s is

$$L^*_{d,s}(\kappa, \beta, \gamma_{0d},\ |\mathbf{X}_d, \mathbf{W}_s) = \begin{cases} \exp\Big(-\int_0^{\tau_{d,s}} \lambda_{\text{JPLP}}(u)du\Big), & \text{if } n_{d,s} = 0, \\ \Big(\prod_{i=1}^{n_{d,s}} \lambda_{\text{JPLP}}(t_{d,s,i})\Big) \exp\Big(-\int_0^{\tau_{d,s}} \lambda_{\text{JPLP}}(u)du\Big), & \text{if } n_{d,s} > 0, \end{cases} \tag{3.11}$$

where the piecewise intensity function $\lambda_{\text{JPLP}}(t_{d,s,i})$ is given in Equation 3.9.

However, since the intensity function depends on the trip r for the same driver d and shift s, it is hard to write out specific form of Equation 3.11. Instead, we can rewrite the likelihood function at trip level, where the intensity function λ_{JPLP} is fixed for driver d on shift s and trip r:

$$L^*_{d,s,r}(\kappa, \beta, \gamma_{0d},\ |\mathbf{X}_d, \mathbf{W}_r) = \begin{cases} \exp\Big(-\int_{a_{d,s,r-1}}^{a_{d,s,r}} \lambda_{\text{JPLP}}(u)du\Big), & \text{if } n_{d,s,r} = 0, \\ \Big(\prod_{i=1}^{n_{d,s,r}} \lambda_{\text{JPLP}}(t_{d,s,r,i})\Big) \exp\Big(-\int_{a_{d,s,r-1}}^{a_{d,s,r}} \lambda_{\text{JPLP}}(u)du\Big), & \text{if } n_{d,s,r} > 0, \end{cases} \tag{3.12}$$

where $t_{d,s,r,i}$ is the time to the i-th SCE for driver d on shift s and trip r measured from the beginning of the shift, $n_{d,s,r}$ is the number of SCEs for driver d on shift s and trip r. Compared to the PLP likelihood function given in Equation 3.8 where $\mathbf{W}_s$ are assumed to be a constant during an entire shift, the rewritten likelihood function for JPLP in Equation 3.12 assumes external covariates $\mathbf{W}_r$ vary between different trips in a shift. In this way, JPLP can account for the variability between different trips within a shift.

Therefore, the overall likelihood function for drivers $d = 1, 2, \ldots, D$, their corresponding

shifts $s = 1, 2, \ldots, S_d$, and trips $r = 1, 2, \ldots, R_{d,s}$ is:

$$L^* = \prod_{d=1}^{D} \prod_{s=1}^{S_d} \prod_{r=1}^{R_{d,s}} L^*_{d,s,r}, \tag{3.13}$$

where $L^*_{d,s,r}$ is a likelihood function given in Equation 3.12, in which the intensity function λ_{JPLP} has a fixed functional form provided in the last line of Equation 3.9 for a certain driver d in a given shift s and trip r.

Simulation setting

To validate the R and Stan code I wrote for the PLP and proposed JPLP, I conducted a simulation study to evaluate the performance of our proposed JPLP. I performed 1,000 simulations to each of following three scenarios with different number of drivers ($D = 10, 25, 50, 75, 100$):

1. Data generated from a PLP and estimated assuming a PLP (PLP),
2. Data generated from a JPLP, but estimated assuming a PLP (PLP $\leftarrow$ JPLP),
3. Data generated from a JPLP and estimated assuming a JPLP (JPLP).

The scenario "data generated from a PLP, but estimated assuming a JPLP" is not considered here since it is not theoretically possible: if the data is generated from a PLP, then there are no breaks within shift and it is impossible to estimate the data assuming a JPLP.

Specifically, for each driver, the number of shifts is simulated from a Poisson distribution with the mean parameter of 10. We assume three predictor variables x_1, x_2, x_3 for θ ($k =$

3) and shift time $\tau_{d,s}$ are generated from the following process:

$$\begin{aligned} x_1 &\sim \text{Normal}(1, 1^2) \\ x_2 &\sim \text{Gamma}(1, 1) \\ x_3 &\sim \text{Poisson}(2) \\ \tau_{d,s} &\sim \text{Normal}(10, 1.3^2) \end{aligned} \tag{3.14}$$

The parameters and hyperparameters are assigned the following values or generated from the following process:

$$\begin{aligned} \mu_0 = 0.2,\ \sigma_0 = 0.5, \\ \gamma_{01}, \gamma_{02}, \cdots, \gamma_{0D} \sim \text{i.i.d. } N(\mu_0, \sigma_0^2) \\ \gamma_1 = 1,\ \gamma_2 = 0.3,\ \gamma_3 = 0.2 \\ \theta_{d,s} = \exp(\gamma_{0d} + \gamma_1 x_1 + \gamma_2 x_2 + \gamma_3 x_3) \\ \beta = 1.2,\ \kappa = 0.8. \end{aligned} \tag{3.15}$$

After the predictor variables, shift time, and parameters are generated, the time to events $t_{d,s,1}$, $t_{d,s,2}$, ⋯, $t_{d,s,n_{d,s}}$ and $t^*_{d,s,1}, t^*_{d,s,2}, \cdots, t^*_{d,s,n_{d,s}}$ are generated from PLP and JPLP:

$$\begin{aligned} t_{d,s,1}, t_{d,s,2}, \cdots, t_{d,s,n_{d,s}} &\sim \text{PLP}(\beta, \theta_{d,s}, \tau_{d,s}) \\ t^*_{d,s,1}, t^*_{d,s,2}, \cdots, t^*_{d,s,n_{d,s}} &\sim \text{JPLP}(\beta, \theta_{d,s}, \tau_{d,s}, \kappa) \end{aligned} \tag{3.16}$$

The parameters are then inferred using the likelihood functions given in Equation 3.8 and 3.13 with probabilistic programming language Stan in R (Carpenter et al., 2017; Stan Development Team, 2018a), which uses efficient Hamiltonian Monte Carlo to sample from the posterior distributions. For each simulation, one chain is applied, with 2,000 warmup and 2,000 post-warmup iterations drawn from the posterior distributions.

Statistical software and cloud platform

All data reduction, cleaning, visualization, and statistical analysis are done on the RStudio Server on the Ohio Supercomputer Center (OSC), which utilizes the statistical computing environment R 3.5.1 (R Core Team, 2018). The OSC provides high performance computing resources and expertise to academic researchers (Center, 1987).

The data processing is based on the high-performance **R** package `data.table` and `dplyr` (Dowle and Srinivasan, 2019; Wickham et al., 2019). All the visualization is performed in the **R** package `ggplot2` (Wickham, 2016). The Bayesian negative binomial models are performed using the `rstanarm` package (Gabry and Goodrich, 2016), while the hierarchical logistic and negative binomial regression models are performed using the `lme4` package (Bates et al., 2015). The hierarchical PLP and JPLP were constructed using self-defined likelihood functions in `rstan` package in **R** (Stan Development Team, 2018b; R Core Team, 2018).

CHAPTER 4 RESULTS

Aim 1

The association of SCEs with crashes, injuries, and fatalities

Table 4.1 presents the Bayesian negative binomial models' results explaining the variation in number of crashes, injuries, and fatalities separately using either a pooled SCE predictor (i.e., four SCEs as one variable) or four SCE predictors (i.e., the four SCEs represented using four variables). The IRR of the "all SCEs" predictor in column 2 suggests that a unit increase in the number of any type of SCEs per 10,000 miles was associated with an 8.4% (95% CI: 8.0-8.8%) increase in the rate of crashes. The four SCEs predictors in column 3 suggest: (a) the incidence rate ratios (IRRs) of all four predictors and their 95% CI are greater than 1, indicating that an increase in any of the four SCEs is associated with an increase in the number of crashes; (b) a unit increase in the number of instances of rolling stability system initiation was associated with the largest, 50.4% (95% CI: 41.4-60.0%), in the number of crashes per mile; and (c) when holding other SCEs constant, a unit increase in either the initiation of the rolling stability or collision mitigation systems has larger effects when compared to increases in hard brakes or headyway alerts.

Compared with the models for crashes, the results for injuries and fatalities (columns 4-7) tend to be less conclusive since the number of recorded injuries and fatalities are much smaller. In the injuries-pooled model (column 4), a unit increase in the number of any type of SCEs per 10,000 miles was associated with 8.7% (95% CI: 4.8%-13.6%) increase in the number of injuries per mile. When stratified into four different types of SCEs, all 95% CIs of incidence rate ratios included one, which indicated weak (statistically insignificant) evidence for modeling injuries or fatalities, although the posterior means were positive. In the two models using the number of fatalities as the outcome variable (columns 6 and 7), all 95% CIs of incidence rate ratios included one and the CIs were very wide, which suggested that the observed sample size was not sufficient to yield statistically significant

Table 4.1: *Bayesian negative binomial regressions with the rate of SCEs predicting crashes, injuries, and fatalities*

Variables	Crashes: pooled	Crashes: four SCEs	Injuries: pooled	Injuries: four SCEs	Fatalities: pooled	Fatalities: four SCEs
Intercept	0.054 (0.047, 0.062)	0.048 (0.042, 0.054)	0.013 (0.002, 0.070)	0.012 (0.002, 0.064)	0.008 (0.000, 1.855)	0.011 (0.000, 5.179)
All SCEs	1.084 (1.080, 1.088)		1.087 (1.048, 1.136)		0.973 (0.791, 1.149)	
Headways		1.033 (1.026, 1.040)		1.061 (0.961, 1.181)		0.955 (0.592, 1.478)
Hard brakes		1.081 (1.075, 1.087)		1.080 (0.995, 1.177)		0.957 (0.652, 1.387)
Rolling stability		1.504 (1.414, 1.600)		1.773 (0.684, 5.439)		1.631 (0.043, 102.8)
Collision mitigation		1.222 (1.198, 1.245)		1.174 (0.987, 1.535)		0.866 (0.200, 3.632)
Age	0.992 (0.990, 0.993)	0.992 (0.991, 0.993)	0.987 (0.970, 1.004)	0.986 (0.969, 1.004)	0.966 (0.912, 1.020)	0.965 (0.906, 1.030)
Mean speed	0.979 (0.976, 0.982)	0.982 (0.979, 0.985)	0.967 (0.929, 1.007)	0.970 (0.931, 1.009)	0.915 (0.797, 1.049)	0.910 (0.778, 1.050)
Gender: male	0.817 (0.756, 0.886)	0.808 (0.754, 0.867)	0.825 (0.301, 2.149)	0.800 (0.298, 2.176)	1.770 (0.074, 54.444)	1.953 (0.062, 80.045)
Gender: unknown	0.975 (0.785, 1.199)	0.954 (0.777, 1.149)	1.022 (0.094, 8.499)	0.993 (0.092, 9.338)	0.093 (0, 76.2)	0.093 (0, 115.6)
Business unit: Inter-modal	0.698 (0.670, 0.727)	0.717 (0.690, 0.745)	0.459 (0.265, 0.788)	0.467 (0.280, 0.789)	0.354 (0.068, 1.573)	0.341 (0.044, 2.057)
Business unit: Final-mile	0.907 (0.861, 0.954)	0.897 (0.852, 0.943)	0.710 (0.352, 1.420)	0.675 (0.330, 1.321)	1.576 (0.209, 10.438)	1.536 (0.140, 13.475)
Type: Over-the-road	1.071 (0.994, 1.151)	1.094 (1.022, 1.174)	0.785 (0.321, 1.942)	0.801 (0.306, 1.955)	0.410 (0.022, 5.402)	0.388 (0.014, 6.205)
Type: Regional	1.003 (0.957, 1.045)	1.012 (0.969, 1.057)	0.472 (0.265, 0.821)	0.463 (0.263, 0.820)	0.389 (0.064, 1.970)	0.379 (0.050, 2.214)
Fit statistics:						
sample size	31828	31828	31828	31828	31828	31828
elpd_loo	-39985.2 (236.5)	-39770.2 (233.5)	-1134.5 (80.8)	-1137.3 (81.1)	-182.4 (37.9)	-182.4 (37.9)
p_loo	18.1 (1.1)	30 (2.4)	13.9 (3.6)	16.4 (4)	11.3 (3.2)	11.3 (3.2)
looic	79970.4 (472.9)	79540.5 (467.1)	2269.1 (161.5)	2274.6 (162.1)	364.7 (75.7)	364.7 (75.7)

Notes: The SCEs were measured as the number of events per 10,000 miles driven.
Incidence rate ratios and their associated 95% credible intervals are reported for all variables (predictors and covariates).
For the fit statistics, (.) indicates the standard error of the computed statistic.

results.

Consistent association between crashes and the four different SCEs

Table 4.2 shows the estimates of posterior IRR and their CIs in the Bayesian negative binomial models for all the included drivers. Similar conclusions can be made to the insights gained from examining the pooled and four SCE models. Specifically, the coefficients of

each of the main predictors were larger than one (with the associated 95% credible interval excluding one), providing statistically strong evidence that the rates of the individual SCEs were positively associated with the rates of crashes. Furthermore, the rolling stability coefficient is the largest, followed by the crash mitigation coefficient, confirming that an

Table 4.2: *Bayesian negative binomial regressions predicting crashes with different combination of SCEs*

Variables	Pooled model	Four SCEs	Headways	Hard brakes	Rolling stability	Collision mitigation
Intercept	0.054 (0.047, 0.062)	0.048 (0.042, 0.054)	0.090 (0.079, 0.103)	0.057 (0.050, 0.066)	0.082 (0.072, 0.093)	0.073 (0.064, 0.083)
All SCEs	1.084 (1.080, 1.088)					
Headways		1.033 (1.026, 1.040)	1.077 (1.069, 1.085)			
Hard brakes		1.081 (1.075, 1.087)		1.109 (1.102, 1.116)		
Rolling stability		1.504 (1.414, 1.600)			2.147 (2.015, 2.295)	
Collision mitigation		1.222 (1.198, 1.245)				1.343 (1.316, 1.369)
Age	0.992 (0.990, 0.993)	0.992 (0.991, 0.993)	0.989 (0.988, 0.990)	0.991 (0.989, 0.992)	0.989 (0.988, 0.991)	0.990 (0.988, 0.991)
Mean speed	0.979 (0.976, 0.982)	0.982 (0.979, 0.985)	0.971 (0.968, 0.973)	0.980 (0.977, 0.983)	0.973 (0.970, 0.976)	0.975 (0.973, 0.978)
Gender: male	0.817 (0.756, 0.886)	0.808 (0.754, 0.867)	0.848 (0.785, 0.919)	0.823 (0.762, 0.887)	0.845 (0.787, 0.909)	0.826 (0.770, 0.891)
Gender: unknown	0.975 (0.785, 1.199)	0.954 (0.777, 1.149)	1.097 (0.896, 1.347)	1.096 (0.884, 1.349)	1.018 (0.842, 1.239)	1.058 (0.870, 1.299)
Business unit: Inter-modal	0.698 (0.670, 0.727)	0.717 (0.690, 0.745)	0.706 (0.679, 0.735)	0.701 (0.672, 0.730)	0.735 (0.706, 0.765)	0.729 (0.700, 0.758)
Business unit: Final-mile	0.907 (0.861, 0.954)	0.897 (0.852, 0.943)	0.925 (0.882, 0.971)	0.904 (0.865, 0.948)	0.922 (0.880, 0.967)	0.901 (0.859, 0.942)
Type: Over-the-road	1.071 (0.994, 1.151)	1.094 (1.022, 1.174)	1.053 (0.981, 1.131)	1.064 (0.994, 1.140)	1.067 (0.990, 1.144)	1.106 (1.030, 1.182)
Type: Regional	1.003 (0.957, 1.045)	1.012 (0.969, 1.057)	0.971 (0.928, 1.015)	0.994 (0.950, 1.037)	0.973 (0.932, 1.016)	0.984 (0.943, 1.028)
Fit statistics:						
sample size	31828	31828	31828	31828	31828	31828
elpd_loo	-39985.2 (236.5)	-39770.2 (233.5)	-40792.7 (238.9)	-40315.5 (237.2)	-40710.1 (237.8)	-40503.2 (239.4)
p_loo	18.1 (1.1)	30 (2.4)	19.8 (1.9)	18.2 (1.2)	15.9 (0.8)	16.1 (1)
looic	79970.4 (472.9)	79540.5 (467.1)	81585.4 (477.8)	80631 (474.5)	81420.1 (475.7)	81006.5 (478.7)

Notes: The SCEs were measured as the number of events per 10,000 miles driven.
Incidence rate ratios and their associated 95% credible intervals are reported for all variables (predictors and covariates).
For the fit statistics, (.) indicates the standard error of the computed statistic.

increase in more aggressive interventions results in more crashes.

The influence of business units and driver types on the association between crashes and SCEs

Table 4.3 shows the four SCEs models stratified by different business units and types. The posterior IRRs and CIs of four SCEs are consistent with those in Table 4.1. All four type of SCEs were positively associated with the number of crashes per mile. None of the CIs included one except for headways in the dedicated and over-the-road unit. These stratified results indicate strong evidence that SCEs were positively associated with crashes in different business units and driver types. Among the four types of SCEs, rolling stability had the highest IRRs, followed by collision mitigation, hard brake, and headway.

Table 4.3: *Bayesian negative binomial regressions with SCEs predicting crashes, stratified by business units and driver types*

Variables	Dedicated			Inter-modal		Final-mile	
	Local	OTR	Regional	Local	Regional	OTR	Regional
Intercept	0.055 (0.040, 0.076)	0.015 (0.008, 0.027)	0.062 (0.046, 0.084)	0.026 (0.020, 0.033)	0.021 (0.013, 0.033)	0.047 (0.021, 0.102)	0.033 (0.022, 0.049)
Headways	1.026 (1.011, 1.042)	1.001 (0.993, 1.010)	1.048 (1.032, 1.067)	1.026 (1.012, 1.042)	1.060 (1.038, 1.082)	1.082 (1.020, 1.149)	1.050 (1.031, 1.068)
Hard brakes	1.069 (1.057, 1.080)	1.241 (1.194, 1.293)	1.163 (1.140, 1.188)	1.047 (1.040, 1.054)	1.114 (1.093, 1.138)	1.086 (1.049, 1.131)	1.183 (1.154, 1.211)
Rolling stability	1.528 (1.367, 1.733)	1.648 (1.269, 2.229)	1.676 (1.467, 1.951)	1.419 (1.284, 1.578)	2.477 (1.590, 3.717)	4.320 (2.210, 9.522)	1.175 (1.039, 1.369)
Collision mitigation	1.163 (1.127, 1.203)	1.318 (1.132, 1.540)	1.362 (1.292, 1.440)	1.212 (1.174, 1.252)	1.577 (1.422, 1.766)	1.134 (0.952, 1.353)	1.170 (1.121, 1.234)
Age	0.992 (0.989, 0.995)	0.988 (0.982, 0.993)	0.993 (0.990, 0.996)	0.995 (0.993, 0.998)	0.986 (0.982, 0.990)	0.999 (0.989, 1.010)	0.997 (0.993, 1.000)
Mean speed	0.976 (0.970, 0.983)	1.016 (1.005, 1.027)	0.968 (0.962, 0.974)	0.994 (0.987, 1.000)	1.000 (0.988, 1.012)	0.973 (0.958, 0.988)	0.983 (0.973, 0.994)
Gender: male	0.883 (0.702, 1.083)	0.868 (0.631, 1.227)	0.844 (0.716, 0.997)	0.749 (0.650, 0.862)	0.841 (0.691, 1.029)	0.675 (0.433, 1.027)	0.751 (0.634, 0.893)
Gender: unknown	1.065 (0.589, 1.908)	1.378 (0.706, 2.617)	0.576 (0.325, 0.980)	1.287 (0.774, 2.079)	0.194 (0.044, 0.626)		0.816 (0.571, 1.158)
Fit statistics:							
sample size	6950	1797	7405	6429	3339	943	4963
elpd_loo	-9300.8 (125.3)	-2416.9 (52.2)	-9799.1 (112.1)	-7624.2 (90.8)	-3912.6 (70.5)	-1139.8 (40)	-5293.9 (85.4)
p_loo	30.6 (5)	14.3 (2.5)	20.4 (2.6)	17.4 (2.3)	13.9 (1.7)	11 (1.6)	19.4 (2.9)
looic	18601.6 (250.6)	4833.8 (104.5)	19598.1 (224.2)	15248.3 (181.6)	7825.2 (141)	2279.5 (79.9)	10587.7 (170.8)

Notes: The SCEs were measured as the number of events per 10,000 miles driven.
Incidence rate ratios and their associated 95% credible intervals are reported for all variables (predictors and covariates).
For the fit statistics, (.) indicates the standard error of the computed statistic.

Model validation

Table 4.4 demonstrates the variance inflation factor results for all the four SCEs models. The results capture all three investigated outcomes (crashes, fatalities and injuries) as well as the results stratified by business unit and driver type. Since all VIFs are less than 1.3, there is no evidence of serious multicollinearity issues in all the models.

Table 4.4: *Variance inflation factor test for multicollinearity.*

Outcome	Crash	Fatality	Injury	Crash			Crash		Crash	
Samples	All Drivers			Dedicated			Intermodal		Final-mile	
				Local	Regional	OTR	Local	Regional	Regional	OTR
Headways	1.030	1.030	1.030	1.049	1.024	1.012	1.044	1.040	1.073	1.037
Hard brakes	1.031	1.031	1.031	1.027	1.063	1.047	1.020	1.031	1.077	1.184
Rolling stability	1.024	1.024	1.024	1.029	1.013	1.047	1.024	1.010	1.053	1.159
Collision mitigation	1.038	1.038	1.038	1.024	1.048	1.056	1.056	1.046	1.118	1.236
Age	1.006	1.006	1.006	1.003	1.007	1.005	1.004	1.004	1.004	1.011
Ping speed	1.270	1.270	1.270	1.007	1.012	1.012	1.015	1.015	1.011	1.047
Gender	1.005	1.005	1.005	1.001	1.002	1.005	1.004	1.004	1.010	1.008
Business unit	1.096	1.096	1.096							
Driver type	1.140	1.140	1.140							

All the models and truck drivers have Pareto k diagnostic statistics of less than 0.7 (not shown in the tables), which suggests no signal for model misspecification (Vehtari et al., 2015,0). The estimated effective number of parameters (`p_loo` in Tables 4.1, 4.2, 4.3), were similar to the total number of parameters in the models. These two results suggest that the negative binomial models were reasonably specified models given the large number of observations in this study (Vehtari et al., 2017,0). The LOOIC in the tables can be used to compare different models, with lower values indicating better models. For example, in Table 4.2, the "Four SCEs" model has the lowest LOOIC (79,540.5) among the six models. However, the standard errors of the LOOIC statistic (in the bracket) suggest that the model with all four SCEs was not significantly better than the pooled model.

To investigate the models' predictive accuracy, we adopt the approach of Gelman et al. (2013, Section 6.3) who suggested simulating some function of the data and parameter, and comparing it with the observed value of a particular quantity. For our trucking safety

application, we examined the proportion of zero crashes since it corresponds to a crash-free trip, which is of interest to truck drivers and operators alike. The probability of having zero crashes is, of course, an unknown quantity, but its posterior distribution can be estimated by simulating samples using Hamiltonian Monte Carlo. In this section, we limit our analysis to the models whose outcomes were crashes since the accident and fatality models indicated that our observed events were insufficient for statistical inference (based on the size of the credible interval in Table 4.1).

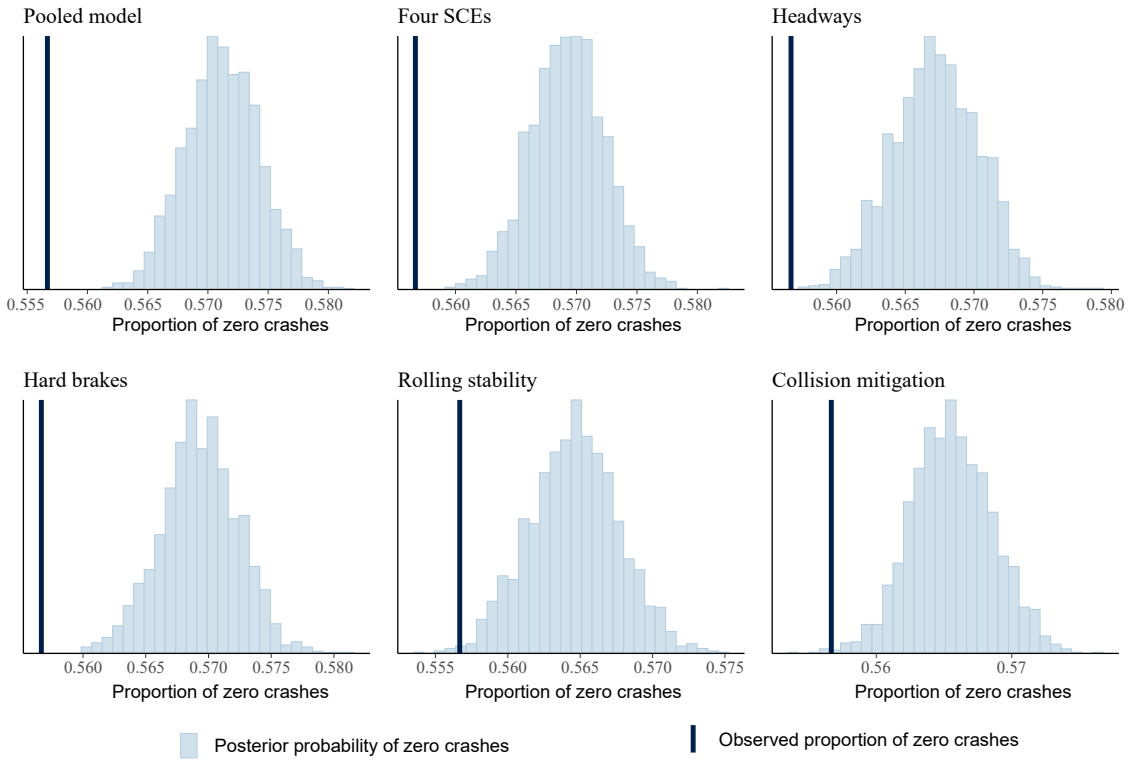


Figure 4.1: *Graphical posterior predictive checks with zero count test statistic for the Bayesian negative binomial models for all drivers. The x-axis is the proportion of zero crashes and y-axis is probability density. The solid black line is the observed proportion, while the light blue histogram is from 100 simulated predictions*

Figure 4.1 shows the posterior distributions, which are indicated by the histograms in light blue, for the posterior probability of zero crashes under each of the six models considered in Table 4.2. The observed proportion of zero crashes is indicated by the vertical line in each part of Figure 4.1. For all six models, the observed proportion of zero crashes was considerably less than what would be predicted by the model. Note that the magnitude

of this prediction bias is small, usually around 0.015. In other words, while both models (with and without business units and driver types) perform reasonably well in predicting the mean numbers of SCEs, the model with business units and driver types does a better job predicting the proportion of zero crashes. This suggests that different business units and driver types should be accounted for in the model.

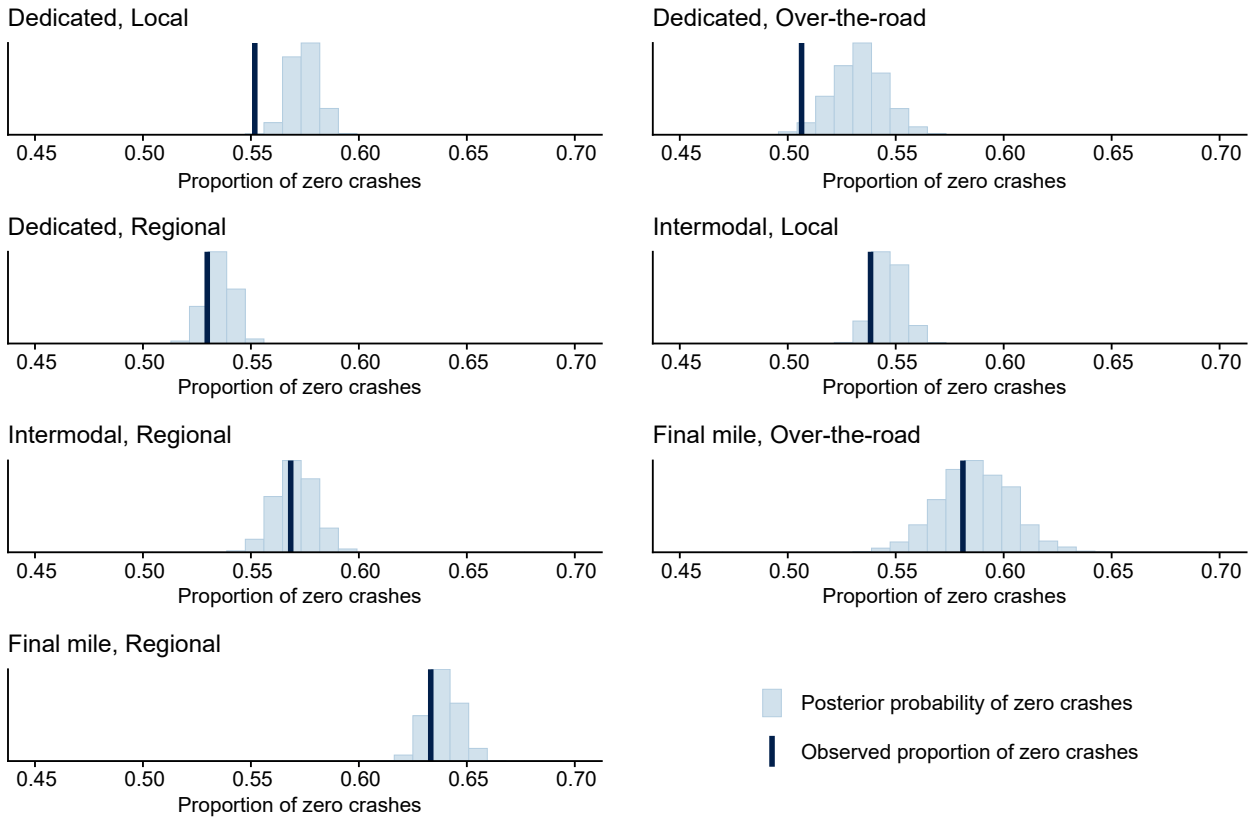


Figure 4.2: *Graphical posterior predictive checks with zero count test statistic for the Bayesian negative binomial models, stratified by business unit and driver types. The x-axis is the proportion of zero crashes and y-axis is probability density. The solid black line is the observed proportion, while the light blue histogram is from 100 simulated predictions.*

Based partly on the result from Figure 4.1, we ran the model with all four SCEs (model 2) separately for each of the seven business units and driver types. The corresponding posterior predictive check for zero crashes is shown in Figure 4.2. Here, the vertical lines are much closer to the simulated posterior distribution. This suggests that different business units and driver types should be accounted for in the model.

Aim 2

Exploratory analysis of cumulative driving time and risk of SCEs

Figure 4.3 presents the univariate relationship between cumulative driving hours and the rate of SCEs (the number of SCEs per 0.5 hour). The black points are SCE rates calculated from the aggregated data, surrounded by 95% confidence intervals (grey bands), and the blue smooth curve is the Locally Weighted Scatterplot Smoothing (LOESS) estimates of the black points. It shows that the rate of SCEs increases as cumulative driving time goes from zero to six hours, while the trend levels off after six hours of cumulative driving. It is worth the attention that the magnitude of change in the y-axis is very small, and this is the raw SCE rate estimate without adjusting for other variables.

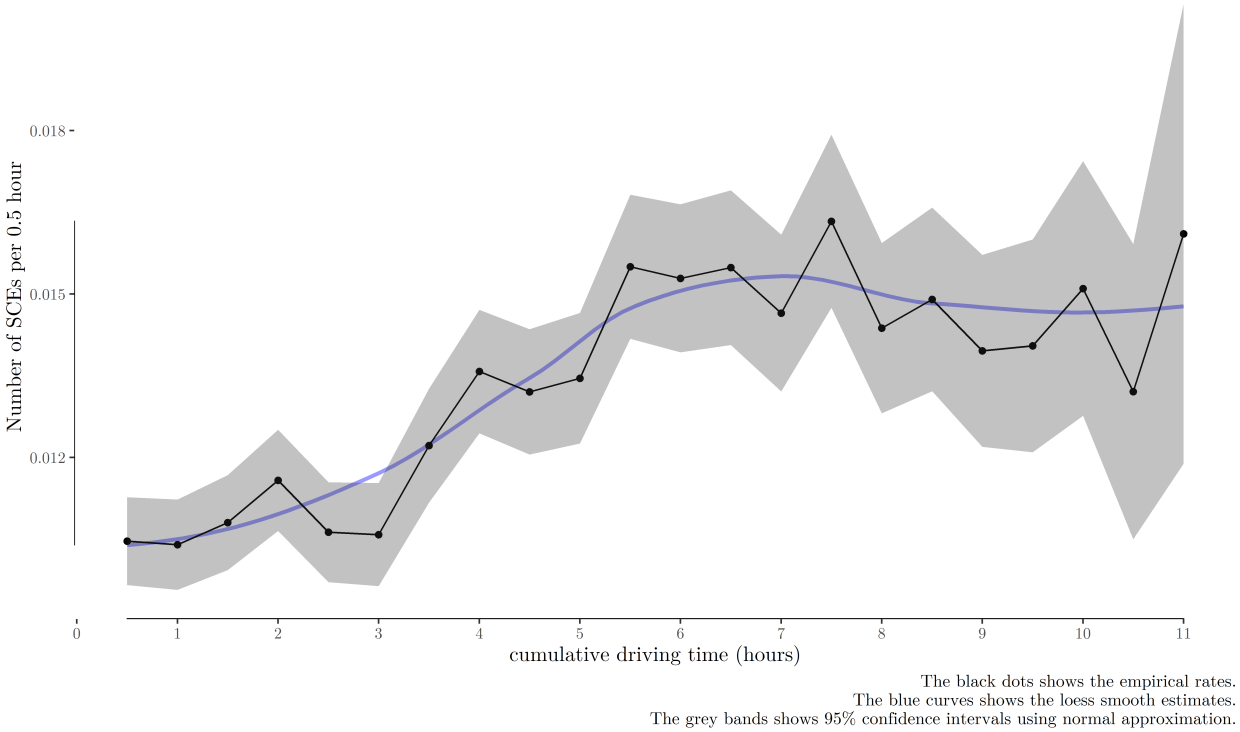
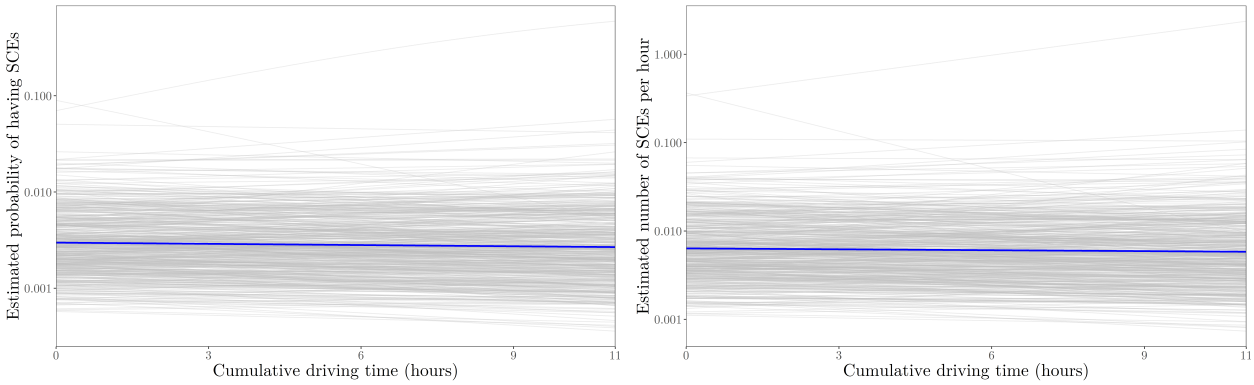


Figure 4.3: *The rate of safety critical events and cumulative driving time among 496 sample regional truck drivers*

Hierarchical logistic and negative binomial models for SCEs

Table 4.5 presents the results of the four statistical models predicting any types of SCEs: (1) logistic regression without random effects, (2) NB regression without random effects, (3) hierarchical logistic regression with driver-level random intercepts and random slopes for cumulative driving time, and (4) hierarchical NB regression with driver-level random intercepts and random slopes. Compared to model (1) and (2) in which most predictors were significant, the predictors in model (3) and (4) were less significant. This reduction in the significance of predictors was because the variation of the outcome variable in model (3) and (4) is explained by the driver-level random effects. In all four models, the estimated parameters for cumulative driving time were not significant and the values were close to zero, indicating that cumulative driving time was not associated with the risk of SCEs among the sample drivers. The estimated values of the hyperparameters (σ_0 and σ_1) were not small, which suggested that there were fair amount of variability across drivers and the mixed-effects models were appropriate.



(a) *Hierarchical Logistics model* ***(b)*** *Hierarchical NB model*

Figure 4.4: *Simulated relationship between cumulative driving time and probability (left)/rate (right) of SCEs the 497 sample drivers. The y-axises are on the log 10 scale.*

To better understand the relationship between cumulative driving time and the risk of SCEs, as well as driver-to-driver variability, we visualized the estimated risk of SCEs and cumulative driving hours for each driver (the grey lines) and the overall trend (the bold blue lines), as shown in Figure 4.4. It is worth noting that the y-axis in the two plots are

Table 4.5: *Estimated results for the standard and hierarchical logistic and NB models*

Predictors	Logistic (1)	NB (2)	Hierarchical logistic (3)	Hierarchical NB (4)
Intercept (μ_0)	-4.924***	-7.175***	-6.229***	-8.872***
	(0.099)	(0.090)	(0.240)	(0.242)
Cumulative driving hours (μ_1)	-0.006	-0.011**	-0.005	-0.005
	(0.005)	(0.005)	(0.007)	(0.007)
Mean speed	-0.00004	-0.0002	0.003***	0.001
	(0.001)	(0.001)	(0.001)	(0.001)
Speed s.d.	0.020***	0.017***	0.023***	0.020***
	(0.001)	(0.001)	(0.001)	(0.001)
Weekend	-0.144***	-0.187***	-0.119***	-0.117***
	(0.033)	(0.035)	(0.035)	(0.035)
Holiday	-0.315***	-0.339***	-0.351***	-0.345***
	(0.108)	(0.111)	(0.108)	(0.109)
Hour of the day		(reference: 9 p.m. - 5 a.m.)		
6 a.m. - 10 a.m.	0.318***	0.247***	0.514***	0.485***
	(0.045)	(0.046)	(0.051)	(0.051)
11 a.m. - 2 p.m.	0.583***	0.583***	0.753***	0.722***
	(0.044)	(0.045)	(0.049)	(0.050)
3 p.m. - 8 p.m.	0.356***	0.376***	0.492***	0.478***
	(0.045)	(0.046)	(0.050)	(0.050)
Age	-0.010***	-0.016***	-0.006	-0.007
	(0.001)	(0.001)	(0.004)	(0.004)
Race		(reference: white)		
Black	-0.060**	-0.127***	0.092	0.094
	(0.025)	(0.026)	(0.105)	(0.109)
Other	0.238***	0.155***	0.378**	0.359*
	(0.042)	(0.046)	(0.180)	(0.187)
Gender: female	-0.303***	-0.364***	-0.099	-0.101
	(0.050)	(0.053)	(0.186)	(0.192)
Precipitation intensity	0.550	0.388	1.029	0.955
	(0.666)	(0.709)	(0.672)	(0.665)
Precipitation probability	-0.204***	-0.181**	-0.040	0.047
	(0.073)	(0.076)	(0.074)	(0.073)
Wind speed	-0.021***	-0.024***	-0.033***	-0.033***
	(0.004)	(0.004)	(0.004)	(0.004)
Visibility	-0.036***	-0.051***	0.007	0.005
	(0.005)	(0.005)	(0.006)	(0.006)
Interval time	0.015***		0.016***	
	(0.002)		(0.002)	
Observations	1,019,482	1,019,482	1,019,482	1,019,482
θ		0.037*** (0.001)		0.151
sd: Intercept (σ_0)			0.955	1.00
sd: cumulative driving (σ_1)			0.076	0.080
cor: μ_0 & μ_1			-0.192	-0.225
Note:				$^{*}p < 0.1$; $^{**}p < 0.05$; $^{***}p < 0.01$

on the log 10 scale not on a linear scale, which is to avoid an overwhelm of grey lines on the lower part of the plots. Both of the two figures suggest that there seems to be no association between cumulative driving time and SCEs among the sample drivers, although there is fair amount of variability in both the intercept and slope across drivers.

Table 4.6 presents the model fit statistics of the four models. Higher log likelihood values and c-statistics indicate better model fit, while lower AIC and BIC values suggest better model fit. All four model fit statistics suggest that the hierarchical logistic regression model has the best fit among the four models. Adding driver-level random effects substantially improved the model fit statistics, with c-statistics increased by 0.17 and 0.169 for the logistic and NB regression models.

Table 4.6: *Model fit statistics for the standard and hierarchical logistic and NB models*

Model	Log likelihood	AIC	BIC	c-statistic	Accuracy	Sensitivity	Specificity
Logistic	$-46,181$	$92,397$	$92,610$	0.602	0.285	0.845	0.281
NB	$-49,500$	$99,035$	$99,248$	0.585			
Hierarchical logistic	$-42,895$	$85,832$	$86,080$	0.770	0.593	0.795	0.591
Hierarchical NB	$-45,830$	$91,701$	$91,950$	0.700			

Although the model fit can be improved substantially by adding driver-level random effects, we should acknowledge that the models are generally underfitting. The models without driver-level random effects have c-statistics of 0.59 and 0.71, which are only slightly higher than a random classification model that has the c-statistics of 0.5. Even for the best fit model, the c-statistic is only 0.76, which is acceptable but not strong enough (a model with the c-statistics of 0.8 is usually viewed as a strong model). The model fit statistics could be further improved by adding other important predictors such as traffic and road geometry, which are current not available or accessible to the research team.

Stratified analyses by SCE types

Table 4.7 presents the coefficients of the hierarchical logistic regression models stratified by four different SCE types (hard brake, headway, collision mitigation, and rolling stability). The model for rolling stability clearly had fitting issues, which suggested by the incredibly large standard errors of holiday and other race. This fitting issue may arise from the rareness of rolling stability (only 631 events out of 1 million 30-minute intervals), so the results for this model is not meaningful here. In the stratified models except for the badly fitted rolling stability model, none of coefficients for cumulative driving hours were statistically significant.

Table 4.7: *Estimated results for hierarchical logistic models for four types of SCEs*

	Hard brake	Headway	Collision mitigation	Rolling stability
	(1)	(2)	(3)	(bad fit) (4)
Intercept (μ_0)	-5.377^{***}	-10.542^{***}	-8.193^{***}	-15.896^{***}
	(0.271)	(0.410)	(0.408)	(1.526)
Cumulative driving hours (μ_1)	−0.0003	0.005	−0.005	-0.401^{***}
	(0.009)	(0.011)	(0.017)	(0.074)
Mean speed	-0.018^{***}	0.038^{***}	-0.020^{***}	0.077^{***}
	(0.001)	(0.002)	(0.002)	(0.010)
Speed s.d.	0.039^{***}	0.028^{***}	0.029^{***}	0.060^{***}
	(0.002)	(0.002)	(0.003)	(0.013)
Weekend	0.104^{**}	-0.388^{***}	-0.081	0.282
	(0.049)	(0.057)	(0.103)	(0.312)
Holiday	-0.071	-0.621^{***}	-0.462	-16.924
	(0.152)	(0.171)	(0.356)	(5,126.211)
Hour of the day		(reference: 9 p.m. - 5 a.m.)		
6 a.m. - 10 a.m.	-0.024	1.017^{***}	1.293^{***}	1.546^{***}
	(0.067)	(0.087)	(0.200)	(0.562)
11 a.m. - 2 p.m.	0.006	1.402^{***}	1.606^{***}	2.177^{***}
	(0.066)	(0.085)	(0.195)	(0.376)
3 p.m. - 8 p.m.	-0.007	1.004^{***}	1.277^{***}	1.250^{***}
	(0.066)	(0.087)	(0.198)	(0.321)
Age	-0.010^{**}	0.002	-0.004	-0.070^{**}
	(0.005)	(0.007)	(0.005)	(0.028)
Race		(reference: white)		
Black	0.278^{**}	-0.197	-0.094	-0.198
	(0.115)	(0.174)	(0.134)	(0.665)
Other	0.517^{***}	-0.161	-0.025	-16.798
	(0.196)	(0.312)	(0.241)	(2,561.048)
Gender: female	-0.266	0.229	-0.208	-0.052
	(0.207)	(0.301)	(0.247)	(1.279)
Precipitation intensity	1.014	-0.001	-0.324	4.647
	(0.799)	(1.505)	(2.678)	(5.048)
Precipitation probability	0.339^{***}	-0.385^{***}	-0.396	-1.097
	(0.097)	(0.134)	(0.255)	(0.732)
Wind speed	-0.030^{***}	-0.033^{***}	-0.037^{***}	-0.073^{*}
	(0.006)	(0.006)	(0.012)	(0.040)
Visibility	-0.009	0.022^{***}	0.002	-0.024
	(0.009)	(0.009)	(0.018)	(0.035)
Interval time	-0.002	0.030^{***}	0.012^{**}	0.117^{***}
	(0.003)	(0.004)	(0.006)	(0.020)
Observations	1,019,482	1,019,482	1,019,482	1,019,482
sd: Intercept (σ_0)	0.999	1.61	0.929	2.08
sd: cumulative driving (σ_1)	0.071	0.088	0.081	0.289
cor: μ_0 & μ_1	−0.285	−0.422	−0.276	1

Note: $^{*}p < 0.1$; $^{**}p < 0.05$; $^{***}p < 0.01$

Table 4.8 shows the model fit statistics of the four stratified models. The predictive performance of the stratified models were improved compared to the not stratified models (the hierarchical logistic model in Table 4.5): the *c*-statistics improved from 0.77 to 0.8 for hard brake, 0.86 for headway, and 0.83 for collision mitigation. Log likelihood, AIC, and BIC

cannot be compared across these models as they are modeling different outcome variables. Accuracy, sensitivity, and specificity cannot be compared across these models either since they depend on the cutoff threshold. Given the improvement in c-statistics, we found that the model predictive performance can be improved if the outcome variable is more homogeneous (keeping the same type of events instead of pooling different types of events).

Table 4.8: *Model fit statistics for the four stratified hierarchical logistic models*

Model	Log likelihood	AIC	BIC	c-statistic	Accuracy	Sensitivity	Specificity
Hard brake	$-20,435$	$40,911$	$41,160$	0.800	0.654	0.829	0.654
Headway	$-20,491$	$41,024$	$41,272$	0.860	0.710	0.860	0.709
Collision mitigation	$-6,593$	$13,228$	$13,476$	0.830	0.614	0.872	0.614
Rolling stability	-596	$1,235$	$1,483$	0.990	0.984	0.971	0.984

Aim 3

Simulation results

The simulation results are shown in the following Table 4.9. For the five sets of drivers ($D = 10, 25, 50, 75, 100$) in each of the three scenarios, the mean of posterior mean estimates, mean of estimation bias $\Delta = |\hat{\mu} - \mu|$, and mean of standard error estimates for parameters $\beta, \kappa, \gamma_1, \gamma_2, \gamma_3$ and hyperparameters μ_0 and σ are calculated. The biases are converging to 0 as the number of drivers increases, and the standard errors also converges to 0 proportional to $\sqrt{D}$ (the square root of the number of drivers), which is consistent with the central limit theorem. These large-scale simulation and estimation results in Table 4.9 suggest that the Stan code I set up for PLP and JPLP is valid and can be applied for real data estimation.

Hierarchical PLP and JPLP

Table 4.10 presents the parameter estimation, 95% credible interval (CI), Gelman-Rubin diagnostic statistics $\hat{R}$, and effective sample size (ESS) for the sample 496 regional drivers. In both the PLP and JPLP estimation, the shape parameter were not significantly different from 1, which suggests no significant pattern of SCE distribution within shifts. In JPLP estimation, the reliability recovery percent parameter κ was close to 0, suggesting that within-shift rests have very minor effects on the reliability of SCEs. Figure 4.5 shows

Table 4.9: *Simulation results for PLP, JPLP, and PLP ← JPLP*

Scenario	D	estimate	γ_1	γ_2	γ_3	β	κ	μ_0	σ_0
PLP	10	mean $\hat{\mu}$	1.0203	0.3095	0.2067	1.1898		0.1718	0.5527
PLP	25	mean $\hat{\mu}$	1.0066	0.3046	0.2012	1.1955		0.1985	0.5220
PLP	50	mean $\hat{\mu}$	1.0040	0.3033	0.2005	1.1983		0.1932	0.5077
PLP	75	mean $\hat{\mu}$	1.0034	0.3004	0.2007	1.1983		0.1974	0.5091
PLP	100	mean $\hat{\mu}$	1.0009	0.3009	0.2003	1.1994		0.1966	0.5042
PLP	10	bias Δ	0.0203	0.0095	0.0067	0.0102		0.0282	0.0527
PLP	25	bias Δ	0.0066	0.0046	0.0012	0.0045		0.0015	0.0220
PLP	50	bias Δ	0.0040	0.0033	0.0005	0.0017		0.0068	0.0077
PLP	75	bias Δ	0.0034	0.0004	0.0007	0.0017		0.0026	0.0091
PLP	100	bias Δ	0.0009	0.0009	0.0003	0.0006		0.0034	0.0042
PLP	10	s.e.	0.0777	0.0696	0.0413	0.0589		0.2401	0.1722
PLP	25	s.e.	0.0459	0.0414	0.0247	0.0360		0.1392	0.0916
PLP	50	s.e.	0.0316	0.0286	0.0172	0.0254		0.0960	0.0610
PLP	75	s.e.	0.0258	0.0232	0.0139	0.0207		0.0784	0.0497
PLP	100	s.e.	0.0220	0.0198	0.0119	0.0179		0.0667	0.0420
JPLP	10	mean $\hat{\mu}$	1.0331	0.3218	0.2092	1.1774	0.8149	0.1599	0.5696
JPLP	25	mean $\hat{\mu}$	1.0158	0.3081	0.2039	1.1869	0.8084	0.1798	0.5219
JPLP	50	mean $\hat{\mu}$	1.0037	0.3012	0.2039	1.1943	0.8032	0.2014	0.5111
JPLP	75	mean $\hat{\mu}$	1.0060	0.3012	0.2006	1.1942	0.8028	0.2057	0.5097
JPLP	100	mean $\hat{\mu}$	1.0048	0.3003	0.2008	1.1957	0.8023	0.1996	0.5041
JPLP	10	bias Δ	0.0331	0.0218	0.0092	0.0226	0.0149	0.0401	0.0696
JPLP	25	bias Δ	0.0158	0.0081	0.0039	0.0131	0.0084	0.0202	0.0219
JPLP	50	bias Δ	0.0037	0.0012	0.0039	0.0057	0.0032	0.0014	0.0111
JPLP	75	bias Δ	0.0060	0.0012	0.0006	0.0058	0.0028	0.0057	0.0097
JPLP	100	bias Δ	0.0048	0.0003	0.0008	0.0043	0.0023	0.0004	0.0041
JPLP	10	s.e.	0.0992	0.0834	0.0498	0.0828	0.0573	0.2556	0.1854
JPLP	25	s.e.	0.0586	0.0477	0.0288	0.0512	0.0360	0.1453	0.0960
JPLP	50	s.e.	0.0406	0.0334	0.0201	0.0366	0.0256	0.0999	0.0647
JPLP	75	s.e.	0.0331	0.0272	0.0164	0.0298	0.0208	0.0812	0.0519
JPLP	100	s.e.	0.0287	0.0233	0.0141	0.0258	0.0179	0.0699	0.0442
PLP ← JPLP	10	mean $\hat{\mu}$	1.1923	0.3645	0.2434	1.0157		0.0766	0.6599
PLP ← JPLP	25	mean $\hat{\mu}$	1.1769	0.3514	0.2374	1.0260		0.1134	0.6053
PLP ← JPLP	50	mean $\hat{\mu}$	1.1718	0.3531	0.2355	1.0266		0.1146	0.5977
PLP ← JPLP	75	mean $\hat{\mu}$	1.1686	0.3511	0.2346	1.0276		0.1126	0.5960
PLP ← JPLP	100	mean $\hat{\mu}$	1.1674	0.3512	0.2349	1.0287		0.1189	0.5925
PLP ← JPLP	10	bias Δ	0.1923	0.0645	0.0434	0.1843		0.1234	0.1599
PLP ← JPLP	25	bias Δ	0.1769	0.0514	0.0374	0.1740		0.0866	0.1053
PLP ← JPLP	50	bias Δ	0.1718	0.0531	0.0355	0.1734		0.0854	0.0977
PLP ← JPLP	75	bias Δ	0.1686	0.0511	0.0346	0.1724		0.0874	0.0960
PLP ← JPLP	100	bias Δ	0.1674	0.0512	0.0349	0.1713		0.0811	0.0925
PLP ← JPLP	10	s.e.	0.1041	0.0946	0.0559	0.0580		0.2952	0.2078
PLP ← JPLP	25	s.e.	0.0609	0.0546	0.0329	0.0354		0.1671	0.1095
PLP ← JPLP	50	s.e.	0.0423	0.0383	0.0230	0.0250		0.1167	0.0743
PLP ← JPLP	75	s.e.	0.0344	0.0310	0.0186	0.0204		0.0946	0.0601
PLP ← JPLP	100	s.e.	0.0297	0.0266	0.0160	0.0177		0.0810	0.0514

Table 4.10: *Parameter estimates, Rhat, and effective sample size (ESS) for PLP and JPLP on the 496 regional truck drivers*

Variables	PLP				JPLP			
	estimate	95% CI	$\hat{R}$	ESS	estimate	95% CI	$\hat{R}$	ESS
μ_0	3.200	(2.556, 3.826)	1.001	2,431	3.424	(2.855, 4.003)	1.001	2,334
σ_0	0.974	(0.899, 1.055)	1.001	10,585	0.968	(0.892, 1.050)	1.000	10,003
β	1.003	(0.983, 1.022)	1.000	5,641	1.014	(0.994, 1.035)	1.001	4,368
κ					0.988	(0.969, 0.999)	1.000	7,356
Age	0.005	(-0.003, 0.014)	1.001	1,586	0.005	(-0.003, 0.014)	1.002	1,865
Race: black	-0.068	(-0.283, 0.145)	1.004	1,523	-0.087	(-0.299, 0.122)	1.003	1,438
Race: other	-0.288	(-0.659, 0.078)	1.002	2,294	-0.317	(-0.684, 0.042)	1.002	2,305
Gender: female	-0.071	(-0.428, 0.293)	1.000	2,779	-0.069	(-0.437, 0.290)	1.003	2,905
Mean speed	0.019	(0.015, 0.022)	1.000	19,673	0.012	(0.010, 0.015)	1.000	20,510
Speed variation	0.010	(0.002, 0.018)	1.000	21,992	0.012	(0.008, 0.017)	1.000	14,427
Precip. intensity	-4.089	(-6.549, -1.561)	1.000	19,732	-2.242	(-3.826, -0.568)	1.000	20,447
Precip. prob.	0.476	(0.255, 0.697)	1.000	18,825	0.160	(-0.004, 0.322)	1.000	19,782
Wind speed	0.022	(0.012, 0.031)	1.000	30,373	0.014	(0.006, 0.023)	1.000	31,700

the histogram of random intercepts γ_{0d} estimated in the hierarchical PLP and JPLP models. It indicates that the random intercepts γ_{0d} are larger in JPLP than those in PLP models. The variability of random intercepts is similar in the two models, given similar estimates of σ_0.

Since the reciprocal of the rate parameter is proportional to the intensity function ($\lambda(t) \sim 1/\theta$), negative parameter estimates suggest positive association with higher rates of SCEs. Table 4.10 suggests that mean speed, speed variation, precipitation probability, and wind speed were negative associated with the intensity of SCEs, while precipitation intensity was positive associated with the intensity of SCEs. The demographic variables were not significantly associated with the intensity of SCEs.

In table 4.10, all the Gelman-Rubin diagnostic statistics $\hat{R}$ were less than 1.1, and the effective sample sizes were all greater than 1,000. Besides, the trace plots of four independent markov chains for the parameters μ_0, σ_0, β, and κ in both PLP and JPLP were well-mixed in Figure 4.6. All these evidence suggests that the Hamiltonian Monte Carlo simulation were stable and converge to the true parameter posterior distributions, and the point and interval estimates were valid and trustworthy.

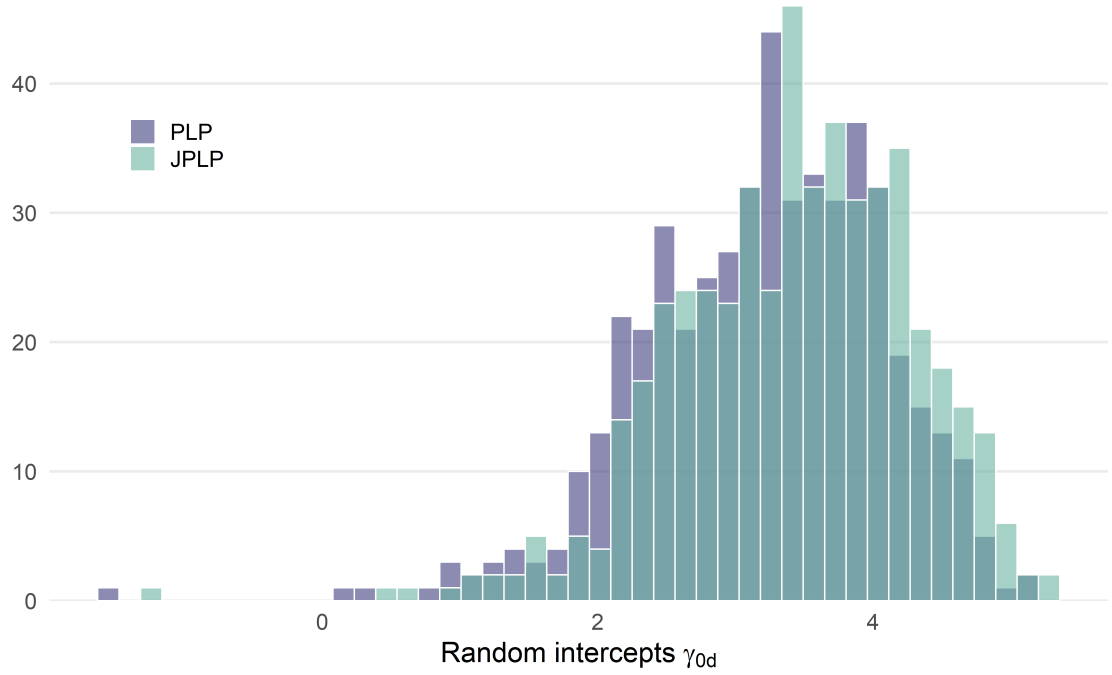


Figure 4.5: *Histogram of random intercepts γ_{0d} across the 496 drivers.*

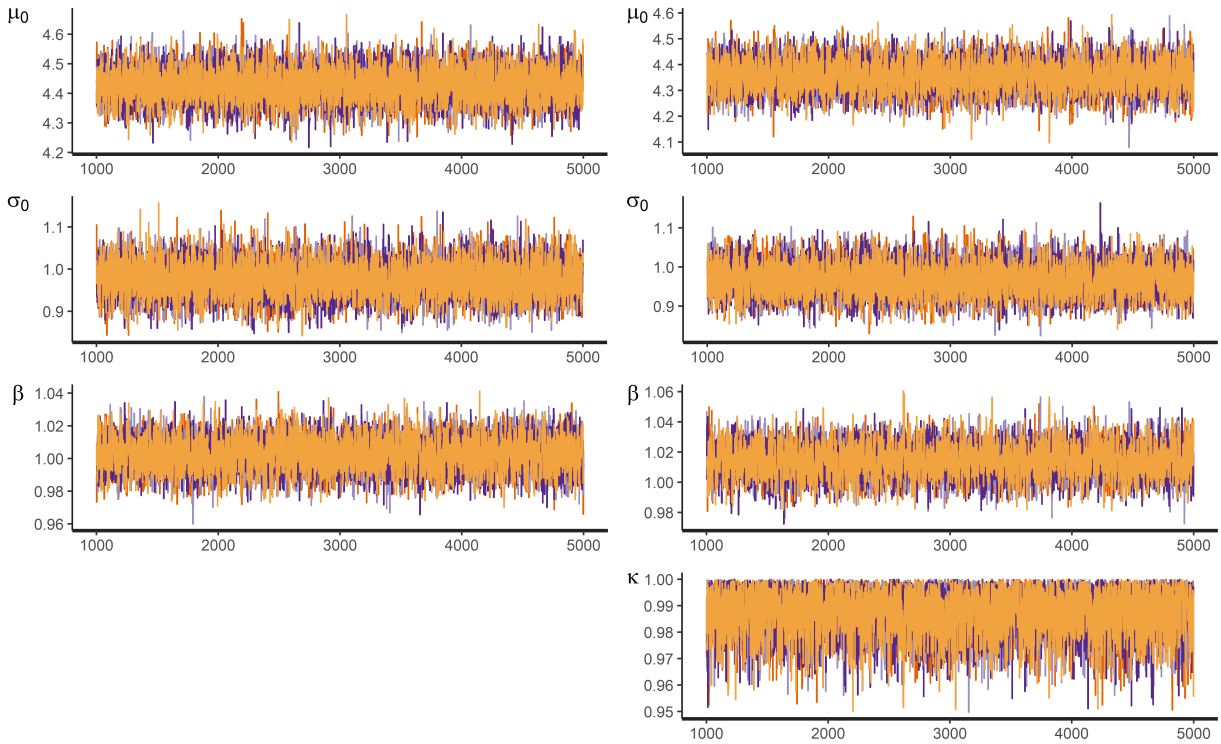


Figure 4.6: *Trace plots of μ_0, σ_0, β, and κ. The left column is trace plots for parameters in PLP; the right column is the trace plot for parameters in JPLP.*

CHAPTER 5 DISCUSSION

Summary of Key Findings

The overarching goal of this dissertation is to construct a generalizable analysis framework for naturalistic driving data among commercial truck drivers, and understand how different factors impact truck driver performance.

Aim 1

Based on around 1.3 billion observations of routinely collected kinematic data from 31,828 truck drivers in a large commercial trucking company, aim 1 examines the association between four types of SCEs (headway, hard brake, collision mitigation, and rolling stability) and crashes, as well as injuries and fatalities. Compared to existing studies on the subject, that are based on up to about 2 million miles driven, our study involves an estimated 2.3 billion miles driven. Bayesian negative binomial models were applied to examine the association between three outcomes (crashes, injuries, and fatalities) and the four SCEs. It was found that a unit increase in the number of any type of SCEs per 10,000 miles was associated with 8.4% (95% credible interval (CI): 8.0-8.8%) increase in crashes per mile and 8.7% (95% CI: 4.8-13.6%) increase in the number of injuries per mile. The increase was different in different types of SCEs: 3.3% (95% CI: 2.6-4%) for headways, 8.1% (95% CI: 7.5-8.7%) for hard brakes, 50.4% (95% CI: 41.4-60%) for rolling stability, and 22.2% (95% CI: 19.8-24.5%) for collision mitigation. The results are consistent when stratified by different business units and driver types. This study provides statistically strong and robust evidence that SCEs are positively associated with crashes and injuries among commercial truck drivers. NDS and kinematic data routinely collected by trucking companies provide a promising opportunity for future data analytic research.

Aim 2

Naturalistic driving studies produce high-resolution, large-scale, and real-world driving data sets, but there is no consistent data aggregation and analysis framework for these

data. Using routinely collected ping and safety-critical events (SCEs) data from 497 commercial truck drivers, this study proposes a driver-centric framework for data cleaning, aggregation, and statistical modeling. We aggregated around 13 million real-time driving ping data into shifts, trips, and 30-minute intervals. Safety-critical events (SCEs), driver demographics, and weather data obtained from a third-party data provider were then merged to the aggregated 30-minute intervals. Driver-centric hierarchical logistic and negative binomial (NB) models with driver-level random intercepts and random slopes for cumulative driving time were used to predict safety-critical events. Although exploratory analysis indicated that the rate of SCEs had a slightly increasing and non-linear (concave down) trend as cumulative driving time increases, the hierarchical models suggested no significant relationship. Stratified analyses by different SCE types showed consistent results. The hierarchical models substantially improved the *c*-statistics compared to their non-hierarchical counterparts (from 0.602 to 0.77 for logistic regression and from 0.585 to 0.7 for NB regression). As more companies are routinely collecting ping and SCEs data, the proposed driver-centric analysis framework provides a powerful tool for future studies to create high-quality evidence to optimize scheduling, empower policy making, and improve transportation safety.

Aim 3

Strengths

Large scale automatically collected data.

Flexible and reproducible to other similar data sets.

Limitations

Traffic data are not available in this study (Wang et al., 2013; Theofilatos and Yannis, 2014; Roshandel et al., 2015). Aim 2 and 3 are restricted within 500 regional drivers, who may not have fatigue issues as serious as over-the-road drivers.

The hierarchical Bayesian models are not scalable with the number of observations exceed-

ing 500K or the number of drivers exceeding 1K.

Public Health Implications

An implication is that the effect of driver-less maneuvers are larger than alerts and actions where the driver may be involved. Conceptually, this is an intuitive result since these maneuvers can be categorized as more "aggressive last-minute" interventions that attempt to mitigate a crash by overriding the driver's control. The coefficients of the covariates suggested that older drivers, higher average speed, male, as well as final-mile and intermodal (compared to dedicated) drivers, were associated with lower rate of crashes.

Future Directions

Subsampling MCMC.

Over-the-road drivers.

With rapid development of automated data collection system, more tall and wide data are becoming commonly available to researchers. A tall dataset has many observations or rows, while a wide dataset has many variables or columns. The emergence of big data poses a threat to the existing MCMC methods, as most of these methods require that the full data likelihood be evaluated at each iteration, which will be computationally intensive in the case of tall and wide data. One way to tackle the computational burden of evaluating the full data likelihood is subsampling MCMC, which means evaluating the likelihood based on multiple subsets of data and then combining the results. Subsampling MCMC via simple random sample often does not work as it does not account for the variability of the log likelihood estimator among different subsamples. The most popular technique of performing subsampling MCMC is via introducing auxiliary variables that reduce the variability of log likelihood estimators (Quiroz et al., 2018b).

The first well-known subsampling MCMC algorithm is the *firefly MCMC* by Maclaurin and Adams (2015), which introduces an auxiliary variable for each observation that can be turned on or off to determine if the observation should be included in likelihood evalu-

ation. Starting from this firefly MCMC algorithm, an increasing number of studies have been published on subsampling MCMC algorithms. Korattikara et al. (2014) proposed to use a sequential hypothesis test to generate *accept-reject samples* with high confidence on a fraction of data. Similar studies that use accept-reject samples include Bardenet et al. (2014) and Bardenet et al. (2017). Another category of widely discussed subsampling MCMC algorithm is Pseudo-Marginal MCMC (PMCMC), which replaces the likelihood or the natural logarithm of likelihood with an unbiased estimate from a subset of data based on control variates at each MCMC iteration (Quiroz et al., 2018b,0). They proposed two types of bias-correction log-likelihood estimates: a) parameter expanded control variates via Taylor expansion around a reference value in parameter space, and b) data expanded control variate via Taylor expansion around the nearest centroid in data space. Other subsampling MCMC algorithms include Block-Poisson estimator (Quiroz et al., 2016), delayed acceptance (Quiroz et al., 2018a), noisy MCMC (Alquier et al., 2016), and zig-zag process MCMC (Bierkens et al., 2019).

Apart from the subsampling MCMC algorithms mentioned above, subsampling MCMC using the Hamiltonian mechanism deserves special attention as it more efficiently explores the posterior in high-dimensional parameter space. Chen et al. (2014) proposed a stochastic gradient HMC, which introduces a friction term that counteracts the effects of noisy gradient. In contrast, Betancourt (2015) argued that the stochastic gradient HMC proposed by Chen et al. (2014) compromised the scalability of the HMC with respect to the complexity of the target distribution. The paper claimed that subsampled data does not have sufficient information to efficiently explore the target distributions, and devastates the scalable performance of HMC. A algorithm called HMC with energy conserving subsampling (HMC-ECS) by Dang et al. (2019) extended the PMCMC algorithm proposed by Quiroz et al. (2019) to HMC by introducing a fictitious momentum vector $\vec{p}$, which has the same dimension as the parameter vector .

APPENDIX

R code for Aim 1
Aggregating ping data into trips

The self-defined function `segment_0()` below is used to separate the fake ping data into trips according to a threshold value.

```
segment_0 = function(speed, threshold, time_diff) {
  speed1 = speed
  speed[time_diff >= threshold] <- 0
  r1 = rle(speed != 0)
  r1$values <- replicate(length(r1$values), 1)
  r1$values <- cumsum(r1$values)
  order_tmp <- inverse.rle(r1)
  dat_tmp1 <- data.table::data.table(speed, order_tmp, time_diff)
  dat_tmp2 <- dat_tmp1[,.(sumdiff = sum(time_diff)), by = order_tmp]
  r2 = rle(speed != 0); first_rle = r2$values[1]
  r2$values[r2$values == 0 & dat_tmp2$sumdiff < threshold] <- TRUE
  r2$values[1] = first_rle
  r2 <- rle(inverse.rle(r2))
  r2$values[r2$values] = cumsum(r2$values[r2$values])
  id = inverse.rle(r2)
  jump_speed = which(id == 0 & speed1 != 0)
  id[jump_speed] = id[jump_speed + 1]
  return(id)
}
```

Bayesian NB regression using `rstanarm`

The code below shows the code for Bayesian NB regression models. For demonstration purpose, we only use the first 1,000 observations of the data, 1 Markov chain with 1,000 iterations and the first 500 of them are warm-up iterations.

```
pacman::p_load(rstanarm, broom)
fit <-
  stan_glm(
    crash ~ SCE + speed + age + gender + bus_unit + d_type,
    offset = log(distance / 1000),
    data = data,
    family = neg_binomial_2,
    prior = normal(0, 10),
    prior_intercept = normal(0, 10),
    QR = TRUE,
    iter = 4000,
    chains = 4,
    cores = 4,
```

```
    seed = 123
  )
```

```
broom::tidy(fit, intervals = TRUE, prob = 0.95) %>%
  mutate(estimate = exp(estimate),
         lower = exp(lower),
         upper = exp(upper)) %>%
  select(term, IRR = estimate, `95% CI left` = lower, `95% CI right` = upper) %>%
  knitr::kable(align = "c",
               caption = "Posterior estimates of Bayesian NB model.")
```

Model comparison and diagnostics using `loo`

```
prop_zero <- function(y) mean(y == 0)
pp_check(fit, plotfun = "stat", stat = "prop_zero")
```

The code above will give a figure showing the posterior predictive checks, which is a measure of the prediction accuracy. It compares the observed data to 100 replicated datasets generated from the posterior parameters distributions. For each simulated dataset, the proportion of zero crashes was computed, and the blue histograms shows the simulated distribution of the proportions. The black solid vertical lines are the observed proportion of zero crashes in observed data. When the observed proportion (black solid line) is near the center of the plot, it demonstrates good model fit.

```
(fit_loo = loo(fit))
```

The above block shows the expected log predicted density (`elpd_loo`), estimate number of parameters (`p_loo`), and the LOO Information Criterion (`looic`) for a new dataset from Pareto smoothed importance-sampling leave-one-out (PSIS-LOO) cross-validation (CV).

```
fit_new <- stan_glm(
  crash ~ SCE + speed + age + gender,
  offset = log(distance / 1000),
  data = data,
  family = neg_binomial_2,
  prior = normal(0, 10),
  prior_intercept = normal(0, 10),
  QR = TRUE,
  iter = 4000,
  chains = 4,
  cores = 4,
  seed = 123
)
fit_new_loo = loo(fit_new)
```

```
loo::compare(fit_loo, fit_new_loo)
```

With two model fits `fit` and `fit_new` above, researchers can compare the model fit using `compare()` from the `loo` package, as shown above. It compares the expected predictive accuracy by the dif-

ference in `elpd_loo`, with positive difference `elpd_diff` suggesting the second model while negative difference favoring the first model.

R code for Aim 2

Read and clean data

```
pacman::p_load(data.table, dplyr, lubridate, fst)

# 1. original ping -> d
d = fread("data/original/Meeeting-11-16-18%2Fpings_500drivers_11.csv") %>%
  .[,.(driver = gsub("\"", "", V7), ping_time = ymd_hms(V2),
       speed = V3, lat = V4, lon = V6)]  %>%
  .[!is.na(ping_time)] %>%
  setkey(driver, ping_time) %>%
  .[,diff := as.integer(difftime(ping_time,
                shift(ping_time, type = "lag",
                      fill = 0), units = "mins")), driver] %>%
  .[,diff := {diff[1] = 0L; diff}, driver]
fst::write_fst(d, "data/cleaned/01a_ping_original_500drivers.fst", compress = 100)

# 1b. create driver list -> d_list
d_list = d[,.(n_ping = .N), driver]
fst::write_fst(d_list, "data/cleaned/00driver_list.fst", compress = 100)

# 2. weather -> w
w = vroom::vroom(dir("data/weather", "\\.csv$", full.names = T)) %>%
  as.data.table() %>%
  .[,DATIME := lubridate::mdy_hm(DATIME)] %>%
  .[,.(DATIME, LATITUDE, LONGITUDE, PRECIP_INTENSITY, PRECIP_PROBABILITY,
       WIND_SPEED, VISIBILITY, SUNRISE_TIME, SUNSET_TIME,
       DURING_SUNSET, DURING_DUSK, DURING_SUNRISE, DURING_DAWN)]
fst::write_fst(w, "data/cleaned/02weather.fst", compress = 100)

# 3. driver demographic -> dinfo
alldr = fread("data/original/ALL_DRIVERS_DATA2016-09-30 10-53-42.csv") %>%
  .[,EMPLID := stringr::str_replace_all(EMPLID, " ", "")]
d_list = fst::read_fst("data/cleaned/driver_list.fst")
dinfo = fread("data/original/ALPHA_TO_EMPLID2016-10-21 14-00-24.csv") %>%
  .[,driver := tolower(ALPHA)] %>%
  .[,`:=`(driver = stringr::str_replace_all(driver, " ", ""),
          EMPLID = stringr::str_replace_all(EMPLID, " ", ""))] %>%
  .[,.(driver, EMPLID)] %>%
  setkey(driver) %>%
  merge(d_list, by = "driver", all.y = TRUE) %>%
  merge(alldr, by = "EMPLID", all.x = TRUE) %>%
  .[,BIRTHDATE := ymd(BIRTHDATE)] %>%
  .[!is.na(BIRTHDATE),] %>%
  setkey(driver) %>%
```

```
  .[,n_missing := rowSums(is.na(.))] %>%
  .[order(driver, -n_missing)] %>%
  .[,head(.SD, 1), by = driver] %>%
  .[, age := 2015 - year(BIRTHDATE)] %>%
  #.[!(driver %in% c("kisi", "codc"))] %>%
  .[,.(driver, EMPLID, age, race = ETHNIC_GROUP, gender = GENDER)] %>%
  .[,race := case_when(race == "BLACK     " ~ "Black",
                       race == "WHITE     " ~ "White",
                       TRUE ~ "Other")]
fst::write_fst(dinfo, "data/cleaned/03driver_information.fst", compress = 100)

# 4. SCEs -> ce
dinfo = fst::read_fst("data/cleaned/03driver_information.fst")
ce = fread("data/original/CRITICAL_EVENT_QUERY2016-09-30 10-58-28.csv") %>%
  .[,`:=`(EMPLID = stringr::str_replace_all(EMPLID, " ", ""),
          EVT_TYP = stringr::str_replace_all(EVT_TYP, " ", ""))] %>%
  .[,event_time := ymd_hms(paste(EVENT_DATE, EVENT_HOUR, sep = " "))] %>%
  .[,.(EMPLID, event_time, event_type = EVT_TYP)] %>%
  merge(dinfo, by = "EMPLID", all.y = TRUE) %>%
  .[!is.na(event_time),.(driver, event_time, event_type)] %>%
  .[,event_type := case_when(event_type == "HEADWAY" ~ "HW",
                             event_type == "HARD_BRAKING" ~ "HB",
                             event_type == "COLLISION_MITIGATION" ~ "CM",
                             event_type == "ROLL_STABILITY" ~ "RS")] %>%
  unique()
fst::write_fst(ce, "data/cleaned/04safety_critical_events.fst", compress = 100)
```

Mark shift and trip ID

```
pacman::p_load(data.table, dplyr, lubridate, fst)
d = fst::read_fst("data/cleaned/01a_ping_original_500drivers.fst") %>%
  as.data.table()

## PART 1: mark all ping with shift_id
threshold_shift = 8*60
s1 = d %>%
  .[diff >= threshold_shift|speed <= 10, speed := 0] %>%
  .[,rleid := rleid(speed != 0), driver] %>%
  .[,`:=`(speed1 = speed)] %>%
  .[,`:=`(sum_speed = sum(speed), sum_time = sum(diff)), .(driver, rleid)] %>%
  .[sum_speed == 0 & sum_time < threshold_shift, speed1 := 3] %>%
  .[,`:=`(sum_speed = sum(speed1)), .(driver, rleid)] %>%
  .[,shift_id := fifelse(sum_speed == 0, 0, rleid(speed1 != 0)), driver] %>%
  .[,`:=`(rev_cums_sp = rev(cumsum(rev(speed))),
          cums_sp = cumsum(speed)), .(driver, shift_id)] %>%
  .[rev_cums_sp == 0|cums_sp == 0, shift_id := 0]
```

```
# STATS: pings
s1[shift_id == 0,.N] # 3,550,935 -> shift_id == 0
s1[,round(sum(shift_id == 0)*100/.N, 2)] # percent of stopping pings: 26.93%
s1[shift_id != 0,.N] # 9,636,349 -> shift_id != 0
s1[,round(sum(shift_id != 0)*100/.N, 2)] # 73.07%

s_len = s1 %>%
  .[shift_id != 0] %>%
  .[,.(driver, shift_id, shift_length = sum(diff)), .(driver, shift_id)]

# STATS: shifts
s_len[,.N] # 77,870 unique shifts
s_len[shift_length > 14*60, .N] # 2,198 very long shifts
s_len[, round(sum(shift_length > 14*60)*100/.N, 2)] # 1.72%
s_len[shift_length <= 0.5*60, .N] # 773 very short shifts
s_len[, round(sum(shift_length <= 0.5*60)*100/.N, 2)] # 0.99%
s_len[shift_length > 0.5*60 & shift_length <= 14*60, .N] # 75,760 eligible shifts
s_len[, round(sum(shift_length > 0.5*60 & shift_length <= 14*60)*100/.N, 2)] # 97.29%


# filter eligible shifts
s2 = s1 %>%
  .[shift_id != 0] %>%
  .[,shift_length := sum(diff), .(driver, shift_id)] %>%
  .[,.(driver, ping_time, speed, lat, lon, shift_id, shift_length)] %>%
  .[shift_length > 30 & shift_length <= 14*60] %>%
  .[,shift_length := NULL] %>%
  setkey(driver, ping_time)

# STATS: pings
s1[shift_id == 0,.N] # 3,550,935 -> shift_id == 0
s1[,sum(shift_id == 0)/.N] # percent of stopping pings: 26.93%
s2[,.N] # 9,349,312
round(s2[,.N]*100/s1[,.N], 2) # 70.9%

fst::write_fst(s2, "data/cleaned/01b_ping_shift_id_500drivers.fst")


## PART 2: mark all ping with trip_id ##
threshold_trip = 30
t1 = s2 %>%
  setkey(driver, ping_time) %>%
  .[,diff := as.integer(difftime(ping_time, shift(ping_time,
              type = "lag",fill = 0), units = "mins")), driver] %>%
  .[,diff := {diff[1] = 0L; diff}, driver] %>%
  .[diff >= threshold_trip, speed := 0] %>%
  .[,rleid := rleid(speed != 0), driver] %>%
```

```
  .[,`:=`(rleid1 = rleid, speed1 = speed)] %>%
  .[,`:=`(sum_speed = sum(speed), sum_time = sum(diff)), .(driver, rleid)] %>%
  .[sum_speed == 0 & sum_time < threshold_trip, speed1 := 3] %>%
  .[,`:=`(sum_speed = sum(speed1)), .(driver, rleid)] %>%
  .[,trip_id := data.table::fifelse(sum_speed == 0, 0,
              rleid(speed1 != 0)), driver] %>%
  .[trip_id != 0,trip_length := sum(diff), .(driver, trip_id)] %>%
  .[,.(driver, ping_time, speed, lat, lon, diff, shift_id, trip_id)]

t2 = t1 %>%
  .[trip_id != 0,] %>%
  .[,`:=`(lon1 = shift(lon, type = "lag", fill = NA),
          lat1 = shift(lat, type = "lag", fill = NA)),
    by = .(driver, shift_id, trip_id)] %>%
  .[,distance := geosphere::distHaversine(cbind(lon, lat), cbind(lon1, lat1))] %>%
  .[,distance := round(distance/1609.344, 3)] %>%
  .[,distance := {distance[1] = 0; distance}, .(driver, shift_id, trip_id)] %>%
  .[,c("lon1", "lat1") := NULL] %>%
  setkey(driver, ping_time)

fst::write_fst(t2, "data/cleaned/01c_ping_trip_id_500drivers.fst")
```

Aggregate data

```
pacman::p_load(data.table, dplyr, lubridate, fst)

d = fst::read_fst("data/cleaned/01c_ping_trip_id_500drivers.fst") %>%
  as.data.table() %>%
  setkey(driver, ping_time)
w = fst::read_fst("data/cleaned/02weather.fst") %>% as.data.table()
dinfo = fst::read_fst("data/cleaned/03driver_information.fst") %>% as.data.table()
ce = fst::read_fst("data/cleaned/04safety_critical_events.fst") %>% as.data.table()

# Merge weather & driver information to ping
d1 = d %>%
  .[,`:=`(ping_id = 1:.N,
          DATIME = lubridate::floor_date(ping_time, "hours"),
          LATITUDE = as.numeric(gsub("([0-9]+\\.[0-9]{2})(.*)", "\\1", lat)),
          LONGITUDE = as.numeric(gsub("([0-9]+\\.[0-9]{2})(.*)", "\\1", lon)))] %>%
  merge(w, by = c("DATIME", "LATITUDE", "LONGITUDE"), all.x = TRUE) %>%
  merge(dinfo[,.(driver, age, race, gender)], by = "driver", all.x = TRUE)

########################  trip  ########################
dtrip = d1 %>%
  .[trip_id != 0,] %>%
  setkey(driver, ping_time) %>%
  .[,.(start_time = ping_time[1], end_time = ping_time[.N],
```

```
       start_lat = LATITUDE[1], start_lon = LONGITUDE[1],
       end_lat = LATITUDE[.N], end_lon = LONGITUDE[.N],
       speed_mean = mean(speed, na.rm = TRUE),
       speed_sd = sd(speed, na.rm = TRUE),
       distance = sum(distance, na.rm = TRUE),
       age = age[1], race = race[1], gender = gender[1],
       prep_inten = mean(PRECIP_INTENSITY, na.rm = TRUE),
       prep_prob = mean(PRECIP_PROBABILITY, na.rm = TRUE),
       wind_speed = mean(WIND_SPEED, na.rm = TRUE),
       visibility = mean(VISIBILITY, na.rm = TRUE),
       sunrise = fifelse(sum(DURING_SUNRISE == "Y", na.rm = TRUE) > 0, 1, 0),
       sunset = fifelse(sum(DURING_SUNSET == "Y", na.rm = TRUE) > 0, 1, 0),
       dusk = fifelse(sum(DURING_DUSK == "Y", na.rm = TRUE) > 0, 1, 0),
       dawn = fifelse(sum(DURING_DAWN == "Y", na.rm = TRUE) > 0, 1, 0)),
    .(driver, shift_id, trip_id)] %>%
  .[,`:=`(trip_time = as.integer(difftime(end_time, start_time,
                                          units = "mins")),
          speed_sd  = fifelse(is.na(speed_sd), 0, speed_sd),
          prep_inten = fifelse(is.na(prep_inten), 0, prep_inten),
          prep_prob = fifelse(is.na(prep_prob), 0, prep_prob),
          wind_speed = fifelse(is.na(wind_speed), mean(wind_speed, na.rm = TRUE),
                               wind_speed),
          visibility = fifelse(is.na(visibility), mean(visibility, na.rm = TRUE),
                               visibility))] %>%
  .[order(driver, trip_id)]

trip_range = dtrip %>%
  .[,.(driver, shift_id, trip_id, start_time, end_time)] %>%
  setkey(driver, start_time, end_time)

n_CE_trip = ce %>%
  .[,dummy := event_time] %>%
  setkey(driver, event_time, dummy) %>%
  foverlaps(trip_range, mult = "all", type = "within", nomatch = NA) %>%
  .[!is.na(shift_id)|!is.na(trip_id),] %>%
  .[,.(driver, shift_id, trip_id, event_time, event_type)] %>%
  .[,.(nCE = .N), .(driver, shift_id, trip_id)] %>%
  setkey(driver, shift_id, trip_id)

dtrip1 = dtrip %>%
  merge(n_CE_trip, by = c('driver', 'shift_id', 'trip_id'), all.x = TRUE) %>%
  .[, nCE := fifelse(is.na(nCE), 0, nCE)]
fst::write_fst(dtrip1, "data/cleaned/12dtrip.fst", compress = 100)

########################  shift  ########################
dshift = d1 %>%
  .[,.(start_time = ping_time[1], end_time = ping_time[.N],
```

```
      start_lat = LATITUDE[1], start_lon = LONGITUDE[1],
      end_lat = LATITUDE[.N], end_lon = LONGITUDE[.N]),
    .(driver, shift_id)] %>%
  .[,shift_time := as.integer(difftime(end_time, start_time,
                                       units = "mins"))] %>%
  .[order(driver, shift_id)]
fst::write_fst(dshift, "data/cleaned/13dshift.fst", compress = 100)

######################## intervals ########################
dtrip = fst::read_fst("data/cleaned/12dtrip.fst") %>% as.data.table()
dshift = fst::read_fst("data/cleaned/13dshift.fst") %>% as.data.table()

source("data/function_mkint.R")
agg_int30 = mkint(30)
agg_int60 = mkint(60)

# add critical events
ce = fst::read_fst("data/cleaned/04safety_critical_events.fst") %>% as.data.table()
source("data/function_indexce.R")
ceindexed30 = indexce(agg_int30, ce)
ceindexed60 = indexce(agg_int60, ce)


ce_int30 = agg_int30 %>%
  merge(ceindexed30[,.(nCE = .N), .(driver, interval_id)],
        by = c("driver", "interval_id"), all.x = TRUE) %>%
  .[,nCE := fifelse(is.na(nCE), 0, nCE)]
ce_int60 = agg_int60 %>%
  merge(ceindexed60[,.(nCE = .N), .(driver, interval_id)],
        by = c("driver", "interval_id"), all.x = TRUE) %>%
  .[,nCE := fifelse(is.na(nCE), 0, nCE)]

# delete shifts with more than 11 hours of driving time
ce_int30_11 = ce_int30 %>%
  .[,max_drive := sum(interval_time), .(driver, shift_id)] %>%
  .[max_drive <= 11*60]
ce_int60_11 = ce_int60 %>%
  .[,max_drive := sum(interval_time), .(driver, shift_id)] %>%
  .[max_drive <= 11*60]
```

Hierarchical logisitc and negative binomial models

```
pacman::p_load(data.table, dplyr, ggplot2, fst, MASS)
d = fst::read_fst("data/cleaned/32interval30_CE_11hours_limit.fst") %>% as.data.table()

z = d %>%
  .[,`:=`(cumdrive = cumdrive/60,
```

```
          CE_binary = fifelse(nCE > 0, 1, 0),
          race = factor(race, levels = c("White", "Black", "Other")),
          gender = factor(gender, levels = c("M", "F", "U")))]

f_logit = glm(CE_binary ~ cumdrive + speed_mean + speed_sd + age + race + gender +
                prep_inten + prep_prob + wind_speed + visibility + interval_time,
              family = "binomial", data = z)
saveRDS(f_logit, "fit/f_logit.rds")

f_nb = glm.nb(nCE ~ cumdrive + speed_mean + speed_sd + age + race + gender +
                prep_inten + prep_prob + wind_speed + visibility +
                offset(log(interval_time)),
                data = z)
saveRDS(f_nb, "fit/f_nb.rds")
```

R code for Aim 3

Functions for simulating PLP and JPLP data

```
# Function: simulating PLP - time truncated case
sim_plp_tau = function(tau = 30,
                       beta = 1.5,
                       theta = 10){
  # initialization
  s = 0; t = 0
  while (max(t) <= tau) {
    u <- runif(1)
    s <- s - log(u)
    t_new <- theta*s^(1/beta)
    t <- c(t, t_new)
  }
  t = t[c(-1, -length(t))]

  return(t)
}

# Function: simulate multiple NHPPs
sim_hier_plp_tau = function(N, beta = 1.5, theta){
  t_list = list()
  len_list = list()
  tau_vector = rnorm(N, 10, 1.3)

  for (i in 1:N) {
    t_list[[i]] = sim_plp_tau(tau_vector[i], beta = beta, theta = theta[i])
    len_list[[i]] = length(t_list[[i]])
  }

  event_dat = data.frame(
    shift_id = rep(1:N, unlist(len_list)),
    event_time = Reduce(c, t_list)
```

```
  )

  start_end_dat = data.frame(
    shift_id = 1:N,
    start_time = rep(0, N),
    end_time = tau_vector #difference2
  )

  return(list(event_dat = event_dat,
              start_end_dat = start_end_dat,
              shift_length = unlist(len_list)))
}

# Function: Simulating hierarchical PLP data for D drivers
sim_hier_nhpp = function(
  beta = 1.5,              # Shape parameter for PLP
  D = 10,                  # the number of drivers
  K = 3,                   # the number of predictor variables
  group_size_lambda = 10, # the mean number of shifts for each driver
  mu0 = 0.2,               # Hyperparameters: mean
  sigma0 = 0.5,            # Hyperparameters: s.e.
  R_K = c(1, 0.3, 0.2)     # Fixed-effects parameters
)
{
  # 1. Random-effect intercepts
  r_0D = rnorm(D, mean = mu0, sd = sigma0)

  # 3. The number of shifts in the $d$-th driver: $N_{d}$
  N_K = rpois(D, group_size_lambda)
  N = sum(N_K) # the total number of obs
  id = rep(1:D, N_K)

  # 4. Generate data: x_1, x_2, .. x_K
  sim1 = function(group_sizes = N_K)
  {
    ntot = sum(group_sizes)

    int1 = rep(1, ntot)
    x1 = rnorm(ntot, 1, 1)
    x2 = rgamma(ntot, 1, 1)
    x3 = rpois(ntot, 2)

    return(data.frame(int1, x1, x2, x3))
  }
  X = sim1(N_K)

  # 5. Scale parameters of a NHPP
```

```
  # 5a. parameter matrix: P
  P = cbind(r0 = rep(r_0D, N_K),
            t(replicate(N, R_K)))
  M_logtheta = P*X

  # returned parameter for each observed shift
  theta_vec = exp(rowSums(M_logtheta))

  df = sim_hier_plp_tau(N = N, beta = beta, theta = theta_vec)

  hier_dat = list(
    N = nrow(df$event_dat),
    K = K,
    S = nrow(df$start_end_dat),
    D = max(id),
    id = id,     # driver index at shift level
    tau = df$start_end_dat$end_time,
    event_time = df$event_dat$event_time,
    group_size = df$shift_length,  # the number of events in each shift
    X_predictors = X[,2:4]
  )

  true_params = list(
    mu0 = mu0, sigma0 = sigma0,
    r0 = r_0D, r1_rk = R_K,
    beta = beta,
    theta = theta_vec
  )

  return(list(hier_dat = hier_dat, true_params = true_params))
}
```

JPLP

```
# Function: sample the number of stops from 1:4
get_n_stop = function() sample(1:4, 1, TRUE)

# Function: Define a inverse function for mean function Lambda
inverse = function (f, lower = 0.0001, upper = 10000) {
  function (y) uniroot((function (x) f(x) - y), lower = lower, upper = upper)[1]
}

# Function: Mean function Lambda for PLP
Lambda_PLP = function(t, beta = 1.5, theta = 4) return((t/theta)^beta)

# Function: Mean function Lambda for JPLP
Lambda_JPLP = function(
  t,                          # Time of the event
```

```
  tau = 12,                  # Shift end time (right-censor time)
  kappa = 0.8,               # "Jump" parameter in JPLP
  t_trip = c(3.5, 6.2, 9),   # trip stop time
  beta = 1.5,                # Shape parameter
  theta = 4)                 # Rate parameter
{
  t_trip1 = c(0, t_trip)
  n_trip = length(t_trip1)
  comp = Lambda_PLP(t_trip, beta, theta)
  kappa_vec0 = rep(kappa, n_trip - 1)^(0:(n_trip - 2))
  kappa_vec1 = rep(kappa, n_trip - 1)^(1:(n_trip - 1))
  cum_comp0 = comp*kappa_vec0
  cum_comp1 = comp*kappa_vec1
  index_trip = max(cumsum(t > t_trip1)) - 1

  if(index_trip == 0){
    return((t/theta)^beta)
  }else{
    return(sum(cum_comp0[1:index_trip]) - sum(cum_comp1[1:index_trip]) +
             kappa^index_trip*(t/theta)^beta)
  }
}

# Function: sim_jplp:  simulate event times generated from a JPLP
sim_jplp = function(
  tau0 = 12,                   # Shift end time (right-censor time)
  kappa0 = 0.8,                # "Jump" parameter in JPLP
  t_trip0 = c(3.5, 6.2, 9),    # trip stop time
  beta0 = 1.2,                 # Shape parameter
  theta0 = 0.5                 # Rate parameter
)
{ s = 0; t = 0
  Lambda1 = function(t, tau1 = tau0, kappa1 = kappa0, t_trip1 = t_trip0,
                     beta1 = beta0, theta1 = theta0)
    {
    return(Lambda_JPLP(t, tau = tau1, kappa = kappa1,
                       t_trip = t_trip1, beta = beta1, theta = theta1))
    }
  inv_Lambda = inverse(Lambda1, 0.0001, 10000)

  while (max(t) <= tau0)
    {
    u <- runif(1)
    s <- s - log(u)
    t_new <- inv_Lambda(s)$root
    t <- c(t, t_new)
    }
```

```
  t = t[c(-1, -length(t))]

  return(t)
}


# Function: sim_mul_jplp: simulate event times for multiple shifts
sim_mul_jplp = function(
  kappa = 0.8,       # "Jump" parameter in JPLP
  beta = 1.2,        # Shape parameter
  theta = 2,         # Rate parameter
  n_shift = 10       # Number of shifts
)
{
  t_shift_vec = list()
  n_trip_vec = list()
  id_trip_vec = list()
  t_start_vec = list()
  t_stop_vec = list()
  n_event_shift_vec = list()
  t_event_vec = list()
  n_event_trip_vec = list()

  for (i in 1:n_shift) {
    sim_tau = rnorm(1, 10, 1.3)
    n_stop = get_n_stop()
    sim_t_trip = round((1:n_stop)*sim_tau/(n_stop + 1) +
                         rnorm(n_stop, 0, sim_tau*0.15/n_stop), 2)
    t_events = sim_jplp(tau0 = sim_tau,
                        kappa0 = kappa,
                        t_trip0 = sim_t_trip,
                        beta0 = beta,
                        theta0 = theta)
    t_shift_vec[[i]] = sim_tau
    n_trip_vec[[i]] = n_stop + 1
    id_trip_vec[[i]] = 1:(n_stop + 1)
    t_start_vec[[i]] = c(0, sim_t_trip)
    t_stop_vec[[i]]  = c(sim_t_trip, sim_tau)
    n_event_shift_vec[[i]] = length(t_events)
    t_event_vec[[i]] = t_events

    tmp_n_event_trip = rep(NA_integer_, (n_stop + 1))
    for (j in 1:(n_stop + 1)) {
       tmp_n_event_trip[j] = sum(t_events > t_start_vec[[i]][j] &
                                   t_events <= t_stop_vec[[i]][j])
    }
    n_event_trip_vec[[i]] = tmp_n_event_trip
```

```
  }

  event_dt = data.frame(
    shift_id = rep(1:n_shift, unlist(n_event_shift_vec)),
    trip_id = rep(Reduce(c, id_trip_vec), Reduce(c, n_event_trip_vec)),
    event_time = Reduce(c, t_event_vec)
  )

  trip_dt = data.frame(
    shift_id = rep(1:n_shift, Reduce(c, n_trip_vec)),
    trip_id = Reduce(c, id_trip_vec),
    t_trip_start = Reduce(c, t_start_vec),
    t_trip_end = Reduce(c, t_stop_vec),
    N_events = Reduce(c, n_event_trip_vec)
  )

  shift_dt = data.frame(
    shift_id = 1:n_shift,
    start_time = rep(0, n_shift),
    end_time = Reduce(c, t_shift_vec)
  )

  stan_dt = list(N = nrow(event_dt),
                 S = nrow(trip_dt),
                 r_trip = trip_dt$trip_id,
                 t_trip_start = trip_dt$t_trip_start,
                 t_trip_end = trip_dt$t_trip_end,
                 event_time = event_dt$event_time,
                 group_size = trip_dt$N_events)

  return(list(event_dt = event_dt,
              trip_dt = trip_dt,
              shift_dt = shift_dt,
              stan_dt = stan_dt))

}

# Function: sim_hier_JPLP:  hierarchical JPLP for D different drivers
sim_hier_JPLP = function(
  beta = 1.2,               # Shape parameter for JPLP
  kappa = 0.8,              # "jump" parameter in JPLP
  D = 10,                   # the number of drivers
  K = 3,                    # the number of predictor variables
  group_size_lambda = 10,   # the mean number of shifts for each driver
  mu0 = 0.2,                # hyperparameter 1
  sigma0 = 0.5,             # hyperparameter 2
  R_K = c(1, 0.3, 0.2)      # Fixed-effects parameters
```

```
)
{
  # 1. Random-effect intercepts
  r_0D = rnorm(D, mean = mu0, sd = sigma0)

  # 3. The number of observations (shifts) in the $d$-th driver: $N_{d}$
  N_K = rpois(D, group_size_lambda)
  N = sum(N_K) # the total number of shifts for all D drivers
  # id = rep(1:D, N_K)

  # 4. Generate data: x_1, x_2, .. x_K
  simX = function(group_sizes = N_K)
  {
    ntot = sum(group_sizes)

    int1 = rep(1, ntot)
    x1 =  rnorm(ntot, 1, 1)
    x2 = rgamma(ntot, 1, 1)
    x3 =  rpois(ntot, 2)

    return(data.frame(int1, x1, x2, x3))
  }
  X = simX(N_K)

  # 5. Scale parameters of a JPLP
  # 5a. parameter matrix: P
  P = cbind(r0 = rep(r_0D, N_K), t(replicate(N, R_K)))
  M_logtheta = P*X
  theta = exp(rowSums(M_logtheta))

  # Initialization of lists
  t_shift_vec = list()
  n_trip_vec = list()
  id_trip_vec = list()
  t_start_vec = list()
  t_stop_vec = list()
  n_event_shift_vec = list()
  t_event_vec = list()
  n_event_trip_vec = list()

  for (i in 1:N)
  {
    sim_tau = rnorm(1, 10, 1.3)
    n_stop = get_n_stop()
    sim_t_trip = round((1:n_stop)*sim_tau/(n_stop + 1) +
                         rnorm(n_stop, 0, sim_tau*0.15/n_stop), 2)
    t_events = sim_jplp(tau0 = sim_tau,
```

```
                      kappa0 = kappa,
                      t_trip0 = sim_t_trip,
                      beta0 = beta,
                      theta0 = theta[i])
  t_shift_vec[[i]] = sim_tau                # end time for each shift
  n_trip_vec[[i]] = n_stop + 1              # number of trips
  id_trip_vec[[i]] = 1:(n_stop + 1)         # index for trips
  t_start_vec[[i]] = c(0, sim_t_trip)       # start time for each trip
  t_stop_vec[[i]]  = c(sim_t_trip, sim_tau) # end time for each trip
  n_event_shift_vec[[i]] = length(t_events) # number of events for each shift
  t_event_vec[[i]] = t_events               # time of SCEs

  # Create a vector of number of SCEs for each trip
  tmp_n_event_trip = rep(NA_integer_, (n_stop + 1))
  for (j in 1:(n_stop + 1)) {
    tmp_n_event_trip[j] = sum(t_events > t_start_vec[[i]][j] &
                              t_events <= t_stop_vec[[i]][j])
  }
  n_event_trip_vec[[i]] = tmp_n_event_trip
}

# shifts data
shift_dt = data.frame(
  driver_id = rep(1:D, N_K),
  shift_id = 1:N,
  start_time = rep(0, N),
  end_time = Reduce(c, t_shift_vec),
  n_trip = Reduce(c, n_trip_vec),
  n_event = Reduce(c, n_event_shift_vec)
)

# trips data set
trip_dt = data.frame(
  driver_id = rep(shift_dt$driver_id, shift_dt$n_trip),
  shift_id = rep(1:N, Reduce(c, n_trip_vec)),
  trip_id = Reduce(c, id_trip_vec),
  t_trip_start = Reduce(c, t_start_vec),
  t_trip_end = Reduce(c, t_stop_vec),
  N_events = Reduce(c, n_event_trip_vec)
)

# TEMPORARY vector: a temporary vector for events per driver
n_event_driver = shift_dt %>%
  group_by(driver_id) %>%
  summarise(n_event = sum(n_event)) %>%
  pull(n_event)
# TEMPORARY vector: a temporary vector for # of trips per driver
```

```
  n_trip_driver = trip_dt %>%
    group_by(driver_id) %>%
    summarise(n_trip = length(shift_id)) %>%
    pull(n_trip)

  # events data set
  event_dt = data.frame(
    driver_id = rep(1:D, n_event_driver),
    shift_id = rep(1:N, Reduce(c, n_event_shift_vec)),
    event_time = Reduce(c, t_event_vec)
  )

  stan_dt = list(
    N = nrow(event_dt),
    K = K,
    S = nrow(trip_dt),
    D = D,
    id = rep(1:D, n_trip_driver),
    #driver index, must be an array
    r_trip = trip_dt$trip_id,
    t_trip_start = trip_dt$t_trip_start,
    t_trip_end = trip_dt$t_trip_end,
    event_time = event_dt$event_time,
    group_size = trip_dt$N_events,
    X_predictors = as.matrix(X[rep(row.names(X), shift_dt$n_trip), 2:4])
  )

  stan_jplp_dt_for_plp = list(
    N = nrow(event_dt),
    K = K,
    S = nrow(shift_dt),
    D = D,
    id = rep(1:D, N_K), # driver index at shift level
    tau = shift_dt$end_time,
    event_time = event_dt$event_time,
    group_size = shift_dt$n_event, # the number of events in each shift
    X_predictors = X[,2:4]
  )

  return(list(event_time = event_dt,
              trip_time = trip_dt,
              shift_time = shift_dt,
              stan_dt = stan_dt,
              stan_jplp_dt_for_plp = stan_jplp_dt_for_plp))
}

# Function: pull_use: pull wanted estimates from the posterior distributions
```

```r
pull_use = function(var = "theta", est_obj = f){
  z = est_obj %>%
    broom::tidy() %>%
    filter(grepl(var, term))
  return(z)
}
```

Stan code for estimating PLP and JPLP data

This following code chunk demonstrates the Stan code to estimate the parameters of a NHPP with PLP intensity function.

```
functions{
  real nhpp_log(vector t, real beta, real theta, real tau){
    vector[num_elements(t)] loglik_part;
    real loglikelihood;
    for (i in 1:num_elements(t)){
      loglik_part[i] = log(beta) - beta*log(theta) + (beta - 1)*log(t[i]);
    }
    loglikelihood = sum(loglik_part) - (tau/theta)^beta;
    return loglikelihood;
  }
  real nhppnoevent_lp(real tau, real beta, real theta){
    real loglikelihood = - (tau/theta)^beta;
    return(loglikelihood);
  }
}
data {
  int<lower=1> N;                // total # of failures
  int<lower=1> K;                // number of predictors
  int<lower=1> S;                // total # of shifts
  int<lower=1> D;                // total # of drivers
  int<lower=1> id[S];            // driver index, must be an array
  vector<lower=0>[S] tau;        // truncated time
  vector<lower=0>[N] event_time; // failure time
  int group_size[S];             // group sizes
  matrix[S, K] X_predictors;     // predictor variable matrix
}
transformed data{
  matrix[S, K] X_centered;
  vector[K] X_means;
  for(k0 in 1:K){
    X_means[k0] = mean(X_predictors[, k0]);
    X_centered[,k0] = X_predictors[, k0] - X_means[k0];
  }
}
parameters{
  real mu0;             // hyperparameter: mean
```

```
  real<lower=0> sigma0; // hyperparameter: s.e.
  real<lower=0> beta;   // shape parameter
  vector[K] R1_K;       // fixed parameters each of K predictors
  vector[D] R0;         // random intercept for each of D drivers
}
model{
  int position = 1;
  vector[S] theta_temp;

  for (s0 in 1:S){
    theta_temp[s0] = exp(R0[id[s0]] + X_centered[s0,]*R1_K);
  }

  for (s1 in 1:S){
    if(group_size[s1] == 0) {
      target += nhppnoevent_lp(tau[s1], beta, theta_temp[s1]);
    }else{
      segment(event_time, position, group_size[s1]) ~ nhpp(beta, theta_temp[s1], tau[s1]);
      position += group_size[s1];
    }
  }
  beta ~ gamma(1, 1);
  R0 ~ normal(mu0, sigma0);
  R1_K  ~ normal(0, 10);
  mu0 ~ normal(0, 10);
  sigma0 ~ gamma(1, 1);
}
generated quantities{
  real mu0_true = mu0 - dot_product(X_means, R1_K);
  vector[D] R0_true = R0 - dot_product(X_means, R1_K);
  //real theta_correct = theta_temp - dot_product(X_centered, R1_K);
}
```

Different from NHPP with PLP intensity function, in which the likelihood function was evaluated by shifts, this JPLP likelihood function is evaluated by TRIPS, which are nested within shifts. In this way, the likelihood function can be evaluated using the `segment` function in Stan.

```
// Stan code to estimate a hierchical JPLP process
functions{
  // LogLikelihood function for shifts with events (N_{event} > 0)
  real jplp_log(vector t_event, // time of SCEs
                real trip_start,
                real trip_end,
                int r,     // trip index
                real beta,
                real theta,
                real kappa)
  {
```

```
    vector[num_elements(t_event)] loglik;
    real loglikelihood;
    for (i in 1:num_elements(t_event))
    {
      loglik[i] = (r - 1)*log(kappa) + log(beta) - beta*log(theta) +
            (beta - 1)*log(t_event[i]);
    }
    loglikelihood = sum(loglik) -
          kappa^(r - 1)*theta^(-beta)*(trip_end^beta - trip_start^beta);
    return loglikelihood;
  }
  // LogLikelihood function for shifts with no event (N_{event} = 0)
  real jplpoevent_lp(real trip_start,
                     real trip_end,
                     int r,
                     real beta,
                     real theta,
                     real kappa)
  {
    real loglikelihood = - kappa^(r - 1)*theta^(-beta)*(trip_end^beta -
                   trip_start^beta);
    return(loglikelihood);
  }
}
data {
  int<lower=0> N;                   // total # of events
  int<lower=1> D;                   // total # of drivers
  int<lower=1> K;                   // number of predictors
  int<lower=0> S;                   // total # of trips, not shifts!!
  int<lower=1> id[S];               // driver index, must be an array
  int r_trip[S];                    // index of trip $r$
  vector<lower=0>[S] t_trip_start;  // trip start time
  vector<lower=0>[S] t_trip_end;    // trip end time
  vector<lower=0>[N] event_time;    // failure time
  int group_size[S];                // group sizes
  matrix[S, K] X_predictors;        // predictor variable matrix
}
transformed data{
  matrix[S, K] X_centered;
  vector[K] X_means;
  for(k0 in 1:K){
    X_means[k0] = mean(X_predictors[, k0]);
    X_centered[,k0] = X_predictors[, k0] - X_means[k0];
  }
}
parameters{
  real mu0;                     // hyperparameter
  real<lower=0> sigma0;         // hyperparameter
```

```
  real<lower=0> beta;           // Shape parameter
  real<lower=0, upper=1> kappa; // Jump parameter
  vector[K] R1_K;               // fixed parameters for K predictors
  vector[D] R0;                 // random intercept for D drivers
}
model{
  int position = 1;
  vector[S] theta_temp;

  for (s0 in 1:S){
    theta_temp[s0] = exp(R0[id[s0]] + X_centered[s0,]*R1_K);
  }

  for (s1 in 1:S){ // Likelihood estimation for JPLP based on trips, not shifts
    if(group_size[s1] == 0){
      target += jplpoevent_lp(t_trip_start[s1], t_trip_end[s1],
            r_trip[s1], beta, theta_temp[s1], kappa);
      }else{
      segment(event_time, position, group_size[s1]) ~ jplp_log(t_trip_start[s1],
            t_trip_end[s1], r_trip[s1], beta, theta_temp[s1], kappa);
      position += group_size[s1];
    }
  }
//PRIORS
  beta ~ gamma(1, 1);
  kappa ~ uniform(0, 1);
  R0 ~ normal(mu0, sigma0);
  R1_K  ~ normal(0, 10);
  mu0 ~ normal(0, 10);
  sigma0 ~ gamma(1, 1);
}
generated quantities{
  real mu0_true = mu0 - dot_product(X_means, R1_K);
  vector[D] R0_true = R0 - dot_product(X_means, R1_K);
  //real theta_correct = theta_temp - dot_product(X_centered, R1_K);
}
```

Scale up PLP and JPLP simulation

This following chunk demonstrates how to scale up the JPLP data simulation and JPLP Stan estimation to 1000 simulations.

```
N_sim = 1000

set.seed(123)
#  D = 10
sim10 = list()
for (i in 1:N_sim) {
```

```
  print(paste0("D = 10, progress: ",
               round(i*100/N_sim, 2),
               "% (", i, " out of 1000)"))

  tryCatch({z = sim_hier_JPLP(beta = 1.2, D = 10)
  fit0 = stan("stan/jplp_hierarchical.stan",
              chains = 1, iter = 3000, refresh = 0,
              data = z$stan_dt, seed = 123
  )}, error=function(e){})

  sim10[[i]] = pull_use("beta|kappa|mu0_true|sigma0|R1_K", fit0)
}
data.table::fwrite(data.table::rbindlist(sim10),
                   "fit/JPLP_fit_sim_hierarchical/sim10.csv")

#  D = 25
sim25 = list()
for (i in 1:N_sim) {
  print(paste0("D = 25, progress: ",
               round(i*100/N_sim, 2),
               "% (", i, " out of 1000)"))

  tryCatch({z = sim_hier_JPLP(beta = 1.2, kappa = 0.8,
                              mu0 = 0.2, sigma0 = 0.5,
                              R_K = c(1, 0.3, 0.2), D = 25)},
           error=function(e){})

  tryCatch({fit0 = stan("stan/jplp_hierarchical.stan",
                        chains = 1, iter = 4000, refresh = 0,
                        data = z$stan_dt, seed = 123)},
           error=function(e){})

  sim25[[i]] = pull_use("beta|kappa|mu0_true|sigma0|R1_K", fit0)
}
data.table::fwrite(data.table::rbindlist(sim25),
                   "fit/JPLP_fit_sim_hierarchical/sim25.csv")

#  D = 50
sim50 = list()
for (i in 1:N_sim) {
  print(paste0("D = 50, progress: ",
               round(i*100/500, 2),
               "% (", i, " out of 500)"))

  tryCatch({z = sim_hier_JPLP(beta = 1.2, kappa = 0.8,
                              mu0 = 0.2, sigma0 = 0.5,
                              R_K = c(1, 0.3, 0.2), D = 50)},
```

```
          error=function(e){})

  tryCatch({fit0 = stan("stan/jplp_hierarchical.stan",
                        chains = 1, iter = 4000, refresh = 0,
                        data = z$stan_dt, seed = 123)},
           error=function(e){})

  sim50[[i]] = pull_use("beta|kappa|mu0_true|sigma0|R1_K", fit0)
}
data.table::fwrite(data.table::rbindlist(sim50),
                   "fit/JPLP_fit_sim_hierarchical/sim50.csv")

#  D = 75
sim75 = list()
for (i in 1:N_sim) {
  print(paste0("D = 75, progress: ",
               round(i*100/N_sim, 2),
               "% (", i, " out of 1000)"))

  tryCatch({z = sim_hier_JPLP(beta = 1.2, kappa = 0.8,
                              mu0 = 0.2, sigma0 = 0.5,
                              R_K = c(1, 0.3, 0.2), D = 75)},
           error=function(e){})

  tryCatch({fit0 = stan("stan/jplp_hierarchical.stan",
                        chains = 1, iter = 4000, refresh = 0,
                        data = z$stan_dt, seed = 123)},
           error=function(e){})

  sim75[[i]] = pull_use("beta|kappa|mu0_true|sigma0|R1_K", fit0)
}
data.table::fwrite(data.table::rbindlist(sim75),
                   "fit/JPLP_fit_sim_hierarchical/sim75.csv")

#  D = 100
sim100 = list()
for (i in 1:N_sim) {
  print(paste0("D = 100, progress: ", round(i*100/N_sim, 2),
               "% (", i, " out of 1000)"))

  tryCatch({z = sim_hier_JPLP(beta = 1.2, kappa = 0.8, mu0 = 0.2,
                              sigma0 = 0.5, R_K = c(1, 0.3, 0.2), D = 100)},
           error=function(e){})

  tryCatch({fit0 = stan("stan/jplp_hierarchical.stan",
                        chains = 1, iter = 4000, refresh = 0,
                        data = z$stan_dt, seed = 123)},
```

```
           error=function(e){})

  sim100[[i]] = pull_use("beta|kappa|mu0_true|sigma0|R1_K", fit0)
}
data.table::fwrite(data.table::rbindlist(sim100),
                   "fit/JPLP_fit_sim_hierarchical/sim100.csv")
```

Bayesian hierarchical reliability models: real data estimation

```
# *******************************************************
# *********       NHPP real data estimation     **********

# *********  Run Stan with some simulated data  **********
source('Functions/NHPP_functions.R')
df = sim_hier_nhpp(D = 5, beta = 1.2)
fit0 = stan("Stan/nhpp_plp_hierarchical.stan",
            chains = 1, iter = 1000, data = df$hier_dat, refresh = 1)
broom::tidy(fit0)

# ******************   Read in data    ******************
dnhpp = as.data.table(read_fst('Data/dnhpp.fst'))
djplp = as.data.table(read_fst('Data/djplp.fst'))
sce = read_fst('Data/sce.fst') %>%
  dplyr::select(driver_id, shift_id_num, trip_id_num, t_trip_start,
                t_trip_end, T2SCE_trip, event_type) %>%
  left_join(djplp[,.(driver_id, shift_id_num, trip_id_num, tau)],
            by = c('driver_id', 'shift_id_num', 'trip_id_num')) %>%
  mutate(T2SCE = T2SCE_trip + t_trip_start) %>%
  mutate(T2SCE = fifelse(T2SCE == 0, 0.1, T2SCE)) %>%
  as.data.table()

# ************  Create a list data for stan  *************
dt_nhpp = list(
  N = sce[,.N],
  K = 9,
  S = dnhpp[,.N],
  D = dnhpp[,.N,driver_id][,.N],
  id = dnhpp[,driver_id],
  tau = dnhpp[,tau/60],
  event_time = sce[,T2SCE/60],
  group_size = dnhpp[,N_SCE],
  X_predictors = dnhpp[,.(age, Black, Other_Race, Female, speed_mean,
                          speed_sd, prep_inten,  prep_prob, wind_speed)]
)

# *********  Run Stan with real data  **********
start_time = Sys.time()
```

```r
fit_NHPP = stan("Stan/nhpp_plp_hierarchical.stan", data = dt_nhpp, seed = 123,
                chains = 4, cores = 4, iter = 5000, warmup = 1000, refresh = 1)
(Time_diff = Sys.time() - start_time)
broom::tidy(fit_NHPP)
saveRDS(fit_NHPP, 'Fit/fit_NHPP.rds')
```

```r
# *******************************************************
# *********       JPLP real data estimation     **********
# *********  Run Stan with some simulated data  **********
source('Functions/JPLP_functions.R')
sim_df = sim_hier_JPLP(D = 10, beta = 1.2)
fit0 = stan("Stan/jplp_hierarchical.stan",
            chains = 1, iter = 1000, data = sim_df$stan_dt, refresh = 1)
broom::tidy(fit0)

# ******************   Read in data    *******************
djplp = as.data.table(read_fst('Data/djplp.fst'))
sce = read_fst('Data/sce.fst') %>%
  dplyr::select(driver_id, shift_id_num, trip_id_num, t_trip_start,
                t_trip_end, T2SCE_trip, event_type) %>%
  mutate(T2SCE = t_trip_start + T2SCE_trip) %>%
  mutate(T2SCE = fifelse(T2SCE == 0, 0.1, T2SCE)) %>%
  as.data.table()

# ************  Create a list data for stan  *************
dt_JPLP = list(
  N = sce[,.N],
  K = 9,
  S = djplp[,.N],
  D = djplp[,.N,driver_id][,.N],
  id = djplp[,driver_id],
  r_trip = djplp[,trip_id_num],
  t_trip_start = djplp[,t_trip_start/60],
  t_trip_end = djplp[,t_trip_end/60],
  event_time = sce[,T2SCE/60],
  group_size = djplp[,N_SCE],
  X_predictors = djplp[,.(age, Black, Other_Race, Female, speed_mean,
                          speed_sd, prep_inten,  prep_prob, wind_speed)]
)

# *********  Run Stan with real data  **********
nchain = 4
n_iter = 5000

start_time = Sys.time()
fit_JPLP = stan("Stan/jplp_hierarchical.stan", data = dt_JPLP, seed = 123,
                chains = nchain, cores = nchain, iter = n_iter,
                warmup = 1000, refresh = 1)
```

```r
(Time_diff = Sys.time() - start_time)
broom::tidy(fit_JPLP)
saveRDS(fit_JPLP, 'Fit/fit_JPLP.rds')
```

REFERENCES

Abdel-Aty, M. A., Hassan, H. M., Ahmed, M., and Al-Ghamdi, A. S. (2012). Real-time prediction of visibility related crashes. *Transportation research part C: emerging technologies*, 24:288–298.

Ahmed, M. M., Franke, R., Ksaibati, K., and Shinstine, D. S. (2018). Effects of truck traffic on crash injury severity on rural highways in wyoming using bayesian binary logit models. *Accident Analysis & Prevention*, 117:106–113.

Åkerstedt, T. (1988). Sleepiness as a consequence of shift work. *Sleep*, 11(1):17–34.

Al-Ghamdi, A. S. (2007). Experimental evaluation of fog warning system. *Accident Analysis & Prevention*, 39(6):1065–1072.

Alden, A. S., Mayer, B., Mcgowen, P., Sherony, R., and Takahashi, H. (2016). Animal-vehicle encounter naturalistic driving data collection and photogrammetric analysis. Technical report, SAE Technical Paper.

Alquier, P., Friel, N., Everitt, R., and Boland, A. (2016). Noisy monte carlo: Convergence of markov chains with approximate transition kernels. *Statistics and Computing*, 26(1-2):29–47.

Ameratunga, S., Hijar, M., and Norton, R. (2006). Road-traffic injuries: confronting disparities to address a global-health problem. *The Lancet*, 367(9521):1533–1540.

American Automobile Association Foundation for Traffic Safety (2010). Asleep at the Wheel: The Prevalence and Impact of Drowsy Driving. https://www.aaafoundation.org/sites/default/files/2010DrowsyDrivingReport_1.pdf. [Online; accessed 20-February-2019].

Anderson, J. R., Ogden, J. D., Cunningham, W. A., and Schubert-Kabban, C. (2017). An exploratory study of hours of service and its safety impact on motorists. *Transport Policy*, 53:161–174.

Baker, C. and Reynolds, S. (1992). Wind-induced accidents of road vehicles. *Accident Analysis & Prevention*, 24(6):559–575.

Bardenet, R., Doucet, A., and Holmes, C. (2014). Towards scaling up Markov chain Monte Carlo: an adaptive subsampling approach .

Bardenet, R., Doucet, A., and Holmes, C. (2017). On Markov chain Monte Carlo methods for tall data. *The Journal of Machine Learning Research*, 18(1):1515–1557.

Barnard, Y., Utesch, F., van Nes, N., Eenink, R., and Baumann, M. (2016). The study design of udrive: the naturalistic driving study across europe for cars, trucks and scooters. *European Transport Research Review*, 8(2):14.

Barp, A., Briol, F.-X., Kennedy, A. D., and Girolami, M. (2018). Geometry and dynamics for markov chain monte carlo. *Annual Review of Statistics and Its Application*, 5:451–471.

Basagaña, X., Escalera-Antezana, J. P., Dadvand, P., Llatje, Ò., Barrera-Gómez, J., Cunillera, J., Medina-Ramón, M., and Pérez, K. (2015). High ambient temperatures and risk of motor vehicle crashes in catalonia, spain (2000–2011): a time-series analysis. *Environmental health perspectives*, 123(12):1309–1316.

Bates, D., Mächler, M., Bolker, B., and Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48.

Bendix® (2007). Bendix® ABS-6 Advanced with ESP® Stability System - frequently asked

questions to help you make an intelligent investment in stability. Bendix Commercial Vehicle Systems LLC, a member of the Knorr-Bremse Group. https://www.bendix.com/media/documents/products_1/absstability/truckstractors/StabilityFAQ.pdf. [Published March 2007; accessed April 19, 2020].

Betancourt, M. (2015). The fundamental incompatibility of scalable hamiltonian monte carlo and naive data subsampling. In *International Conference on Machine Learning*, pages 533–540.

Betancourt, M. (2017). A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434*.

Betancourt, M. and Girolami, M. (2015). Hamiltonian monte carlo for hierarchical models. *Current trends in Bayesian methodology with applications*, 79:30.

Bierkens, J., Fearnhead, P., Roberts, G., et al. (2019). The zig-zag process and super-efficient sampling for bayesian analysis of big data. *The Annals of Statistics*, 47(3):1288–1320.

Braver, E. R., Zador, P. L., Thum, D., Mitter, E. L., Baum, H. M., and Vilardo, F. J. (1997). Tractor-trailer crashes in indiana: A case-control study of the role of truck configuration. *Accident Analysis & Prevention*, 29(1):79–96.

Breiman, L. et al. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, 16(3):199–231.

Campbell, K. L. (1991). Fatal accident involvement rates by driver age for large trucks. *Accident Analysis & Prevention*, 23(4):287–295.

Cantor, D. E., Corsi, T. M., Grimm, C. M., and Özpolat, K. (2010). A driver focused truck crash prediction model. *Transportation Research Part E: Logistics and Transportation Review*, 46(5):683–692.

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of statistical software*, 76(1).

Cavuoto, L. and Megahed, F. (2017). Understanding fatigue: Implications for worker safety. *Professional Safety*, 62(12):16–19.

Center, O. S. (1987). Ohio supercomputer center.

Chang, L.-Y. and Chen, W.-C. (2005). Data mining of tree-based models to analyze freeway accident frequency. *Journal of safety research*, 36(4):365–375.

Chen, C. and Xie, Y. (2014). Modeling the safety impacts of driving hours and rest breaks on truck drivers considering time-dependent covariates. *Journal of safety research*, 51:57–63.

Chen, C., Zhang, G., Liu, X. C., Ci, Y., Huang, H., Ma, J., Chen, Y., and Guan, H. (2016a). Driver injury severity outcome analysis in rural interstate highway crashes: a two-level bayesian logistic regression interpretation. *Accident Analysis & Prevention*, 97:69–78.

Chen, G. X., Fang, Y., Guo, F., and Hanowski, R. J. (2016b). The influence of daily sleep patterns of commercial truck drivers on driving performance. *Accident analysis & prevention*, 91:55–63.

Chen, T., Fox, E., and Guestrin, C. (2014). Stochastic gradient hamiltonian monte carlo. In *International conference on machine learning*, pages 1683–1691.

Clarke, D. D., Ward, P., Bartle, C., and Truman, W. (2006). Young driver accidents in the

uk: The influence of age, experience, and time of day. *Accident Analysis & Prevention*, 38(5):871–878.

Cooper, P. (1984). Experience with traffic conflicts in canada with emphasis on "post encroachment time" techniques. In *International calibration study of traffic conflict techniques*, pages 75–96. Springer.

Craye, C., Rashwan, A., Kamel, M. S., and Karray, F. (2016). A multi-modal driver fatigue and distraction assessment system. *International Journal of Intelligent Transportation Systems Research*, 14(3):173–194.

Crum, M. R. and Morrow, P. C. (2002). The influence of carrier scheduling practices on truck driver fatigue. *Transportation Journal*, pages 20–41.

Dalal, K., Lin, Z., Gifford, M., and Svanström, L. (2013). Economics of global burden of road traffic injuries and their relationship with health system variables. *International journal of preventive medicine*, 4(12):1442.

Dang, K.-D., Quiroz, M., Kohn, R., Tran, M.-N., and Villani, M. (2019). Hamiltonian monte carlo with energy conserving subsampling. *Journal of Machine Learning Research*, 20(100):1–31.

Davis, A., Hacker, E., Savolainen, P. T., and Gates, T. J. (2015). Longitudinal analysis of rural interstate fatalities in relation to speed limit policies. *Transportation research record*, 2514(1):21–31.

Dement, W. C. (1997). The perils of drowsy driving. *The New England Journal of Medicine*, 337(11):783–784.

Department of Transportation, Utah (2019). TRUCKS NEED MORE TIME TO STOP. https://www.udot.utah.gov/trucksmart/motorist-home/stopping-distances/. [Online; accessed 20-February-2019].

Di Milia, L., Smolensky, M. H., Costa, G., Howarth, H. D., Ohayon, M. M., and Philip, P. (2011). Demographic factors, fatigue, and driving accidents: An examination of the published literature. *Accident Analysis & Prevention*, 43(2):516–532.

Dingus, T. A., Guo, F., Lee, S., Antin, J. F., Perez, M., Buchanan-King, M., and Hankey, J. (2016). Driver crash risk factors and prevalence evaluation using naturalistic driving data. *Proceedings of the National Academy of Sciences*, 113(10):2636–2641.

Dingus, T. A., Hanowski, R. J., and Klauer, S. G. (2011). Estimating crash risk. *Ergonomics in Design*, 19(4):8–12.

Dingus, T. A., Klauer, S. G., Neale, V. L., Petersen, A., Lee, S. E., Sudweeks, J., Perez, M. A., Hankey, J., Ramsey, D., Gupta, S., et al. (2006a). The 100-car naturalistic driving study. phase 2: Results of the 100-car field experiment. Technical report, United States. Department of Transportation. National Highway Traffic Safety.

Dingus, T. A., Neale, V. L., Klauer, S. G., Petersen, A. D., and Carroll, R. J. (2006b). The development of a naturalistic data collection system to perform critical incident analysis: an investigation of safety and fatigue issues in long-haul trucking. *Accident Analysis & Prevention*, 38(6):1127–1136.

Dong, C., Clarke, D. B., Yan, X., Khattak, A., and Huang, B. (2014). Multivariate random-parameters zero-inflated negative binomial regression model: An application to estimate crash frequencies at intersections. *Accident Analysis & Prevention*, 70:320–329.

Dong, C., Dong, Q., Huang, B., Hu, W., and Nambisan, S. S. (2017). Estimating factors

contributing to frequency and severity of large truck–involved crashes. *Journal of Transportation Engineering, Part A: Systems*, 143(8):04017032.

Dong, C., Nambisan, S. S., Richards, S. H., and Ma, Z. (2015). Assessment of the effects of highway geometric design features on the frequency of truck involved crashes using bivariate regression. *Transportation Research Part A: Policy and Practice*, 75:30–41.

Dowle, M. and Srinivasan, A. (2019). *data.table: Extension of 'data.frame'*. R package version 1.12.8.

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. (1987). Hybrid Monte Carlo. *Physics letters B*, 195(2):216–222.

Duke, J., Guest, M., and Boggess, M. (2010). Age-related safety in professional heavy vehicle drivers: A literature review. *Accident Analysis & Prevention*, 42(2):364–371.

Dunson, D. B. (2001). Commentary: practical advantages of Bayesian analysis of epidemiologic data. *American Journal of Epidemiology*, 153(12):1222–1226.

Eenink, R., Barnard, Y., Baumann, M., Augros, X., and Utesch, F. (2014). Udrive: the european naturalistic driving study. In *Proceedings of Transport Research Arena*. IFSTTAR.

Evans, L. (2014). Traffic fatality reductions: United states compared with 25 other countries. *American journal of public health*, 104(8):1501–1507.

Evans, L. and Wasielewski, P. (1982). Do accident-involved drivers exhibit riskier everyday driving behavior? *Accident Analysis & Prevention*, 14(1):57–64.

Evans, L. and Wasielewski, P. (1983). Risky driving related to driver and vehicle characteristics. *Accident Analysis & Prevention*, 15(2):121–136.

Federal Motor Carrier Safety Administration (2017). Summary of hours of service regulations. [Online; accessed 20-February-2019].

FMCSA (2018a). Large Truck and Bus Crash Facts 2016. https://www.fmcsa.dot.gov/sites/fmcsa.dot.gov/files/docs/safety/data-and-statistics/398686/ltbcf-2016-final-508c-may-2018.pdf. [Online; accessed 20-February-2019].

FMCSA (2018b). Large Truck and Bus Crash Facts 2017. https://www.fmcsa.dot.gov/safety/data-and-statistics/large-truck-and-bus-crash-facts-2017. [Online; accessed 20-June-2019].

Gabry, J. and Goodrich, B. (2016). rstanarm: Bayesian applied regression modeling via stan. *R package version*, 2(1).

Gelfand, A. E., Dey, D. K., and Chang, H. (1992). Model determination using predictive distributions with implementation via sampling-based methods. Technical report, Department of Statistics, Stanford University.

Gelfand, A. E. and Smith, A. F. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American statistical association*, 85(410):398–409.

Gelman, A. and Hill, J. (2006). *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press.

Gelman, A., Rubin, D. B., et al. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–472.

Gelman, A., Stern, H. S., Carlin, J. B., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian data analysis*. Chapman and Hall/CRC.

Geman, S. and Geman, D. (1987). Stochastic relaxation, gibbs distributions, and the

bayesian restoration of images. In *Readings in computer vision*, pages 564–584. Elsevier.

Ghasemzadeh, A. and Ahmed, M. M. (2017). A probit-decision tree approach to analyze effects of adverse weather conditions on work zone crash severity using second strategic highway research program roadway information dataset. Technical report, Transportation Research Board 96th Annual Meeting.

Gilks, W., Richardson, S., and Spiegelhalter, D. (1996). *Model determination using sampling-based methods*, chapter 9, pages 145–161. Chapman and Hall, London.

Girotto, E., de Andrade, S. M., González, A. D., and Mesas, A. E. (2016). Professional experience and traffic accidents/near-miss accidents among truck drivers. *Accident Analysis & Prevention*, 95:299–304.

Gitelman, V., Bekhor, S., Doveh, E., Pesahov, F., Carmel, R., and Morik, S. (2018). Exploring relationships between driving events identified by in-vehicle data recorders, infrastructure characteristics and road crashes. *Transportation research part C: emerging technologies*, 91:156–175.

Gordon, T. J., Kostyniuk, L. P., Green, P. E., Barnes, M. A., Blower, D., Blankespoor, A. D., and Bogard, S. E. (2011). Analysis of crash rates and surrogate events: unified approach. *Transportation research record*, 2237(1):1–9.

Graham, D. J. and Glaister, S. (2003). Spatial variation in road pedestrian casualties: the role of urban scale, density and land-use mix. *Urban Studies*, 40(8):1591–1607.

Grimes, D. A. and Schulz, K. F. (2005). Compared to what? finding controls for case-control studies. *The Lancet*, 365(9468):1429–1433.

Grove, K., Atwood, J., Hill, P., Fitch, G., DiFonzo, A., Marchese, M., and Blanco, M. (2015). Commercial motor vehicle driver performance with adaptive cruise control in adverse weather. *Procedia Manufacturing*, 3:2777–2783.

Guo, F. (2019). Statistical methods for naturalistic driving studies. *Annual review of statistics and its application*, 6:309–328.

Guo, F. and Fang, Y. (2013). Individual driver risk assessment using naturalistic driving data. *Accident Analysis & Prevention*, 61:3–9.

Guo, F., Fang, Y., and Antin, J. F. (2015). Older driver fitness-to-drive evaluation using naturalistic driving data. *Journal of safety research*, 54:49–e29.

Guo, F., Klauer, S. G., Hankey, J. M., and Dingus, T. A. (2010). Near crashes as crash surrogate for naturalistic driving studies. *Transportation Research Record*, 2147(1):66–74.

Han, C., Huang, H., Lee, J., and Wang, J. (2018). Investigating varying effect of road-level factors on crash frequency across regions: a bayesian hierarchical random parameter modeling approach. *Analytic methods in accident research*, 20:81–91.

Hastings, W. K. (1970). Monte carlo sampling methods using markov chains and their applications. *Biometrika*.

Hickman, J. S., Hanowski, R. J., and Bocanegra, J. (2018). A synthetic approach to compare the large truck crash causation study and naturalistic driving data. *Accident Analysis & Prevention*, 112:11–14.

Huang, H. and Abdel-Aty, M. (2010). Multilevel data and bayesian analysis in traffic safety. *Accident Analysis & Prevention*, 42(6):1556–1565.

Huang, Y.-h., Zohar, D., Robertson, M. M., Garabet, A., Lee, J., and Murphy, L. A.

(2013). Development and validation of safety climate scales for lone workers using truck drivers as exemplar. *Transportation Research Part F: Traffic Psychology and Behaviour*, 17:5–19.

Hydén, C. (1987). The development of a method for traffic safety evaluation: The swedish traffic conflicts technique.

im Kampe, E. O., Kovats, S., and Hajat, S. (2016). Impact of high ambient temperature on unintentional injuries in high-income countries: a narrative systematic literature review. *BMJ open*, 6(2):e010399.

Islam, S., Jones, S. L., and Dye, D. (2014). Comprehensive analysis of single-and multi-vehicle large truck at-fault crashes on rural and urban roadways in alabama. *Accident Analysis & Prevention*, 67:148–158.

Janakiraman, V. M., Matthews, B., and Oza, N. (2016). Discovery of precursors to adverse events using time series data. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pages 639–647. SIAM.

Jansen, R. J. and Simone Wesseling, S. (2018). Harsh braking by truck drivers: A comparison of thresholds and driving contexts using naturalistic driving data. In *Proceedings of the 6th Humanist Conference, available at* https://bit.ly/2C2Bw3Z *[Online]*.

Jovanis, P. P., Wu, K.-F., and Chen, C. (2011). Hours of service and driver fatigue: Driver characteristics research. Technical report, U.S. Department of Transportation, Federal Motor Carrier Safety Administration.

Kamla, J., Parry, T., and Dawson, A. (2019). Analysing truck harsh braking incidents to study roundabout accident risk. *Accident Analysis & Prevention*, 122:365–377.

Kjellstrom, T., Kovats, R. S., Lloyd, S. J., Holt, T., and Tol, R. S. (2009). The direct impact of climate change on regional labor productivity. *Archives of Environmental & Occupational Health*, 64(4):217–227.

Klauer, S. G., Dingus, T. A., Neale, V. L., Sudweeks, J. D., and Ramsey, D. J. (2009). Comparing real-world behaviors of drivers with high versus low rates of crashes and near crashes. Technical report, U.S. Department of Transportation, Federal Motor Carrier Safety Administration.

Knipling, R. R. (2017). Threats to scientific validity in truck driver hours-of-service studies. *PROCEEDINGS of the Eighth International Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*.

Kononov, J., Bailey, B., and Allery, B. K. (2008). Relationships between safety and both congestion and number of lanes on urban freeways. *Transportation research record*, 2083(1):26–39.

Korattikara, A., Chen, Y., and Welling, M. (2014). Austerity in mcmc land: Cutting the metropolis-hastings budget. In *International Conference on Machine Learning*, pages 181–189.

Kruschke, J. (2014). *Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan*. Academic Press.

Kruschke, J. K. and Vanpaemel, W. (2015). Bayesian estimation in hierarchical models. *The Oxford handbook of computational and mathematical psychology*, pages 279–299.

Lambert, B. (2018). *A student' s guide to Bayesian statistics*. Sage.

Leard, B., Roth, K., et al. (2015). Weather, traffic accidents, and climate change. *Re-*

sources for the Future Discussion Paper, pages 15–19.

Lemp, J. D., Kockelman, K. M., and Unnikrishnan, A. (2011). Analysis of large truck crash severity using heteroskedastic ordered probit models. *Accident Analysis & Prevention*, 43(1):370–380.

Litman, T. (2013). Transportation and public health. *Annual review of public health*, 34:217–233.

Liu, Y., Guo, F., and Hanowski, R. J. (2019). Assessing the impact of sleep time on truck driver performance using a recurrent event model. *Statistics in medicine*.

Lord, D. (2006). Modeling motor vehicle crashes using poisson-gamma models: Examining the effects of low sample mean values and small sample size on the estimation of the fixed dispersion parameter. *Accident Analysis & Prevention*, 38(4):751–766.

Lord, D. and Mannering, F. (2010). The statistical analysis of crash-frequency data: a review and assessment of methodological alternatives. *Transportation research part A: policy and practice*, 44(5):291–305.

Lord, D., Washington, S., and Ivan, J. N. (2007). Further notes on the application of zero-inflated models in highway safety. *Accident Analysis & Prevention*, 39(1):53–57.

Lord, D., Washington, S. P., and Ivan, J. N. (2005). Poisson, poisson-gamma and zero-inflated regression models of motor vehicle crashes: balancing statistical fit and theory. *Accident Analysis & Prevention*, 37(1):35–46.

Maclaurin, D. and Adams, R. P. (2015). Firefly monte carlo: Exact mcmc with subsets of data. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

McElreath, R. (2018). *Statistical rethinking: A Bayesian course with examples in R and Stan*. Chapman and Hall/CRC.

McElreath, R. (2020). *Statistical rethinking: A Bayesian course with examples in R and Stan*. CRC press.

Mehdizadeh, A., Cai, M., Hu, Q., Alamdar Yazdi, M. A., Mohabbati-Kalejahi, N., Vinel, A., Rigdon, S. E., Davis, K. C., and Megahed, F. M. (2020). A review of data analytic applications in road traffic safety. part 1: descriptive and predictive modeling. *Sensors*, 20(4):1107.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092.

Meuleners, L., Fraser, M. L., Govorko, M. H., and Stevenson, M. R. (2015). Obstructive sleep apnea, health-related factors, and long distance heavy vehicle crashes in western australia: a case control study. *Journal of Clinical Sleep Medicine*, 11(04):413–418.

Mitler, M. M., Miller, J. C., Lipsitz, J. J., Walsh, J. K., and Wylie, C. D. (1997). The sleep of long-haul truck drivers. *New England Journal of Medicine*, 337(11):755–762.

Mohammadi, M. A., Samaranayake, V., and Bham, G. H. (2014). Crash frequency modeling using negative binomial models: An application of generalized estimating equation to longitudinal data. *Analytic Methods in Accident Research*, 2:52–69.

Mollicone, D., Kan, K., Mott, C., Bartels, R., Bruneau, S., van Wollen, M., Sparrow, A. R., and Van Dongen, H. P. (2019). Predicting performance and safety based on driver fatigue. *Accident Analysis & Prevention*, 126:142–145.

Moneta, G. B., Leclerc, A., Chastang, J.-F., Tran, P. D., and Goldberg, M. (1996). Time-trend of sleep disorder in relation to night work: a study of sequential 1-year prevalences within the gazel cohort. *Journal of clinical epidemiology*, 49(10):1133–1141.

Naik, B., Tung, L.-W., Zhao, S., and Khattak, A. J. (2016). Weather impacts on single-vehicle truck crash injury severity. *Journal of safety research*, 58:57–65.

Nakayama, H. (2002). Trial measurments of driver fatigue in extended driving condition. In *9th World Congress on Intelligent Transport SystemsITS America, ITS Japan, ERTICO (Intelligent Transport Systems and Services-Europe)*.

National Sleep Foundation (2008). 2008 State of the States Report on Drowsy Driving. http://drowsydriving.org/resources/2008-state-of-the-states-report-on-drowsy-driving/. [Online; accessed 20-February-2019].

National Transportation Safety Board (1990). Safety study: Fatigue, alcohol, other drugs, and medical factors in fatal-to-the-driver heavy truck crashes.

Neale, V. L., Dingus, T. A., Klauer, S. G., Sudweeks, J., and Goodman, M. (2005). An overview of the 100-car naturalistic study and findings. *National Highway Traffic Safety Administration, Paper*, 5:0400.

Née, M., Contrand, B., Orriols, L., Gil-Jardiné, C., Galéra, C., and Lagarde, E. (2019). Road safety and distraction, results from a responsibility case-control study among a sample of road users interviewed at the emergency room. *Accident Analysis & Prevention*, 122:19–24.

Neeley, G. W. and Richardson Jr, L. E. (2009). The effect of state regulations on truck-crash fatalities. *American journal of public health*, 99(3):408–415.

Noland, R. B. and Oh, L. (2004). The effect of infrastructure and demographic change on traffic-related fatalities and crashes: a case study of illinois county-level data. *Accident Analysis & Prevention*, 36(4):525–532.

Olson, R., Wipfli, B., Thompson, S. V., Elliot, D. L., Anger, W. K., Bodner, T., Hammer, L. B., and Perrin, N. A. (2016). Weight control intervention for truck drivers: The shift randomized controlled trial, united states. *American journal of public health*, 106(9):1698–1706.

Otmani, S., Rogé, J., and Muzet, A. (2005). Sleepiness in professional drivers: effect of age and time of day. *Accident Analysis & Prevention*, 37(5):930–937.

Pack, A. I., Pack, A. M., Rodgman, E., Cucchiara, A., Dinges, D. F., and Schwab, C. W. (1995). Characteristics of crashes attributed to the driver having fallen asleep. *Accident Analysis & Prevention*, 27(6):769–775.

Pahukula, J., Hernandez, S., and Unnikrishnan, A. (2015). A time of day analysis of crashes involving large trucks in urban areas. *Accident Analysis & Prevention*, 75:155–163.

Pande, A., Chand, S., Saxena, N., Dixit, V., Loy, J., Wolshon, B., and Kent, J. D. (2017). A preliminary investigation of the relationships between historical crash and naturalistic driving. *Accident Analysis & Prevention*, 101:107–116.

Pantangi, S. S., Fountas, G., Sarwar, M. T., Anastasopoulos, P. C., Blatt, A., Majka, K., Pierowicz, J., and Mohan, S. B. (2019). A preliminary investigation of the effectiveness of high visibility enforcement programs using naturalistic driving study data: A grouped random parameters approach. *Analytic Methods in Accident Research*, 21:1–12.

Peden, M., Scurfield, R., Sleet, D., Mohan, D., Hyder, A. A., Jarawan, E., Mathers, C. D., et al. (2004). World report on road traffic injury prevention.

Pylkkönen, M., Sihvola, M., Hyvärinen, H., Puttonen, S., Hublin, C., and Sallinen, M. (2015). Sleepiness, sleep, and use of sleepiness countermeasures in shift-working long-haul truck drivers. *Accident Analysis & Prevention*, 80:201–210.

Quiroz, M., Kohn, R., Villani, M., and Tran, M.-N. (2019). Speeding up mcmc by efficient data subsampling. *Journal of the American Statistical Association*, 114(526):831–843.

Quiroz, M., Tran, M.-N., Villani, M., and Kohn, R. (2018a). Speeding up mcmc by delayed acceptance and data subsampling. *Journal of Computational and Graphical Statistics*, 27(1):12–22.

Quiroz, M., Tran, M.-N., Villani, M., Kohn, R., and Dang, K.-D. (2016). The block-poisson estimator for optimally tuned exact subsampling mcmc. *arXiv preprint arXiv:1603.08232*.

Quiroz, M., Villani, M., Kohn, R., Tran, M.-N., and Dang, K.-D. (2018b). Subsampling mcmc-an introduction for the survey statistician. *Sankhya A*, pages 1–37.

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Revelle, W., Condon, D. M., Wilt, J., French, J. A., Brown, A., and Elleman, L. G. (2016). *Web and phone based data collection using planned missing designs*, pages 578–595. Sage Publications, Inc.

Revelle, W., Wilt, J., and Rosenthal, A. (2010). *Individual Differences in Cognition: New Methods for Examining the Personality-Cognition Link*, pages 27–49. Springer New York, New York, NY.

Rifaat, S. M., Tay, R., and De Barros, A. (2012). Severity of motorcycle crashes in calgary. *Accident Analysis & Prevention*, 49:44–49.

Rigdon, S. E. and Basu, A. P. (1989). The power law process: a model for the reliability of repairable systems. *Journal of Quality Technology*, 21(4):251–260.

Rigdon, S. E. and Basu, A. P. (2000). *Statistical methods for the reliability of repairable systems*. Wiley New York.

Risser, R. (1985). Behavior in traffic conflict situations. *Accident Analysis & Prevention*, 17(2):179–197.

Roshandel, S., Zheng, Z., and Washington, S. (2015). Impact of real-time traffic characteristics on freeway crash occurrence: Systematic review and meta-analysis. *Accident Analysis & Prevention*, 79:198–211.

Rudis, B. (2018). darksky: An R interface to the Dark Sky API.

Saleh, J. H., Saltmarsh, E. A., Favaro, F. M., and Brevault, L. (2013). Accident precursors, near misses, and warning signs: critical review and formal definitions within the framework of discrete event systems. *Reliability Engineering & System Safety*, 114:148–154.

Sallinen, M., HÄRMÄ, M., Mutanen, P., RANTA, R., Virkkala, J., and MÜLLER, K. (2005). Sleepiness in various shift combinations of irregular shift systems. *Industrial health*, 43(1):114–122.

Savolainen, P. T., Mannering, F. L., Lord, D., and Quddus, M. A. (2011). The statistical analysis of highway crash-injury severities: a review and assessment of methodologi-

cal alternatives. *Accident Analysis & Prevention*, 43(5):1666–1676.
Sedgwick, P. (2014). Case-control studies: advantages and disadvantages. *Bmj*, 348:f7707.
Shmueli, G. et al. (2010). To explain or to predict? *Statistical science*, 25(3):289–310.
Simons-Morton, B. G., Zhang, Z., Jackson, J. C., and Albert, P. S. (2012). Do elevated gravitational-force events while driving predict crashes and near crashes? *American journal of epidemiology*, 175(10):1075–1079.
Soccolich, S. A., Blanco, M., Hanowski, R. J., Olson, R. L., Morgan, J. F., Guo, F., and Wu, S.-C. (2013). An analysis of driving and working hour on commercial motor vehicle driver safety using naturalistic data collection. *Accident Analysis & Prevention*, 58:249–258.
Solomon, A. J., Doucette, J. T., Garland, E., and McGinn, T. (2004). Healthcare and the long haul: long distance truck drivers—a medically underserved population. *American journal of industrial medicine*, 46(5):463–471.
Sparrow, A. R., Mollicone, D. J., Kan, K., Bartels, R., Satterfield, B. C., Riedy, S. M., Unice, A., and Van Dongen, H. P. (2016). Naturalistic field study of the restart break in us commercial motor vehicle drivers: Truck driving, sleep, and fatigue. *Accident Analysis & Prevention*, 93:55–64.
Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(4):583–639.
Stan Development Team (2018a). RStan: the R interface to Stan. R package version 2.18.2.
Stan Development Team (2018b). RStan: the R interface to Stan. R package version 2.18.2.
Staton, C., Vissoci, J., Gong, E., Toomey, N., Wafula, R., Abdelgadir, J., Zhou, Y., Liu, C., Pei, F., Zick, B., et al. (2016). Road traffic injury prevention initiatives: a systematic review and metasummary of effectiveness in low and middle income countries. *PLoS One*, 11(1):e0144971.
Stern, H. S., Blower, D., Cohen, M. L., Czeisler, C. A., Dinges, D. F., Greenhouse, J. B., Guo, F., Hanowski, R. J., Hartenbaum, N. P., Krueger, G. P., et al. (2019). Data and methods for studying commercial motor vehicle driver fatigue, highway safety and long-term driver health. *Accident Analysis & Prevention*, 126:37–42.
The Dark Sky API (2019a). Dark sky api —overview. [Online; accessed 20-June-2019].
The Dark Sky API (2019b). Data sources. [Online; accessed 20-June-2019].
The National Safety Council (2018). Vehicle deaths estimated at 40,000 for third straight year. https://www.nsc.org/road-safety/safety-topics/fatality-estimates. [Online; accessed 20-February-2019].
The United States, Bureau of Labor Statistics (2017). Fatal occupational injuries by event, 2017. https://www.bls.gov/charts/census-of-fatal-occupational-injuries/fatal-occupational-injuries-by-event-drilldown.htm. [Online; accessed 20-February-2019].
Theofilatos, A. and Yannis, G. (2014). A review of the effect of traffic and weather characteristics on road safety. *Accident Analysis & Prevention*, 72:244–256.
Theofilatos, A., Yannis, G., Kopelias, P., and Papadimitriou, F. (2016). Predicting road accidents: a rare-events modeling approach. *Transportation research procedia*,

14:3399–3405.

Theofilatos, A., Yannis, G., Kopelias, P., and Papadimitriou, F. (2018). Impact of real-time traffic characteristics on crash occurrence: Preliminary results of the case of rare events. *Accident Analysis & Prevention*.

Tsai, Y.-T., Alhwiti, T., Swartz, S. M., and Megahed, F. M. (2015). The effects of socio-economic and public policy factors on us highway safety. *Journal of Transportation Law, Logistics, and Policy*, 82(1/2):31–48.

Van Ravenzwaaij, D., Cassey, P., and Brown, S. D. (2018). A simple introduction to markov chain monte–carlo sampling. *Psychonomic bulletin & review*, 25(1):143–154.

Vehtari, A., Gelman, A., and Gabry, J. (2017). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27(5):1413–1432.

Vehtari, A., Simpson, D., Gelman, A., Yao, Y., and Gabry, J. (2015). Pareto smoothed importance sampling.

Wang, C., Quddus, M. A., and Ison, S. G. (2013). The effect of traffic and road characteristics on road safety: A review and future research direction. *Safety science*, 57:264–275.

Wang, L., Abdel-Aty, M., and Lee, J. (2017). Safety analytics for integrating crash frequency and real-time risk modeling for expressways. *Accident Analysis & Prevention*, 104:58–64.

Washington, S. P., Karlaftis, M. G., and Mannering, F. (2010). *Statistical and econometric methods for transportation data analysis*. Chapman and Hall/CRC.

Watanabe, S. (2010). Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, 11(Dec):3571–3594.

WHO (2018a). Road traffic injuries. http://www.who.int/mediacentre/factsheets/fs358/en/. [Online; accessed 28-August-2018].

WHO (2018b). The top 10 causes of death. http://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death. [Online; accessed 20-February-2019].

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.

Wickham, H., François, R., Henry, L., and Müller, K. (2019). *dplyr: A Grammar of Data Manipulation*. R package version 0.8.3.

Wu, K.-F., Aguero-Valverde, J., and Jovanis, P. P. (2014). Using naturalistic driving data to explore the association between traffic safety-related events and crash risk at driver level. *Accident Analysis & Prevention*, 72:210–218.

Wu, K.-F. and Jovanis, P. P. (2012). Crashes and crash-surrogate events: Exploratory modeling with naturalistic driving data. *Accident Analysis & Prevention*, 45:507–516.

Xie, Y., Zhang, Y., and Liang, F. (2009). Crash injury severity analysis using bayesian ordered probit models. *Journal of Transportation Engineering*, 135(1):18–25.

Xu, C., Wang, W., Liu, P., and Li, Z. (2015). Calibration of crash risk models on freeways with limited real-time traffic data using bayesian meta-analysis and bayesian inference approach. *Accident Analysis & Prevention*, 85:207–218.

Ye, F. and Lord, D. (2011). Investigation of effects of underreporting crash data on three commonly used traffic crash severity models: multinomial logit, ordered probit, and mixed logit. *Transportation Research Record*, 2241(1):51–58.

Yu, R. and Abdel-Aty, M. (2014). Using hierarchical bayesian binary probit models to analyze crash injury severity on high speed facilities with real-time traffic data. *Accident Analysis & Prevention*, 62:161–167.

Yung, M. (2016). *Fatigue at the workplace: Measurement and temporal development*. PhD thesis, University of Waterloo.

Zaloshnja, E., Miller, T., et al. (2008). Unit costs of medium and heavy truck crashes. Technical report, The United States. Federal Motor Carrier Safety Administration.

Zhu, X. and Srinivasan, S. (2011). A comprehensive analysis of factors influencing the injury severity of large-truck crashes. *Accident Analysis & Prevention*, 43(1):49–57.

VITA AUCTORIS

Miao Cai was born and raised in Xinzhou district, Wuhan, Hubei Province, China.