U.S. Department of Transportation
National Highway Traffic Safety Administration

NHTSA

DOT HS 813 225

December 2021

Crash Report Sampling System: Composite Estimator Variance Estimation

Disclaimer

This publication is distributed by the U.S. Department of Transportation, National Highway Traffic Safety Administration, in the interest of information exchange. The opinions, findings, and conclusions expressed in this publication are those of the authors and not necessarily those of the Department of Transportation or the National Highway Traffic Safety Administration. The United States Government assumes no liability for its contents or use thereof. If trade or manufacturers' names or products are mentioned, it is because they are considered essential to the object of the publication and should not be construed as an endorsement. The United States Government does not endorse products or manufacturers.

Suggested APA Format Citation:

Zhang, F., Noh, E. Y., & Boyle, L. (2021, December). *Crash Report Sampling System: Composite estimator variance estimation* (Report No. DOT HS 813 225). National Highway Traffic Safety Administration.

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
DOT HS 813 225		

4. Title and Subtitle	5. Report Date
Crash Report Sampling System: Composite Estimator Variance Estimation	December 2021
	6. Performing Organization Code NSA-210
7. Authors Fan Zhang, Eun Young Noh, Lacey Boyle	8. Performing Organization Report No.
9. Performing Organization Name Mathematical Analysis Division National Center for Statistics and Analysis National Highway Traffic Safety Administration 1200 New Jersey Avenue SE Washington, DC 20590	10. Work Unit No. (TRAIS) 11. Contract or Grant No.
12. Sponsoring Agency Name and Address Mathematical Analysis Division, National Center for Statistics and Analysis National Highway Traffic Safety Administration 1200 New Jersey Avenue SE Washington, DC 20590	13. Type of Report and Period Covered NHTSA Technical Report 14. Sponsoring Agency Code
15. Supplementary Notes	

Abstract

This technical report discusses variance estimation methods for the CRSS/FARS composite estimates. In NHTSA's publications, non-fatal estimates are often made using a composite estimator that combines FARS estimates with CRSS non-fatal crash estimates. The standard error estimation of the composite estimator becomes complicated when the underlying composite estimator is a non-linear function of the FARS estimates and the CRSS estimates. In this report, we proposed and justified a variance estimation method for the CRSS/FARS composite estimators and provided example programs in SAS, SUDAAN, and R software.

17. Key Words	18. Distribution Statement
NHTSA, CRSS, FARS, composite estimator, variance estimation.	Document is available to the public from the DOT, BTS, National Transportation Library, Repository & Open Science Access Portal, rosap.ntl.bts.gov.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	60	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

Acronyms

CRSS – Crash Report Sampling System – a replacement of GES

FARS – Fatality Analysis Reporting System

GES – General Estimates System

GVF – generalized variance function

PCR – police crash report

PJ – police jurisdiction

PSU – primary sampling unit

SAS – Statistical Analysis System (a suite of analytics software)

R - language and environment for statistical computing and graphics

SE – standard error

SRSWOR – simple random sampling without replacement

SUDAAN – SUrvey DAta ANalysis (a proprietary statistical software package for the analysis of correlated data, including complex sample survey data)

WOR – without replacement

WR – with replacement

Table of Contents

1. Introduction

In this technical report, we first discuss existing standard error estimation methods for the composite estimator using the FARS and the CRSS data. We then propose a different method for the standard error estimation of general composite estimators using the FARS and the CRSS data.

NHTSA's CRSS is a national annual probability sample of all police reported traffic crashes, including fatal and non-fatal crashes. NHTSA's FARS is a national annual census of fatal motor vehicle traffic crashes. CRSS data can be used to make fatal and non-fatal estimates. But the CRSS fatal estimates are subject to sampling error because they are made from a sample while FARS counts are not subject to sampling error because they are made from a census. Zhang et al. (2019, April) provided example computer programs using the standalone CRSS data in conjunction with SAS and SUDAAN.

NHTSA makes fatal domain (sub-population) estimates from the FARS. Non-fatal total estimates are often made by combining the non-fatal domain counts from the FARS crashes and the non-fatal domain estimates from the CRSS/GES non-fatal crashes. For example, the total number of injured people can be estimated by the following composite estimator:

$$\hat{t}_C = \hat{t}_{FARS} + \hat{t}_{CRSS\ Non-fatal}$$

here $\hat{t}_C$ is the estimated total number of injured people, $\hat{t}_{FARS}$ is the total number of injured people calculated from the FARS crashes, $\hat{t}_{CRSS\ Non-fatal}$ is the total number of injured people estimated from the CRSS non-fatal crashes. Since the FARS and the CRSS are independent surveys and the FARS is a census, the variance associated with $\hat{t}_C$ is:

$$Var(\hat{t}_C) = Var(\hat{t}_{FARS}) + Var\left(\hat{t}_{CRSS\ Non-fatal}\right) = Var\left(\hat{t}_{CRSS\ Non-fatal}\right)$$

On the other hand, we can also use all sampled CRSS crashes (fatal or non-fatal) to estimate the same population total. Let $\hat{t}_{CRSS} = \hat{t}_{CRSS\ fatal} + \hat{t}_{CRSS\ Non-fatal}$ be the corresponding total estimator using the CRSS fatal crashes ($\hat{t}_{CRSS\ fatal}$) and the CRSS non-fatal crashes ($\hat{t}_{CRSS\ Non-fatal}$). Thus,

$$Var(\hat{t}_{CRSS}) = Var\left(\hat{t}_{CRSS\ fatal}\right) + Var\left(\hat{t}_{CRSS\ Non-fatal}\right) + 2Cov\left(\hat{t}_{CRSS\ fatal}, \hat{t}_{CRSS\ Non-fatal}\right)$$

Notice $\hat{t}_{CRSS\ fatal}$ and $\hat{t}_{CRSS\ Non-fatal}$ are defined on the same PSU and PJ samples. For the same PSU and PJ, the number of fatal crashes is positively correlated with the number of injury crashes. Comparing the right-hand sides of $Var(\hat{t}_C)$ and $Var(\hat{t}_{CRSS})$ we have:

$$Var(\hat{t}_C) \leq Var(\hat{t}_{CRSS})$$

Therefore, the composite estimator produces better injury related total estimates because it uses the FARS census fatal crashes instead of the sampled fatal crashes in the CRSS.

Example 1: Variance Estimation of Linear Composite Estimator

In this example we estimate the total number of people in motor vehicles in transport (PERMVIT) in 2018. We first sum up FARS variable PERMVIT over all 2018 FARS crashes to get the FARS count: $\hat{t}_{FARS}$ = 76,036 with zero sampling variance. CRSS estimate is the weighted sum of variable PERMVIT over all 2018 CRSS non-fatal crashes: $\hat{t}_{CRSS\ Non-fatal}$ = 15,924,248 with estimated standard error 857,870. The composite estimator is a linear combination:

$$\hat{t}_C = \hat{t}_{FARS} + \hat{t}_{CRSS\ Non-fatal} = 76{,}036 + 15{,}924{,}248 = 16{,}000{,}284$$

$$ste(\hat{t}_C) = ste(\hat{t}_{CRSS\ Non-fatal}) = 857{,}870.$$

SAS and R programs for Example 1 can be found in Appendix A and B. The following are excerpts from the SAS output.

Table 1. SAS output for Example 1

FARS Count				
Analysis Variable: PERMVIT Number of People in Motor Vehicles In-Transport				
Sum				
76,036.00				
CRSS Estimate				
Statistics for FATAL_FLAG Domains				
FATAL_FLAG	**Variable**	**Label**	**Sum**	**Std Error of Sum**
0	PERMVIT	Number of People in Motor Vehicles In-Transport	15,924,248	857,870
1	PERMVIT	Number of People in Motor Vehicles In-Transport	72,984	5,266.252678

However, this approach is only applicable to the linear composite estimator of FARS counts and CRSS estimates. For non-linear composite estimator, such as percentage, ratio, or regression estimators, the above variance equation is not correct.

In general, assume a population parameter θ is a non-linear function of K population totals:

$$\theta = f(t_1, \ldots t_K),$$

here $f(t_1, \ldots t_K)$ is a non-linear function of K population totals: $t_1, \ldots t_K$. Examples of a finite population parameter as a non-linear function of population totals include percentage, ratio, least square regression coefficients, etc. The corresponding composite estimator of θ is:

$$\hat{\theta}_C = f(\hat{t}_{c1}, \ldots \hat{t}_{cK}),$$

Here $\hat{t}_{ci} = \hat{t}_{i,FARS} + \hat{t}_{i,CRSS\ Non-fatal},\ (i = 1,2, \dots K)$ is estimated from the FARS crashes ($\hat{t}_{i,FARS}$) and the CRSS non-fatal crashes ($\hat{t}_{i,CRSS\ Non-fatal}$). FARS count $\hat{t}_{i,FARS}$ and CRSS estimate $\hat{t}_{i,CRSS\ Non-fatal}$ can be written by:

$$\hat{t}_{i,FARS} = \sum_{j \in U_{FARS}} y_{ij}$$

$$\hat{t}_{i,CRSS\ Non-fatal} = \sum_{j \in S_{CRSS\ Non-fatal}} w_j y_{ij}.$$

Here U_{FARS} is the set of fatal crashes in the FARS file, y_{ij} is the jth record of study variable i, $S_{CRSS\ Non-fatal}$ is the set of sampled non-fatal crashes in the CRSS, w_j is the CRSS analysis weight.

For example, the percentage estimator of category g among total G categories can be viewed as the ratio of two composite total estimators ($\hat{t}_C^g$ and $\hat{t}_C$):

$$\hat{\theta}_C^g = \frac{\hat{t}_{FARS}^g + \hat{t}_{CRSS\ Non-fatal}^g}{\sum_{g=1}^G \hat{t}_{FARS}^g + \sum_{g=1}^G \hat{t}_{CRSS\ Non-fatal}^g} = \frac{\hat{t}_{FARS}^g + \hat{t}_{CRSS\ Non-fatal}^g}{\hat{t}_{FARS} + \hat{t}_{CRSS\ Non-fatal}} = \frac{\hat{t}_C^g}{\hat{t}_C}$$

The variance of $\hat{\theta}_C^g$ can no longer be easily estimated by the variance estimate of a single component.

In this report we first briefly describe two methods often used to approximate the variance of a non-linear composite estimate $\hat{\theta}_C = f(\hat{t}_{c1}, \dots \hat{t}_{cK})$: approximation by substitution (Section 2) and approximation by GVF (Section 3). We then propose and justify another estimation method using the concatenated FARS and CRSS data set and provide detailed examples in SAS (Section 4), SAS-callable SUDAAN (Section 4), and R (Appendix B). Pros and cons of all methods are summarized in Section 5.

All examples in this report use 2018 CRSS and FARS crash level data except example 7 where both 2018 and 2019 CRSS and FARS crash level data are used for multi-year comparison.

2. Approximation by Substitution

We first consider how Taylor series method estimates the variance of a non-linear estimator. For the simplicity of presentation, we consider the non-linear function of K=2 totals: $\hat{\theta} = f(\hat{t}_1, \hat{t}_2)$. Notice by Taylor series:

$$\hat{\theta} \approx f(t_1, t_2) + f'_{t_1}(t_1, t_2)(\hat{t}_1 - t_1) + f'_{t_2}(t_1, t_2)(\hat{t}_2 - t_2)$$

$$= f(t_1, t_2) - f'_{t_1}(t_1, t_2)t_1 - f'_{t_2}(t_1, t_2)t_2 + f'_{t_1}(t_1, t_2)\hat{t}_1 + f'_{t_2}(t_1, t_2)\hat{t}_2$$

here $t_1 = E(\hat{t}_1)$ and $t_2 = E(\hat{t}_2)$ are two population totals therefore are constants, $f'_{t_1}(t_1, t_2)$ and $f'_{t_2}(t_1, t_2)$ are the first order derivatives of function f evaluated at (t_1, t_2). Notice the first three terms in the last equation are constants. Therefore,

$$Var(\hat{\theta}) \approx Var\left[f'_{t_1}(t_1, t_2)\hat{t}_1 + f'_{t_2}(t_1, t_2)\hat{t}_2\right] \quad (1)$$

Notice now the statistic $f'_{t_1}(t_1, t_2)\hat{t}_1 + f'_{t_2}(t_1, t_2)\hat{t}_2$ on the right-hand side is a linear estimator. Therefore, $Var\left[f'_{t_1}(t_1, t_2)\hat{t}_1 + f'_{t_2}(t_1, t_2)\hat{t}_2\right]$ can be estimated using textbook formulae. To calculate the variance estimate, the unknown t_1 and t_2 are substituted by their point estimates: $\hat{t}_1$ and $\hat{t}_2$. This is how the survey procedures calculate the estimate of $Var(\hat{\theta})$ using the Taylor series method. See, for example, RTI, I., & Bieler, G. (2008) page 53, SAS Institute Inc. (2017) page 9465, and Lumley (2004) page 4.

For composite estimator $\hat{\theta}_C = f(\hat{t}_{c1}, \hat{t}_{c2})$ and $\hat{t}_{ci} = \hat{t}_{i,FARS} + \hat{t}_{i,CRSS\ Non-fatal}$, $(i = 1,2)$, let $\hat{t}_{i,CRSS} = \hat{t}_{i,CRSS\ Fatal} + \hat{t}_{i,CRSS\ Non-fatal}$ be the CRSS total estimate estimated from the CRSS fatal and non-fatal crashes. When the sample size is large, the CRSS total estimate $\hat{t}_{i,CRSS} \approx \hat{t}_{ci} \approx t_i$. In addition, when $\hat{t}_{i,FARS} \ll \hat{t}_{i,CRSS\ Non-fatal}$, the variance inflation is small when $\hat{t}_{ci} = \hat{t}_{i,FARS} + \hat{t}_{i,CRSS\ Non-fatal}$ is replaced by $\hat{t}_{i,CRSS} = \hat{t}_{i,CRSS\ Fatal} + \hat{t}_{i,CRSS\ Non-fatal}$. Under these conditions, equation (1) becomes:

$$Var(\hat{\theta}_C) \approx Var\left[f'_{t_1}(\hat{t}_{1,CRSS}, \hat{t}_{2,CRSS})\hat{t}_{1,CRSS} + f'_{t_2}(\hat{t}_{1,CRSS}, \hat{t}_{2,CRSS})\hat{t}_{2,CRSS}\right] \quad (2)$$

The right-hand side of equation (2) is how the survey procedures estimate the variance of $\hat{\theta}_{CRSS}$ using Taylor series method, here $\hat{\theta}_{CRSS}$ has the same function form as $\hat{\theta}_C$ except $\hat{\theta}_{CRSS}$ uses only the CRSS crashes (fatal and non-fatal). This leads to the first approximation by substitution method: use the all-CRSS-crash estimate of $Var(\hat{\theta}_{CRSS})$ as the estimate of $Var(\hat{\theta}_C)$.

Example 2: Use the Variance Estimate of the All-CRSS-Crash Estimate

In this example we estimate the percentage of fatal or injury crashes among all 2018 CRSS in-scope crashes. Since all CRSS sampled cases are used and the fatal subpopulation is only about 0.5 percent of the CRSS population, we use the all-CRSS-crash variance estimate $var(\hat{\theta}_{CRSS})$ to approximate $var(\hat{\theta}_C)$.

Table 2. SAS Outputs for Example 2

Table of INJURED					
INJURED	**Frequency**	**Weighted Frequency**	**Std Err of Wgt Freq**	**Percent**	**Std Err of Percent**
Injured or Fatal	23927	1927358	88421	28.6195	0.7507
No Injury	24516	4807058	262531	71.3805	0.7507
Total	48443	6734416	333709	100.0000	

SAS program for this example can be found in the Appendix A. Table 2 is the excerpt of the SAS output. The percentage of fatal or injury crashes estimated from the CRSS is: $\hat{\theta}_{CRSS} \approx 29\%$. We use the standard error of $\hat{\theta}_{CRSS}$ to approximate the standard error of $\hat{\theta}_C$:

$$ste(\hat{\theta}_C) \approx ste(\hat{\theta}_{CRSS}) = 0.75\%$$

In Example 5 below, we estimated the actual $\hat{\theta}_C$ using the exact composite estimator and its associated standard error (Table 6). It turns out $\hat{\theta}_C$ =28.62% and $ste(\hat{\theta}_C) = 0.75$, same as $\hat{\theta}_{CRSS}$ and $ste(\hat{\theta}_{CRSS})$. This is because the CRSS fatal crash estimate was calibrated to the FARS count: $\hat{t}_{CRSS\,Fatal} \approx \hat{t}_{FARS}$ (Section 9.5 in Zhang, Noh, et al., 2019). Therefore, for this example, $\hat{\theta}_C = \hat{\theta}_{CRSS}$. However, it should also be noticed that the calibration is normally performed before the FARS publishes its final files. NHTSA may re-calibrate the CRSS fatal estimates if the FARS final counts are significantly different from the ones used earlier for the CRSS calibration.

The second method of approximation by substitution is using $var(\hat{\theta}_{CRSS\,Non-fatal})$ as the estimate of $Var(\hat{\theta}_C)$. To this end, notice for composite estimator $\hat{\theta}_C = f(\hat{t}_{c1}, \hat{t}_{c2})$ and $\hat{t}_{ci} = \hat{t}_{i,FARS} + \hat{t}_{i,CRSS\,Non-fatal}$, $(i = 1,2)$, when sample size is large so $\hat{t}_{ci} \approx t_i$ and when $\hat{t}_{i,FARS} \ll \hat{t}_{i,CRSS\,Non-fatal}$ so $\hat{t}_{ci} \approx \hat{t}_{i,CRSS\,Non-fatal}$, $(i = 1,2)$, the equation (1) becomes:

$$Var(\hat{\theta}_C) \approx Var\left[f'_{t_1}(\hat{t}_{1,CRSS\,Non-fatal}, \hat{t}_{2,CRSS\,Non-fatal})\hat{t}_{1,CRSS\,Non-fatal} + f'_{t_2}(\hat{t}_{1,CRSS\,Non-fatal}, \hat{t}_{2,CRSS\,Non-fatal})\hat{t}_{2,CRSS\,Non-fatal}\right] \quad (3)$$

The right-hand side is how the survey procedures estimate the variance of $\hat{\theta}_{CRSS\,Non-fatal}$ using Taylor series method, here $\hat{\theta}_{CRSS\,Non-fatal}$ has the same function form except it is estimated from the CRSS non-fatal cases. This leads to the second approximation by substitution method: using the variance estimate of the CRSS non-fatal estimate, $Var(\hat{\theta}_{CRSS\,Non-fatal})$, to estimate $Var(\hat{\theta}_C)$.

Example 3: Use the Variance Estimate of the CRSS Non-Fatal Crash Estimate

In this example, we still use $\hat{\theta}_C \approx \hat{\theta}_{CRSS} \approx 29\%$ to estimate the percentage of fatal or injury crashes among all 2018 CRSS in-scope crashes. Since the fatal estimate constitutes only 0.5 percent of the total crashes, we can use the standard error of the percentage of non-fatal injury crashes $\hat{\theta}_{CRSS\ Non-fatal}$ to approximate the standard error of $\hat{\theta}_C$. For this example, $\hat{\theta}_{CRSS\ Non-fatal} = 28\%$ is the estimated percentage of injury crashes. The standard error estimate of $\hat{\theta}_{CRSS\ Non-fatal}$ is:

$$ste(\hat{\theta}_C) \approx ste(\hat{\theta}_{CRSS\ Non-fatal}) = 0.74\%$$

Table 3. SAS Output for Example 3

Table of SEVERITY					
SEVERITY	**Frequency**	**Weighted Frequency**	**Std Err of Wgt Freq**	**Percent**	**Std Err of Percent**
FATAL	884	33,654	2,180	0.4997	0.0368
INJURY	23,043	1,893,704	88,083	28.1198	0.7439
PDO	24,516	4,807,058	262,531	71.3805	0.7507
Total	48,443	6,734,416	333,709	100.0000	

3. Approximation by GVF

CRSS GVF is a function of the CRSS total estimate used to estimate the standard error of the CRSS total estimate:

$$ste(x) = e^{a+b*ln(x)+c*ln^2(x)}$$

Here x is a CRSS total estimate, $ste(x)$ is the standard error estimate of x. GVF above is fitted from the annual CRSS total estimates and their associated standard errors estimated using SAS SURVEY procedures. The estimated coefficients a, b, c and GVF tables for certain range of the total estimates are published in the *CRSS Analytic User's Manual* starting from 2020. Details on how the CRSS GVFs were developed and more examples of using the CRSS GVF can be found in Zhang & Diaz (2020). CRSS GVF provides a simple and consistent way of estimating the standard errors of total estimates. CRSS GVF does not require software specialized in survey data analysis.

CRSS GVF is mainly used for the standard errors of linear estimates. For a linear composite estimate:

$$\hat{t}_C = \hat{t}_{FARS} + \hat{t}_{CRSS\ Non-fatal},$$

let $ste(\hat{t}_C)$ be the standard error estimate of $\hat{t}_C$, $ste_{GVF}(\hat{t}_C)$ be the GVF standard error estimate of $\hat{t}_C$ obtained by replacing x in the CRSS GVF above with $\hat{t}_C$. We use the GVF estimate of $\hat{t}_{CRSS\ Non-fatal}$, $ste_{GVF}\left(\hat{t}_{CRSS\ Non-fatal}\right)$, to approximate $ste(\hat{t}_C)$:

$$ste_{GVF}\left(\hat{t}_{CRSS\ Non-fatal}\right) \approx ste\left(\hat{t}_{CRSS\ Non-fatal}\right) = ste(\hat{t}_C)$$

On the other hand, in many cases $\hat{t}_{FARS} \ll \hat{t}_{CRSS\ Non-fatal}$, therefore:

$$\hat{t}_C = \hat{t}_{FARS} + \hat{t}_{CRSS\ Non-fatal} \approx \hat{t}_{CRSS\ Non-fatal}.$$

Hence we can also use the GVF estimate $ste_{GVF}(\hat{t}_C)$ to approximate $ste_{GVF}\left(\hat{t}_{CRSS\ Non-fatal}\right)$ or $ste(\hat{t}_C)$:

$$ste_{GVF}(\hat{t}_C) \approx ste_{GVF}\left(\hat{t}_{CRSS\ Non-fatal}\right) \approx ste\left(\hat{t}_{CRSS\ Non-fatal}\right) = ste(\hat{t}_C).$$

The above equations indicate when $\hat{t}_{FARS}$ is a small portion of $\hat{t}_C$, we can plug $\hat{t}_C$ instead of $\hat{t}_{CRSS\ Non-fatal}$ into the CRSS GVF to approximate the standard error of $\hat{t}_C$.

For a non-linear composite estimator such as percentage, using the CRSS GVF is complicated and may not produce desirable result. We examine this in detail in the following example.

Example 4: Use CRSS GVF

In this example we recalculate the standard error estimates in Example 1 and 2 using the CRSS GVF. We first show how to use the CRSS GVF to estimate the standard error of a linear composite estimate. We then show why the CRSS GVF does not always produce desirable standard error estimate for non-linear estimate.

In Example 1 the linear composite estimate is:

$$\hat{t}_C = \hat{t}_{FARS} + \hat{t}_{CRSS\ Non-fatal} = 76{,}036 + 15{,}924{,}248 = 16{,}000{,}284$$

We need to use 2018 CRSS crash level GVF to calculate the GVF standard error estimate $ste_{GVF}\left(\hat{t}_{CRSS\ Non-fatal}\right)$. The estimated coefficients for the 2018 crash level GVF:

$$ste(x) = e^{a+b*ln(x)+c*ln^2(x)}$$

are:

a = 2.33242, b = 0.31521, c = 0.02258

(see Table 8 or Appendix C in Zhang & Diaz (2020 December)). Substitute these estimated parameters and point estimate $x = \hat{t}_{CRSS\ Non-fatal} = 15{,}924{,}248$ into the above GVF:

$$ste_{GVF}\left(\hat{t}_{CRSS\ Non-fatal}\right) = 954{,}870$$

compared with $ste\left(\hat{t}_{CRSS\ Non-fatal}\right) = 857{,}870$ in Example 1. On the other hand, $\hat{t}_{FARS} = 76{,}036 \ll \hat{t}_{CRSS\ Non-fatal} = 15{,}924{,}248$. Therefore, if the point estimate $x = \hat{t}_C$=16,000,284 is plugged in the GVF:

$$ste_{GVF}(\hat{t}_C) = 959{,}723.$$

For Example 2, using the CRSS GVF to estimate the standard error of a non-linear estimate is complicated and may produce imaginary number standard error estimates. To use the GVF to estimate the standard error of a percentage estimate (a non-linear estimate), we use the following approximation.

$$ste\left(\hat{R}\right) \approx \hat{R}\sqrt{\frac{var\left(\hat{X}_d\right)}{\hat{X}_d^2} - \frac{var\left(\hat{X}_p\right)}{\hat{X}_p^2}}$$

Here:

$\hat{R} = \frac{\hat{X}_d}{\hat{X}_p}$ is a percentage estimator.

$\hat{X}_p$ is the total estimate of X (numeric or categorical) for population p.

$\hat{X}_d$ is the total estimate of X for domain d within population p.

$var\left(\hat{X}_d\right)$ and $var\left(\hat{X}_p\right)$ are the variance estimates of linear estimates $\hat{X}_d$ and $\hat{X}_p$ respectively.

Assume we need to estimate the percentage of fatal and injury crashes among all crashes. The following SAS output has all the point and variance estimates needed.

Table 4. SAS Output for the Example 4

Table of SEVERITY					
SEVERITY	**Frequency**	**Weighted Frequency**	**Std Err of Wgt Freq**	**Percent**	**Std Err of Percent**
FATAL	884	33,654	2,180	0.4997	0.0368
INJURY	23,043	1,893,704	88,083	28.1198	0.7439
PDO	24,516	4,807,058	262,531	71.3805	0.7507
Total	48,443	6,734,416	333,709	100.0000	
Table of FATAL_FLAG					
FATAL_FLAG	**Frequency**	**Weighted Frequency**	**Std Err of Wgt Freq**	**Percent**	**Std Err of Percent**
NON-FATAL	47,559	6,700,762	333,302	99.5003	0.0368
FATAL	884	33,654	2,180	0.4997	0.0368
Total	48,443	6,734,416	333,709	100.0000	
Table of INJURED					
INJURED	**Frequency**	**Weighted Frequency**	**Std Err of Wgt Freq**	**Percent**	**Std Err of Percent**
NO INJURY	24,516	4,807,058	262,531	71.3805	0.7507
INJURED OR FATAL	23,927	1,927,358	88,421	28.6195	0.7507
Total	48,443	6,734,416	333,709	100.0000	

Because of the calibration performed to the fatal crash estimates in the CRSS weighting process, $\hat{X}_{CRSS\,Fatal} \approx \hat{X}_{FARS}$. Therefore,

$$\hat{X}_d = \hat{X}_{d,FARS} + \hat{X}_{d,CRSS\,Injury} = 1{,}927{,}358$$

$$\hat{X}_p = \hat{X}_{FARS} + \hat{X}_{CRSS\,Injury} + \hat{X}_{CRSS\,PDO} = 6{,}734{,}416$$

$$\hat{R} = \frac{\hat{X}_d}{\hat{X}_p} \approx 29\%$$

$$ste\left(\hat{X}_d\right) = ste\left(\hat{X}_{CRSS\,Injury}\right) = 88{,}083$$

$$ste\left(\hat{X}_p\right) = ste\left(\hat{X}_{CRSS\,Injury} + \hat{X}_{CRSS\,PDO}\right) = 333{,}302$$

These estimates, however, lead to:

$$\frac{var\left(\hat{X}_d\right)}{\hat{X}_d^2} - \frac{var\left(\hat{X}_p\right)}{\hat{X}_p^2} = -0.00036 < 0.$$

Thus, there is no real number estimate using formula:

$$ste(\hat{R}) \approx \hat{R}\sqrt{\frac{var(\hat{X}_d)}{\hat{X}_d^2} - \frac{var(\hat{X}_p)}{\hat{X}_p^2}}.$$

In other words, although the GVF models were developed under the premise that the relative variance $V^2 = Var(X)/X^2$ is a decreasing function of the expectation X, it is still possible the premise does not hold in actual calculation because estimates and approximations are used in actual calculation while the premise is developed for the true variance and true expectation.

For linear composite estimates, GVF method only requires point estimates so it's easy to be implemented. For non-linear estimates, GVF method requires more calculation and there is no guarantee that it always produces a real number GVF estimate.

4. The Proposed Method

FARS is a census of fatal crashes that can be viewed as a probability sample with all fatal crashes selected with certainty. CRSS is a probability sample that covers all police-reported crashes: fatal or non-fatal. The composite estimates can then be viewed as domain estimates estimated from the concatenated FARS and CRSS data set under a nominal sample design.

Let U_{FARS} be the set of fatal crashes in FARS file, $S_{CRSS\,Fatal}$ be the set of sampled CRSS fatal crashes, $S_{CRSS\,Non-fatal}$ be the set of sampled CRSS non-fatal crashes, and $S_{CRSS} = S_{CRSS\,Fatal} \cup S_{CRSS\,Non-fatal}$ be the set of all sampled CRSS crashes (fatal and non-fatal). We first define a domain identifier D_j separately on S_{CRSS} and U_{FARS}:

$$D_j = \begin{cases} 1, & if\ j \in S_{CRSS\,Non-fatal} \\ 0, & if\ j \in S_{CRSS\,Fatal} \end{cases}$$

$$D_j = 1, \qquad all\ j \in U_{FARS}$$

We then define variable y_{ij}^* on S_{CRSS} and U_{FARS} as:

$$y_{ij}^* = D_j y_{ij}, \qquad j \in S_{CRSS}\ or\ j \in U_{FARS}\ (i = 1,2, \dots K\ are\ study\ variable\ subscripts)$$

In other words, y_{ij}^* equals to zero for all CRSS fatal crashes and equals to y_{ij} otherwise.

We also need to introduce the concatenated set of S_{CRSS} and U_{FARS}: $S_{CRSS}||U_{FARS}$. The concatenated set of S_{CRSS} and U_{FARS} is the totality of S_{CRSS} and U_{FARS} with the duplicates since S_{CRSS} and U_{FARS} are overlapping. The difference between the concatenated set $S_{CRSS}||U_{FARS}$ and the union set $S_{CRSS} \cup U_{FARS}$ is that the same unit only appear once in the union set while the concatenated set keeps the duplicates.

We further define new total estimators on U_{FARS} and S_{CRSS}:

$$\hat{t}_{i,U_{FARS}}^* = \sum\nolimits_{j \in U_{FARS}} y_{ij}^* \qquad (i = 1,2, \dots K)$$

$$\hat{t}_{i,S_{CRSS}}^* = \sum\nolimits_{j \in S_{CRSS}} w_j y_{ij}^* \qquad (i = 1,2, \dots K)$$

Then we can define a new composite total estimator on $S_{CRSS}||U_{FARS}$:

$$\hat{t}_{i,S_{CRSS}||U_{FARS}}^* = \hat{t}_{i,U_{FARS}}^* + \hat{t}_{i,S_{CRSS}}^* \quad (i = 1,2, \dots K).$$

Notice in Section 1 we defined:

$$\hat{t}_{i,FARS} = \sum\nolimits_{j \in U_{FARS}} y_{ij} \quad (i = 1,2, \dots K)$$

$$\hat{t}_{i,CRSS\,Non-fatal} = \sum\nolimits_{j \in S_{CRSS\,Non-fatal}} w_j y_{ij} \quad (i = 1,2, \dots K).$$

Notice:

$$\hat{t}^*_{i,U_{FARS}} = \sum_{j \in U_{FARS}} y^*_{ij} = \sum_{j \in U_{FARS}} y_{ij} = \hat{t}_{i,FARS}$$

$$\hat{t}^*_{i,S_{CRSS}} = \sum_{j \in S_{CRSS}} w_j y^*_{ij} = \sum_{j \in S_{CRSS\ Non-fatal}} w_j y_{ij} = \hat{t}_{i,CRSS\ Non-fatal}$$

Therefore, the new composite total estimator $\hat{t}^*_{i,S_{CRSS}||U_{FARS}}$ is equal to $\hat{t}_{ci}$:

$$\hat{t}^*_{i,S_{CRSS}||U_{FARS}} = \hat{t}^*_{i,U_{FARS}} + \hat{t}^*_{i,S_{CRSS}} = \hat{t}_{i,FARS} + \hat{t}_{i,CRSS\ Non-fatal} = \hat{t}_{ci}, \qquad (i = 1,2, \dots K)$$

Therefore, the non-linear composite estimator ($\hat{\theta}^*_C$) as a function of $\hat{t}^*_{i,S_{CRSS}||U_{FARS}}$ is equal to the non-linear composite estimator ($\hat{\theta}_C$) as a function of $\hat{t}_{ci}$:

$$\hat{\theta}^*_C = f\left(\hat{t}^*_{1,S_{CRSS}||U_{FARS}}, \dots \hat{t}^*_{K,S_{CRSS}||U_{FARS}}\right) = f(\hat{t}_{c1}, \dots \hat{t}_{cK}) = \hat{\theta}_C.$$

The composite estimator $\hat{\theta}_C$ can be viewed as a dual frame estimator where one probability sample is selected from the CRSS frame and one census is taken from the FARS frame. The overlapping fatal cases in the CRSS sample are removed from the estimator. The new estimator $\hat{\theta}^*_C$, on the other hand, can be viewed as a domain estimator of a single frame estimation where the FARS is one of the strata of the population selected with certainty and the CRSS is the rest strata. The overlapping fatal cases in the CRSS sample are kept in the estimator but do not contribute to the point estimate because of the definition of y^*_{ij}.

Notice there is a 1-1 correspondence between the dual frame sample and the single frame sample and the selection probabilities of the corresponding samples are the same. In addition, as we shown above, for each pair of the corresponding samples: $\hat{\theta}^*_C = \hat{\theta}_C$. Therefore, by the definition of the sampling expectation and the sampling variance:

$$E\left(\hat{\theta}^*_C\right) = E\left(\hat{\theta}_C\right)$$

$$Var\left(\hat{\theta}^*_C\right) = Var\left(\hat{\theta}_C\right)$$

Therefore, instead of estimating the variance of $\hat{\theta}_C$, we can estimate the variance of $\hat{\theta}^*_C$ defined on the concatenated sample $S_{CRSS}||U_{FARS}$. FARS and CRSS are independent surveys with overlapping fatal crash sub-population. We can view FARS and CRSS as two sets of strata. Within the FARS stratum, we group all fatal crashes into one nominal PSU and all fatal crashes within the singleton PSU are selected with certainty (selection probability equals to 1). For the 25 CRSS strata, the CRSS sampling procedure is carried out to select the CRSS sample: PSU, PJ, and PCR samples. To obtain $\hat{\theta}^*_C$ defined on the concatenated sample $S_{CRSS}||U_{FARS}$, we create the domain identifier D_j on $S_{CRSS}||U_{FARS}$ and use D_j to perform a domain analysis. To estimate variances under this nominal sample design, we need to invoke without-replacement variance estimation method in the FARS stratum because all fatal crashes are selected with certainty and invoke with-replacement variance estimation method in the rest 25 CRSS strata because of the low PSU sampling rates. Notice we do not delete the fatal crashes in the CRSS sample S_{CRSS} because otherwise the CRSS sample is altered.

SUDAAN can invoke different design option in different PSU strata. Therefore, SUDAAN can handle this nominal sample design. There is no need to create the composite variable y^*_{ij}. SUDAAN can use D_j and the existing y_{ij} to perform the domain estimation.

Currently SAS 9.4 cannot invoke different design options in different strata. However, when there is only one PSU in a stratum, SAS uses the observations in the singleton PSU for point estimation but does not use the data in the singleton PSU for variance estimation regardless there is sampling error or not in the second stage sampling within the singleton PSU. In other words, the records in the FARS stratum are used for point estimate but do not contribute to the variance estimation. This is exactly the way we would like to handle the FARS stratum. Therefore, ironically SAS SURVEY procedures can also be used for this nominal sample design.

The survey package in R explicitly provides options of handling the stratum with singleton PSU. Statement:

```
options(survey.lonely.psu = "remove")
```

in R's survey package tells R to ignore the singleton PSU for variance computation (Lumley, 2020, page 56). Therefore, the survey package in R can also be used to make composite estimator standard error estimation under this nominal sample design.

Compared with the other methods, this method of using concatenated data set has the advantage of obtaining point estimates and standard error estimates in one procedure. In addition, it can be used not only to make point estimates and the associated standard errors but also can be used to perform more complicated analysis such as linear regression and logistic regression analysis. Next, we explain how we implement the proposed method in SAS and SAS-callable SUDAAN with three examples:

- Example 5: Simple estimates (using SUDAAN and SAS).
- Example 6: Logistic regression (using SUDAAN and SAS).
- Example 7: Year-to-year comparison (using SUDAAN).

R programs for all examples 1-7 can be found in Appendix B.

Example 5: Use the Proposed Method to Make Simple Estimates

In this example, we use the proposed method to reproduce the Example 1 estimate: the total number of people in motor vehicles in transport (PERMVIT) and the Example 2 estimate: the percentage of fatal or injury crashes among all CRSS in-scope crashes.

FARS and CRSS data sets must be concatenated in a structural way. First, all FARS cases are grouped into one stratum. Within this FARS stratum, all FARS cases are assigned to a single nominal PSU and this nominal PSU is selected with certainty. Then within this nominal PSU, all FARS cases are selected with probability 1. This way, FARS cases can be viewed as the result of a two-stage sample selection: at the first stage the only nominal PSU in the stratum is selected with certainty and at the second stage all PCRs in the selected nominal PSU are selected with certainty using simple random sampling without replacement (SRSWOR).

Second, all 25 CRSS PSU strata defined by the CRSS variable PSUSTRAT are treated parallelly as the FARS stratum so now there are total 26 first stage strata. Within each of the 25 CRSS PSU strata, the CRSS multi-stage sample selection procedure is carried out: PSU, PJ, and PCR sample selection. Because of the low PSU level sampling rate, CRSS PSU sample can be treated as selected with-replacement as we normally do when we make the CRSS variance estimates.

In the following we first describe how to prepare the data for analysis using SUDAAN and how to specify design option in SUDAAN estimation procedure. We then describe how to prepare the data for analysis using SAS followed by examples of SAS estimation procedure.

To specify this nominal sample design in SUDAAN design statements, we need to invoke two-stage SRSWOR in the FARS stratum and invoke multi-stage with-replacement sampling in the rest 25 CRSS strata. The following SAS data step concatenates the FARS and the CRSS data sets and prepares the variables needed for analysis using SUDAAN.

```
/* Prepare all variables for SUDAAN */
DATA CRASH_SEV_C;
      SET CRSS2018.ACCIDENT (IN=CRSS2018) FARS2018.ACCIDENT
      (IN=FARS2018);
      KEEP PSUSTRAT PSU_VAR PSU_CNT WEIGHT DOMAIN_FLAG PERMVIT SEVERITY
      INJURED FATAL_FLAG DAYLIGHT VE_FORMS;
      IF FARS2018 THEN DO;
            PSUSTRAT=99;    /*Pseudo PSU stratum for FARS crashes*/
            PSU_VAR=999;    /*Pseudo PSU for FARS crashes*/
            PSU_CNT=0;      /*Pseudo PSU contributes zero variance*/
            WEIGHT=1;       /*FARS cases were selected with certainty*/
            DOMAIN_FLAG=1; /*Domain flag for FARS cases*/
            SEVERITY=1;     /*Crash severity: Fatal crash*/
            FATAL_FLAG=1;   /*Fatal flag*/
            END;
      ELSE IF CRSS2018 THEN DO;
            PSU_CNT=-1;     /*Invoke with-replacement option at CRSS PSU
            level*/
            IF MAXSEV_IM=4 THEN DOMAIN_FLAG=0; ELSE DOMAIN_FLAG=1;
            /*Domain flag for CRSS cases*/
            IF MAXSEV_IM=4 THEN SEVERITY=1; /*Fatal crash*/
            ELSE IF MAXSEV_IM IN (1,2,3,5) THEN SEVERITY=2; /*Injury
```

```
            crash*/
            ELSE IF MAXSEV_IM IN (0,6,8) THEN SEVERITY=3; /*PDO crash*/
            IF MAXSEV_IM=4 THEN FATAL_FLAG=1; ELSE FATAL_FLAG=0;
            /*Fatal flag*/
            END;
      IF SEVERITY IN (1,2) THEN INJURED=1; ELSE INJURED=0; /*Injured or
      not*/
      DAYLIGHT=(LGT_COND^=1)+1; /*1=Daylight, 2=otherwise*/
      RUN;

PROC SORT DATA=CRASH_SEV_C; BY PSU_VAR; RUN;

PROC MEANS DATA=CRASH_SEV_C NOPRINT;
      BY PSU_VAR;
      OUTPUT OUT=CRASH_CNT (DROP=_TYPE_ _FREQ_) N=CRASH_CNT;
      RUN;

DATA CRASH_SEV_D;
      MERGE CRASH_SEV_C (IN=A) CRASH_CNT (IN=B);
      BY PSU_VAR;
      IF A & B;
      RUN;
```

In the SAS data step above all FARS cases are assigned to a single stratum, PSUSTRAT=99, and a single PSU, PSU_VAR=999. Variable PSU_CNT is created to tell SUDAAN what variance estimation method to be used or how many PSUs in the PSU stratum. All FARS cases are assigned with PSU_CNT=0 that tells SUDAAN that PSU 999 does not contribute any sampling variance at PSU level because PSU 999 is the only PSU in stratum 99 and it is selected with certainty. All FARS cases are assigned WEIGHT=1 because they were selected with certainty. Variable DOMAIN_FLAG is created to define the analysis domain. All FARS cases are assigned with DOMAIN_FLAG=1 as we described earlier.

For the CRSS cases the existing variables PSUSTRAT and PSU_VAR are ready to be used. All CRSS cases are assigned with PSU_CNT=-1 that tells SUDAAN to invoke with-replacement option in all 25 CRSS PSU strata. All CRSS fatal crashes are assigned with DOMAIN_FLAG=0. All CRSS non-fatal crashes are assigned with DOMAIN_FLAG=1.

The PROC MEANS procedure following the data step calculates the crash sample size within each PSU. For the nominal FARS PSU 999, its crash sample size equals to the fatal crash population size because all fatal crashes were sampled. This sample size for FARS PSU 999 will be used as the PSU level crash population size for the second stage variance estimation because we told SUDAAN to invoke SRSWOR for the second stage sample selection in this FARS PSU 999. Since the sample size equals to the population size in this PSU, the second stage sampling does not contribute any sampling variance in this PSU.

The PSU level crash sample sizes for the 25 CRSS PSUs will not be used for variance estimation because we tell SUDAAN to invoke with-replacement sample selection at PSU level. All 26 PSU level sample sizes are saved in variable CRASH_CNT. The following SAS-callable SUDAAN procedure reproduces the results of Example 1 and Example 2.

```
/* Reproduce Example 1 using SUDAAN*/
PROC DESCRIPT DATA=CRASH_SEV_D DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="CRSS Non-Fatal & FARS";
      VAR PERMVIT;
      PRINT NSUM="Sample Size" WSUM="Population Size"
            TOTAL="Total Estimate" SETOTAL="SE of Total"
            LOWTOTAL UPTOTAL;
      RUN;

/* Reproduce Example 2 using SUDAAN*/
PROC CROSSTAB DATA=CRASH_SEV_D DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
      TABLES INJURED;
      CLASS  INJURED;
      SETENV PAGESIZE=80 LINESIZE=70;
      PRINT NSUM="Sample Size" WSUM="Population Size"
            SEWGT="Pop Size SE" ROWPER="Percent" SEROW="Standard Error"
            / NSUMFMT=F6.0 WSUMFMT=F8.0 SEWGTFMT=F8.0;
      RUN;
```

The design option DESIGN=WOR tells SUDAAN to invoke SRSWOR. The NEST statement lists the PSU strata ID variable (PSUSTRAT) and PSU ID variable (PSU_VAR), which also indicates a multi-stage sample selection. The TOTCNT statement lists the PSU stratum level PSU population size variable (PSU_CNT) and the PSU level crash population size variable (CRASH_CNT) for the simple random sampling without-replacement variance estimation. In addition, if a TOTCNT variable equals to 0 then it means the corresponding unit does not contribute sampling variance, or if a TOTCTN variable equals to -1 then it means the corresponding unit was selected with-replacement. In our case, PSU_CNT=0 at PSU level for the FARS stratum 99 that tells SUDAAN the only PSU in the FARS stratum does not contribute any sampling variance. For the remaining CRSS PSUs, PSU_CNT=-1 that tells SUDAAN to invoke with-replacement variance estimation (DESIGN=WR).

At the second stage, variable CRASH_CNT specifies the crash population sizes for each PSU. CRASH_CNT equals to the fatal population size for the FARS PSU 999. A SRSWOR design with the sample size equals to the population size means every crash is selected with certainty. Therefore, the second stage sample selection in this FARS PSU does not contribute any sampling variance. For the 25 CRSS PSU strata, SUDAAN ignores the second stage population sizes in CRASH_CNT when DESIGN=WR is invoked at the first stage.

For Example 1 (the total number of people in motor vehicles in transport), SUDAAN produces the same point and standard error estimates as in Example 1.

Table 5. Use the Proposed Method for Example 1 – SUDAAN Output

Variable		SUDAAN Reserved Variable One	
		Total	1
Number of Persons in Motor Vehicles In-Transport	Sample Size	81478	81478
	Population Size Total Estimate	6734681.34	6734681.34
		16000284.26	16000284.26
	SE of Total	857869.65	857869.65
	Lower 95% Limit Total	14269033.23	14269033.23
	Upper 95% Limit Total	17731535.28	17731535.28

$$\hat{t}_C = 16{,}000{,}284$$

$$ste_d(\hat{t}_C) = 857{,}870.$$

For Example 2 (estimating the percentage of fatal or injury crashes), SUDAAN procedure reproduce the same results as in Example 2.

Table 6. Use the Proposed Method for Example 2 – SUDAAN Output

		INJURED		
		Total	0	1
	Sample Size	81478	24516	56962
	Population Size	6734681	4807058	1927623
	Pop Size SE	333302	262531	88083
	Percent	100.00	71.38	28.62
	Standard Error	0.00	0.75	0.75

$$\hat{\theta}_C \approx 29\%$$

$$ste(\hat{\theta}_C) = 0.75\%$$

To specify this nominal sample design in SAS, we use the default design option: PSUs are selected with-replacement within each PSU stratum. For the FARS stratum, since there is only one PSU, SAS does not use this PSU in the variance estimation and ignores any sampling error in the subsequent sampling stages. Accidentally this is how we want SAS to handle the FARS stratum. Therefore, there is no special treatment for the FARS stratum except creating the nominal stratum, the nominal PSU, and the weight variable. The only change to the data step is deleting the PSU_CNT variable and the calculation of PSU level crash counts. The following SAS data step prepares data for SAS estimation procedure.

```
/* Prepare all variables for SAS */
DATA CRASH_SEV_SAS;
      SET CRSS2018.ACCIDENT (IN=CRSS2018) FARS2018.ACCIDENT
      (IN=FARS2018);
      KEEP PSUSTRAT PSU_VAR WEIGHT DOMAIN_FLAG PERMVIT SEVERITY INJURED
      FATAL_FLAG DAYLIGHT VE_FORMS;
      IF FARS2018 THEN DO;
            PSUSTRAT=99;    /*Pseudo PSU stratum for FARS crashes*/
            PSU_VAR=999;    /*Pseudo PSU for FARS crashes*/
            WEIGHT=1;       /*FARS cases were selected with certainty*/
            DOMAIN_FLAG=1; /*Domain flag for FARS cases*/
            SEVERITY=1;    /*Crash severity: Fatal crash*/
            FATAL_FLAG=1;  /*Fatal flag*/
            END;
      ELSE IF CRSS2018 THEN DO;
            IF MAXSEV_IM=4 THEN DOMAIN_FLAG=0; ELSE DOMAIN_FLAG=1;
            /*Domain flag for CRSS cases*/
            IF MAXSEV_IM=4 THEN SEVERITY=1; /*Fatal crash*/
            ELSE IF MAXSEV_IM IN (1,2,3,5) THEN SEVERITY=2; /*Injury
            crash*/
            ELSE IF MAXSEV_IM IN (0,6,8) THEN SEVERITY=3; /*PDO crash*/
            IF MAXSEV_IM=4 THEN FATAL_FLAG=1; ELSE FATAL_FLAG=0;
            /*Fatal flag*/
            END;
      IF SEVERITY IN (1,2) THEN INJURED=1; ELSE INJURED=0; /*Injured or
```

```
not*/
DAYLIGHT=(LGT_COND^=1)+1; /*1=Daylight, 2=otherwise*/
RUN;
```

The following SAS SURVEY procedures produce the same Example 1 and Example 2 results as SUDAAN procedures:

```
/* Reproduce Example 1 using SAS*/
PROC SURVEYMEANS DATA=CRASH_SEV_SAS SUM STD CLSUM;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      DOMAIN DOMAIN_FLAG;
      WEIGHT WEIGHT;
      VAR PERMVIT;
      RUN;
```

Table 7. Use the Proposed Method for Example 1 – SAS Output

Statistics for DOMAIN_FLAG Domains						
DOMAIN _FLAG	**Variable**	**Label**	**Sum**	**Std Error of Sum**	**95% CL for Sum**	
0	PERMVIT	Number of People in Motor Vehicles In-Transport	72,984	5,266.25	62,356.5	83,611.9
1	PERMVIT	Number of People in Motor Vehicles In-Transport	16,000,284	857,870	14,269,033.2	17,731,535.3

```
/* Reproduce Example 2 using SAS*/
PROC SURVEYFREQ DATA=CRASH_SEV_SAS;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      TABLE DOMAIN_FLAG*INJURED / ROW;
      FORMAT INJURED INJURED.;
      RUN;
```

Table 8. Use the Proposed Method for Example 2 – SAS Output

Table of DOMAIN_FLAG by INJURED								
DOMAIN _FLAG	**INJURED**	**Frequency**	**Weighted Frequency**	**Std Err of Wgt Freq**	**Percent**	**Std Err of Percent**	**Row Percent**	**Std Err of Row Percent**
0	NO INJURY	0	.	.	.	.	.	.

Table of DOMAIN_FLAG by INJURED								
DOMAIN _FLAG	**INJURED**	**Frequency**	**Weighted Frequency**	**Std Err of Wgt Freq**	**Percent**	**Std Err of Percent**	**Row Percent**	**Std Err of Row Percent**
	INJURED OR FATAL	884	33,654	2,180	0.4972	0.0366	100.0000	0.0000
	Total	884	33,654	2,180	0.4972	0.0366	100.0000	
1	NO INJURY	24,516	4,807,058	262,531	71.0228	0.7542	71.3777	0.7527
	INJURED OR FATAL	56,962	1,927,623	88,083	28.4800	0.7470	28.6223	0.7527
	Total	81,478	6,734,681	333,302	99.5028	0.0366	100.0000	
Total	NO INJURY	24,516	4,807,058	262,531	71.0228	0.7542		
	INJURED OR FATAL	57,846	1,961,277	88,421	28.9772	0.7542		
	Total	82,362	6,768,335	333,709	100.0000			

SAS recognizes the single PSU in the FARS stratum and produces the following note in the log.

```
NOTE: Only one cluster in a stratum for variable(s) PERMVIT. The
estimate of variance for PERMVIT will omit this stratum.
```

In SAS SURVEYREG procedure however, SAS automatically collapses singleton PSU stratum with another PSU stratum. Therefore, when using SAS SURVEYREG, option NOCOLLAPSE should be added to the STRATA statement to prevent SAS from collapsing the strata.

Example 6: Use the Proposed Method in Logistic Regression Analysis

In this example, we fit a logistic regression model using the proposed method and the concatenated FARS and CRSS data set and then fit the same model using only the CRSS data to compare the results. The dependent variable is INJURED (1 if injury or fatal crash, 0 if PDO crash). The independent variable is DAYLIGHT (1 if crash occurred in day light condition, 2 otherwise). For the simplicity of presentation, we imputed all missing and unknown in DAYLIGHT as 2.

In the following programs, we first fit the model to the concatenated FARS and CRSS data set created in Example 5 data step using SUDAAN RLOGIST procedure and SAS SURVEYLOGISTIC procedure. Both dependent variable INJURED and independent variable DAYLIGHT were created in the data step of Example 5.

```
PROC FORMAT;
      VALUE DAYLIGHT 1="Daylight" 2="Others";
      VALUE INJURED 1="Injured or fatal" 0="PDO";
      RUN;

/* SUDAAN procedure using the concatenated FARS and CRSS data set. */
PROC RLOGIST DATA=CRASH_SEV_D DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
      SUBGROUP DAYLIGHT;
      LEVELS 2;
      REFLEVEL DAYLIGHT=2;
      MODEL INJURED = DAYLIGHT;
      SETENV PAGESIZE=80 LINESIZE=70;
      PRINT / BETAFMT=F8.4 SEBETAFMT=F8.4;
      RFORMAT DAYLIGHT DAYLIGHT.;
      RFORMAT INJURED INJURED.;
      RUN;

/* SAS procedure using the concatenated FARS and CRSS data set. */
PROC SURVEYLOGISTIC DATA=CRASH_SEV_SAS VARMETHOD=TAYLOR;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      CLASS DAYLIGHT / REF=LAST PARAM=REFERENCE;
      DOMAIN DOMAIN_FLAG;
      MODEL INJURED (EVENT='1') = DAYLIGHT;
      RUN;
```

CRSS is a standalone probability sample therefore the CRSS data alone can also be used for logistic regression analysis. We fit the same model to the CRSS data (fatal and non-fatal crashes) to compare with the model using the concatenated data set. The following programs prepare the data and fit the same model.

```
/* SUDAAN procedure using only CRSS data set. */
DATA CRSS2018;
      SET CRSS2018.ACCIDENT;
      KEEP PSUSTRAT PSU_VAR WEIGHT INJURED DAYLIGHT;
      IF MAXSEV_IM IN (1,2,3,4,5) THEN INJURED=1;
      ELSE INJURED=0; /*Injured or not*/
      DAYLIGHT=(LGT_COND^=1)+1; /*1=Daylight, 2=Otherwise*/
      RUN;

PROC RLOGIST DATA=CRSS2018 NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      WEIGHT WEIGHT;
      SUBGROUP DAYLIGHT;
      LEVELS 2;
      REFLEVEL DAYLIGHT=2;
      MODEL INJURED = DAYLIGHT;
      SETENV PAGESIZE=80 LINESIZE=70;
      PRINT / BETAFMT=F8.4 SEBETAFMT=F8.4;
      RFORMAT DAYLIGHT DAYLIGHT.;
      RFORMAT INJURED INJURED.;
      RUN;

/* SAS procedure using only CRSS data set. */
PROC SURVEYLOGISTIC DATA=CRSS2018;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      CLASS DAYLIGHT / REF=LAST PARAM=REFERENCE;
      MODEL INJURED (EVENT='1') = DAYLIGHT;
      RUN;
```

Table 5 lists some key statistics produced from the above two procedures in SUDAAN and two procedures in SAS. The SUDAAN and SAS results using the same data set are identical. The difference between the two data sets are small as we expected because for this example the fatal domain is a very small portion of the analysis domain. The standard error estimates using the concatenated data are slightly smaller than the CRSS data only estimates as we expected.

Table 9. Statistics Produced by the Same Model Using Different Data Sets

Statistics	Concatenated Data		CRSS Only	
	SUDAAN	SAS	SUDAAN	SAS
Intercept	-0.8695	-0.8695	-0.8708	-0.8708
Intercept SE	0.0451	0.0451	0.0452	0.0452
Daylight	-0.0645	-0.0645	-0.0628	-0.0628
Daylight SE	0.0411	0.0411	0.0413	0.0413
Daylight T-test	-1.57	-1.57	-1.52	-1.52
Daylight T-test P value	0.1236	0.1236	0.1357	0.1357
Daylight odds ratio	0.94	0.94	0.94	0.94

Example 7: Use the Proposed Method for Multiple Year Comparison

The proposed method and the concatenated data set can also be used for multiple year comparison. The following data step prepares the data. Compared with the data step of Example 5, the only changes are multiple years of data are combined and variable YEAR is created for multiple year comparison.

```
/* Prepare all variables */
DATA CRASH_SEV_E;
SET   CRSS2018.ACCIDENT (IN=CRSS2018) CRSS2019.ACCIDENT (IN=CRSS2019)
      FARS2018.ACCIDENT (IN=FARS2018) FARS2019.ACCIDENT (IN=FARS2019);
      KEEP PSUSTRAT PSU_VAR PSU_CNT WEIGHT DOMAIN_FLAG YEAR INJURED;
      YEAR = 2018*SUM(CRSS2018,FARS2018) + 2019*SUM(CRSS2019,FARS2019);
      IF (FARS2018 OR FARS2019) THEN DO;
      PSUSTRAT=99;    /*Pseudo PSU stratum for FARS crashes*/
      PSU_VAR=999;    /*Pseudo PSU for FARS crashes*/
      PSU_CNT=0;      /*Pseudo PSU contributes zero variance*/
      WEIGHT=1;       /*FARS cases were selected with certainty*/
      DOMAIN_FLAG=1; /*Domain flag for FARS cases*/
      INJURED=1;     /*Injury crashes*/
      END;
      ELSE IF (CRSS2018 OR CRSS2019) THEN DO;
      PSU_CNT=-1;     /*Invoke with-replacement option at CRSS PSU
      level*/
      IF MAXSEV_IM=4 THEN DOMAIN_FLAG=0; ELSE DOMAIN_FLAG=1; /*Domain
      flag for CRSS cases*/
      IF MAXSEV_IM IN (1,2,3,4,5) THEN INJURED=1; ELSE INJURED=0;
      /*Injured or not*/
      END;
      RUN;
```

In the following SUDAAN procedure, we first calculate 2018 and 2019 CRSS total and percentage injury and fatal crash estimates. We then call SUDAAN procedure DESCRIPT to make pairwise total comparison followed by another DESCRIPT procedure to make pairwise percentage comparison.

```
/* Calculate two year total estimates */
PROC CROSSTAB DATA=CRASH_SEV_F DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
      CLASS YEAR INJURED;
      TABLES YEAR*INJURED;
      SETENV ROWWIDTH=15 COLWIDTH=15 LABWIDTH=25;
      PRINT  NSUM="Sample Size" WSUM="Population Size"
             SEWGT="Pop Size SE" ROWPER="Percent"
      / NSUMFMT=F8.0 WSUMFMT=F10.0 SEWGTFMT=F9.0 ROWPERFMT=F6.4;
      RFORMAT INJURED INJURED.;
      RUN;

/* Compare multiple year totals */
PROC DESCRIPT DATA=CRASH_SEV_F DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
      CLASS YEAR INJURED;
      TABLES INJURED;
      VAR _ONE_;
      PAIRWISE YEAR / NAME="Year to Year Comparison";
      SETENV ROWWIDTH=15 COLWIDTH=15 LABWIDTH=30;
      PRINT NSUM="Sample Size" TOTAL="Difference"
            SETOTAL="Difference SE" LOWTOTAL UPTOTAL
            / NSUMFMT=F6.0 TOTALFMT=F12.0 SETOTALFMT=F12.0
              LOWTOTALFMT=F12.0 UPTOTALFMT=F12.0;
      RFORMAT INJURED INJURED.;
      RUN;

/* Compare multiple year percentages */
PROC DESCRIPT DATA=CRASH_SEV_F DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
      CLASS YEAR INJURED;
      VAR INJURED;
      CATLEVEL 1;
      PAIRWISE YEAR / NAME="Year to Year Comparison";
      SETENV ROWWIDTH=20 COLWIDTH=40 LABWIDTH=30;
      PRINT NSUM="Sample Size" PERCENT="Percent Difference"
            SEPERCENT="Percent Diff SE" LOWPCT UPPCT
            / NSUMFMT=F20.0 PERCENTFMT=F6.4 SEPERCENTFMT=F6.4
              LOWPCTFMT=F6.4 UPPCTFMT=F6.4;
      RFORMAT INJURED INJURED.;
      RUN;
```

In R, svycontrast() is used to compute linear and non-linear contrasts of estimates produced by survey functions and confint() can be applied to the svycontrast() object to calculate confidence intervals for estimates. When calculating non-linear estimates such as percent, the covmat = TRUE option must be used in the survey function to compute covariance between estimates for different subsets. By default, confint() sets the degrees of freedom to infinity which is larger than the actual degrees of freedom, the number of PSU's minus the number of strata, because of this the default SE is smaller than expected causing the confidence interval to be smaller than expected too. The degrees of freedom can be adjusted within the confint() function using the df option. See Appendix B for example R code.

Table 10. 2018 and 2019 Total and Percentage Injury or Fatal Crash Estimates

Crash Date (Year)		INJURED		
		Total	NO INJURY	INJURED OR FATAL
Total	Sample Size	168152	51689	116463
	Population Size	13490522	9613311	3877211
	Pop Size SE	669284	531143	175301
	Percent	100.0000	71.2597	28.7403
2018	Sample Size	81478	24516	56962
	Population Size	6734681	4807058	1927623
	Pop Size SE	333302	262531	88083
	Percent	100.0000	71.3777	28.6223
2019	Sample Size	86674	27173	59501
	Population Size	6755841	4806253	1949588
	Pop Size SE	340199	272166	88907
	Percent	100.0000	71.1422	28.8578

Table 11. 2018 and 2019 Total Injury or Fatal Crash Comparison

INJURED		Contrast Year to Year Comparison: (2018,2019)
Total	Sample Size	168152
	Difference	-21159
	Difference SE	75569
	Lower 95% Limit Cntrst Total	-173664
	Upper 95% Limit Cntrst Total	131345
NO INJURY	Sample Size	51689
	Difference	805
	Difference SE	62292
	Lower 95% Limit Cntrst Total	-124906
	Upper 95% Limit Cntrst Total	126516
INJURED OR FATAL	Sample Size	116463
	Difference	-21965
	Difference SE	24405
	Lower 95% Limit Cntrst Total	-71215
	Upper 95% Limit Cntrst Total	27286

Table 12. 2018 and 2019 Percentage of Injury or Fatal Crash Comparison

SUDAAN Reserved Variable One		Contrast Year to Year Comparison: (2018,2019)
Total	Sample Size	168152
	Percent	
	Difference	-0.2355
	Percent Diff SE	0.2894
	Lower 95% Limit	
	Cntrst Percent	-0.8195
	Upper 95% Limit	
	Cntrst Percent	0.3486
1	Sample Size	168152
	Percent	
	Difference	-0.2355
	Percent Diff SE	0.2894
	Lower 95% Limit	
	Cntrst Percent	-0.8195
	Upper 95% Limit	
	Cntrst Percent	0.3486

5. Summary

For a linear CRSS/FARS composite estimate such as the estimated number of police-reported crashes nationwide, the standard error estimate of the CRSS portion of the estimate is used as the standard error estimate. This approach requires commonly used statistical software such as SAS or SUDAAN to estimate the variance of the CRSS portion of the estimate. The previously published CRSS GVF method also provides stable standard error estimates for linear composite estimates and it doesn't require any specialized software.

For a non-linear CRSS/FARS composite estimate such as percentage or ratio, if the sample size is large and the fatal portion is small compared to the total, then both all-CRSS-crash standard error or non-fatal-CRSS-crash standard error estimates are good approximation. For non-linear composite estimates, CRSS GVF method are complicated if possible and may not produce real number standard error estimates.

The proposed method uses the concatenated CRSS/FARS data set under a nominal sample design. It can be used for both linear and non-linear composite estimates. It has the advantage of providing point estimates and standard estimates in a single procedure. This method requires access to specialized software such as SUDAAN, SAS, or R. SUDAAN can accommodate this nominal sample by design. It can perform many more complex analyses such as regression and multiple year comparison that SUDAAN provides. Using SAS for this method is opportunistic but simple if SAS keeps the way it currently treats the singleton PSU.

6. References

Lumley, T. (2004, May). Analysis of complex survey samples. *Journal of Statistical Software.* www.researchgate.net/publication/5142840_Analysis_of_Complex_-Survey_Samples/stats#fullTextFileContent

Lumley, T. (2020, April 3). *Analysis of complex survey samples. Package 'survey.'* https://cran.r-project.org/web/packages/survey/survey.pdf

RTI, I., & Bieler, G. (2008). *SUDAAN language manual*, release 10.0. RTI International.

SAS Institute Inc. (2017). *SAS/STAT 14.3 user's guide*. https://support.sas.com/documentation/onlinedoc/stat/143/surveymeans.pdf

Zhang, F., & Diaz, E. (2020, December). *Crash Report Sampling System: Generalized variance functions* (Report No. DOT HS 813 041). National Highway Traffic Safety Administration. https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/813041

Zhang, F., Noh, E. Y., Subramanian, R., & Chen, C.-L. (2019, May). Crash Report Sampling System: Sample design and weighting (Report No. DOT HS 812 706). Washington, DC: National Highway Traffic Safety Administration. https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812706

Zhang, F., Subramanian, R., Chen, C.-L., & Noh, E. Y. (2019, April). *Crash Report Sampling System: Design overview, analytic guidance, and FAQs* (Report No. DOT HS 812 688). National Highway Traffic Safety Administration. https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812688

Appendix A: Example Programs in SAS and SUDAAN

Example A-1: Variance Estimation of Linear Composite Estimator

```
/* Libnames used in all examples */
LIBNAME CRSS2018 "\\nhtsa-nrddata.ad.dot.gov\gesdata\CRSS\2018";
LIBNAME CRSS2019 "\\nhtsa-nrddata.ad.dot.gov\gesdata\CRSS\2019";
LIBNAME FARS2018 "\\nhtsa-nrddata.ad.dot.gov\farsdata\FARS\2018";
LIBNAME FARS2019 "\\nhtsa-nrddata.ad.dot.gov\farsdata\FARS\2019";

OPTIONS NOFMTERR LS=80 PS=100;

/* All formats */
PROC FORMAT;
      VALUE SEVERITY 1="FATAL" 2="INJURY" 3="PDO";
      VALUE INJURED 1="INJURED OR FATAL" 0="NO INJURY";
      VALUE FATAL_FLAG 1="FATAL" 0="NON-FATAL";
      VALUE DAYLIGHT 1="DAYLIGHT" 2="OTHERWISE";
      VALUE SEVERE 1="FATAL/SEVERE" 0="OTHERWISE";
      RUN;

*****************************************************************;
* Example A-1: Linear composite estimator.
*;
*****************************************************************;

/*PERMVIT: Number of People in Motor Vehicles in Transport*/
PROC MEANS DATA=FARS2018.ACCIDENT SUM;
      TITLE "FARS Count";
      VAR PERMVIT;
      RUN;

DATA PERMVIT;
      SET CRSS2018.ACCIDENT;
      KEEP PSUSTRAT PSU_VAR WEIGHT PERMVIT FATAL_FLAG;
      IF MAXSEV_IM=4 THEN FATAL_FLAG=1; ELSE FATAL_FLAG=0; /*Fatal
crash flag*/
      RUN;

PROC SURVEYMEANS DATA=PERMVIT SUM STD;
      TITLE "CRSS Estimate";
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      VAR PERMVIT;
      DOMAIN FATAL_FLAG;
      RUN;
```

Example A-2: Use the Variance Estimate of the CRSS Estimate

```
*****************************************************************;
* Example A-2: Use Variance Estimate of the All-CRSS-Crash Estimate.
*;
*****************************************************************;

/*Injury severity*/
DATA INJURED;
      SET CRSS2018.ACCIDENT (IN=CRSS2018);
      KEEP PSUSTRAT PSU_VAR WEIGHT INJURED;
      IF MAXSEV_IM IN (1,2,3,4,5) THEN INJURED=1;
      ELSE                            INJURED=0;    /*Injured or not*/
      RUN;

 PROC SURVEYFREQ DATA=INJURED;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      TABLES INJURED;
      FORMAT INJURED INJURED.;
      RUN;
```

Example A-3: Use the Variance Estimate of the Non-Fatal CRSS Estimate

```
*****************************************************************;
* Example A-3: Use Variance Estimate of CRSS Non-fatal Crash
Estimate.*;
*****************************************************************;

/*Injury severity*/
DATA SEVERITY;
      SET CRSS2018.ACCIDENT (IN=CRSS2018);
      KEEP PSUSTRAT PSU_VAR WEIGHT SEVERITY;
      IF MAXSEV_IM=4                 THEN SEVERITY=1;
      ELSE IF MAXSEV_IM IN (1,2,3,5) THEN SEVERITY=2;
      ELSE IF MAXSEV_IM IN (0,6,8)   THEN SEVERITY=3;
      RUN;

 PROC SURVEYFREQ DATA=SEVERITY;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      TABLES SEVERITY;
      FORMAT SEVERITY SEVERITY.;
      RUN;
```

Example A-4: Use CRSS GVF

```
*****************************************************************;
* Example A-4: Use CRSS GVF.                                    *;
*****************************************************************;

/*Injury severity*/
DATA CRASH_SEV_B;
      SET CRSS2018.ACCIDENT (IN=CRSS2018);
      KEEP PSUSTRAT PSU_VAR WEIGHT SEVERITY FATAL_FLAG INJURED;
      IF MAXSEV_IM=4                  THEN SEVERITY=1;
      ELSE IF MAXSEV_IM IN (1,2,3,5) THEN SEVERITY=2;
      ELSE IF MAXSEV_IM IN (0,6,8)    THEN SEVERITY=3; /*Injury
      severity*/
      IF MAXSEV_IM=4 THEN FATAL_FLAG=1;
      ELSE FATAL_FLAG=0; /*Fatal crash flag*/
      IF MAXSEV_IM IN (1,2,3,4,5) THEN INJURED=1;
      ELSE                            INJURED=0; /*Injured or not*/
      RUN;

 PROC SURVEYFREQ DATA=CRASH_SEV_B;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      TABLES SEVERITY FATAL_FLAG INJURED;
      FORMAT SEVERITY SEVERITY. FATAL_FLAG FATAL_FLAG.
             INJURED INJURED.;
      RUN;

/* GVF calculation */
data temp;
      linear=2.33242+0.31521*log(15924248)+0.02258*((log(15924248))**2)
;
      ste=exp(linear);
      R=0.286195;
      X_d=1927358;
      X_p=6734416;
      ste_X_d=88083;
      ste_X_p=333302;
      under_sqrt=(ste_X_d/X_d)**2-(ste_X_p/X_p)**2;
      *ste_R=R*sqrt((ste_X_d/X_d)**2-(ste_X_p/X_p)**2);
      run;

proc print data=temp;
      run;
```

Example A-5: Use the Proposed Method for Simple Estimates

```
*****************************************************************;
*Example A-5: Use concatenated FARS and CRSS data for simple
estimates*;
*****************************************************************;

/* Prepare all variables for SUDAAN */
DATA CRASH_SEV_C;
      SET CRSS2018.ACCIDENT (IN=CRSS2018) FARS2018.ACCIDENT
      (IN=FARS2018);
      KEEP PSUSTRAT PSU_VAR PSU_CNT WEIGHT DOMAIN_FLAG PERMVIT SEVERITY
      INJURED FATAL_FLAG DAYLIGHT VE_FORMS;
      IF FARS2018 THEN DO;
            PSUSTRAT=99;    /*Pseudo PSU stratum for FARS crashes*/
            PSU_VAR=999;    /*Pseudo PSU for FARS crashes*/
            PSU_CNT=0;      /*Pseudo PSU contributes zero variance*/
            WEIGHT=1;       /*FARS cases were selected with certainty*/
            DOMAIN_FLAG=1; /*Domain flag for FARS cases*/
            SEVERITY=1;     /*Crash severity: Fatal crash*/
            FATAL_FLAG=1;   /*Fatal flag*/
            END;
      ELSE IF CRSS2018 THEN DO;
            PSU_CNT=-1;     /*Invoke with-replacement option at CRSS PSU
            level*/
            IF MAXSEV_IM=4 THEN DOMAIN_FLAG=0; ELSE DOMAIN_FLAG=1;
            /*Domain flag for CRSS cases*/
            IF MAXSEV_IM=4 THEN SEVERITY=1; /*Fatal crash*/
            ELSE IF MAXSEV_IM IN (1,2,3,5) THEN SEVERITY=2; /*Injury
            crash*/
            ELSE IF MAXSEV_IM IN (0,6,8) THEN SEVERITY=3; /*PDO crash*/
            IF MAXSEV_IM=4 THEN FATAL_FLAG=1; ELSE FATAL_FLAG=0;
            /*Fatal flag*/
            END;
      IF SEVERITY IN (1,2) THEN INJURED=1; ELSE INJURED=0; /*Injured or
      not*/
      DAYLIGHT=(LGT_COND^=1)+1; /*1=Daylight, 2=otherwise*/
      RUN;

PROC SORT DATA=CRASH_SEV_C; BY PSU_VAR; RUN;

PROC MEANS DATA=CRASH_SEV_C NOPRINT;
      BY PSU_VAR;
      OUTPUT OUT=CRASH_CNT (DROP=_TYPE_ _FREQ_) N=CRASH_CNT;
      RUN;

DATA CRASH_SEV_D;
      MERGE CRASH_SEV_C (IN=A) CRASH_CNT (IN=B);
      BY PSU_VAR;
      IF A & B;
      RUN;
```

```
/* Reproduce Example 1 using SUDAAN*/
PROC DESCRIPT DATA=CRASH_SEV_D DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="CRSS Non-Fatal & FARS";
      VAR PERMVIT;
      PRINT NSUM="Sample Size" WSUM="Population Size"
            TOTAL="Total Estimate" SETOTAL="SE of Total"
            LOWTOTAL UPTOTAL;
      RUN;

/* Reproduce Example 2 using SUDAAN*/
PROC CROSSTAB DATA=CRASH_SEV_D DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
      TABLES INJURED;
      CLASS  INJURED;
      SETENV PAGESIZE=80 LINESIZE=70;
      PRINT NSUM="Sample Size" WSUM="Population Size"
            SEWGT="Pop Size SE" ROWPER="Percent" SEROW="Standard Error"
            / NSUMFMT=F6.0 WSUMFMT=F8.0 SEWGTFMT=F8.0;
      RUN;

/* Prepare all variables for SAS */
DATA CRASH_SEV_SAS;
      SET CRSS2018.ACCIDENT (IN=CRSS2018) FARS2018.ACCIDENT
      (IN=FARS2018);
      KEEP PSUSTRAT PSU_VAR WEIGHT DOMAIN_FLAG PERMVIT SEVERITY INJURED
      FATAL_FLAG DAYLIGHT VE_FORMS;
      IF FARS2018 THEN DO;
            PSUSTRAT=99;    /*Pseudo PSU stratum for FARS crashes*/
            PSU_VAR=999;    /*Pseudo PSU for FARS crashes*/
            WEIGHT=1;       /*FARS cases were selected with certainty*/
            DOMAIN_FLAG=1; /*Domain flag for FARS cases*/
            SEVERITY=1;    /*Crash severity: Fatal crash*/
            FATAL_FLAG=1;  /*Fatal flag*/
            END;
      ELSE IF CRSS2018 THEN DO;
            IF MAXSEV_IM=4 THEN DOMAIN_FLAG=0; ELSE DOMAIN_FLAG=1;
            /*Domain flag for CRSS cases*/
            IF MAXSEV_IM=4 THEN SEVERITY=1; /*Fatal crash*/
            ELSE IF MAXSEV_IM IN (1,2,3,5) THEN SEVERITY=2; /*Injury
            crash*/
            ELSE IF MAXSEV_IM IN (0,6,8) THEN SEVERITY=3; /*PDO crash*/
            IF MAXSEV_IM=4 THEN FATAL_FLAG=1; ELSE FATAL_FLAG=0;
            /*Fatal flag*/
            END;
      IF SEVERITY IN (1,2) THEN INJURED=1; ELSE INJURED=0; /*Injured or
```

```
      not*/
      DAYLIGHT=(LGT_COND^=1)+1; /*1=Daylight, 2=otherwise*/
      RUN;

/* Reproduce Example 1 using SAS*/
PROC SURVEYMEANS DATA=CRASH_SEV_SAS SUM STD CLSUM;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      DOMAIN DOMAIN_FLAG;
      WEIGHT WEIGHT;
      VAR PERMVIT;
      RUN;

/* Reproduce Example 2 using SAS*/
PROC SURVEYFREQ DATA=CRASH_SEV_SAS;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      TABLE DOMAIN_FLAG*INJURED / ROW;
      FORMAT INJURED INJURED.;
      RUN;
```

Example A-6: Use the Proposed Method in Logistic Regression Analysis

```
*****************************************************************;
* Example A-6: Logistic regression using the concatenated data set. *;
*****************************************************************;

/* SUDAAN procedure using the concatenated FARS and CRSS data set. */
PROC RLOGIST DATA=CRASH_SEV_D DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
      SUBGROUP DAYLIGHT;
      LEVELS 2;
      REFLEVEL DAYLIGHT=2;
      MODEL INJURED = DAYLIGHT;
      SETENV PAGESIZE=80 LINESIZE=70;
      RFORMAT DAYLIGHT DAYLIGHT.;
      RFORMAT INJURED INJURED.;
      RUN;

/* SAS procedure using the concatenated FARS and CRSS data set. */
PROC SURVEYLOGISTIC DATA=CRASH_SEV_SAS VARMETHOD=TAYLOR;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      CLASS DAYLIGHT / REF=LAST PARAM=REFERENCE;
      DOMAIN DOMAIN_FLAG;
      MODEL INJURED (EVENT='1') = DAYLIGHT;
      RUN;

/* SUDAAN procedure using only CRSS data set. */
DATA CRSS2018;
      SET CRSS2018.ACCIDENT;
      KEEP PSUSTRAT PSU_VAR WEIGHT INJURED DAYLIGHT;
    IF MAXSEV_IM IN (1,2,3,4,5) THEN INJURED=1;
      ELSE INJURED=0; /*Injured or not*/
      DAYLIGHT=(LGT_COND^=1)+1; /*1=Daylight, 2=Otherwise*/
      RUN;

PROC RLOGIST DATA=CRSS2018 NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      WEIGHT WEIGHT;
      SUBGROUP DAYLIGHT;
      LEVELS 2;
      REFLEVEL DAYLIGHT=2;
      MODEL INJURED = DAYLIGHT;
      SETENV PAGESIZE=80 LINESIZE=70;
      PRINT / BETAFMT=F8.4 SEBETAFMT=F8.4;
      RFORMAT DAYLIGHT DAYLIGHT.;
      RFORMAT INJURED INJURED.;
```

```
      RUN;

/* SAS procedure using only CRSS data set. */
PROC SURVEYLOGISTIC DATA=CRSS2018;
      STRATA PSUSTRAT;
      CLUSTER PSU_VAR;
      WEIGHT WEIGHT;
      CLASS DAYLIGHT / REF=LAST PARAM=REFERENCE;
      MODEL INJURED (EVENT='1') = DAYLIGHT;
      RUN;
```

Example A-7: Use the Proposed Method for Multiple Year Comparison

```
*****************************************************************;
* Example A-7: Year-to-year comparison using concatenated data set. *;
*****************************************************************;

/* Prepare all variables */
DATA CRASH_SEV_E;
SET   CRSS2018.ACCIDENT (IN=CRSS2018) CRSS2019.ACCIDENT (IN=CRSS2019)
      FARS2018.ACCIDENT (IN=FARS2018) FARS2019.ACCIDENT (IN=FARS2019);
      KEEP PSUSTRAT PSU_VAR PSU_CNT WEIGHT DOMAIN_FLAG YEAR INJURED;
      YEAR = 2018*SUM(CRSS2018,FARS2018) + 2019*SUM(CRSS2019,FARS2019);
      IF (FARS2018 OR FARS2019) THEN DO;
      PSUSTRAT=99;    /*Pseudo PSU stratum for FARS crashes*/
      PSU_VAR=999;    /*Pseudo PSU for FARS crashes*/
      PSU_CNT=0;      /*Pseudo PSU contributes zero variance*/
      WEIGHT=1;       /*FARS cases were selected with certainty*/
      DOMAIN_FLAG=1; /*Domain flag for FARS cases*/
      INJURED=1;    /*Injury crashes*/
      END;
      ELSE IF (CRSS2018 OR CRSS2019) THEN DO;
      PSU_CNT=-1;     /*Invoke with-replacement option at CRSS PSU
      level*/
      IF MAXSEV_IM=4 THEN DOMAIN_FLAG=0; ELSE DOMAIN_FLAG=1; /*Domain
      flag for CRSS cases*/
      IF MAXSEV_IM IN (1,2,3,4,5) THEN INJURED=1; ELSE INJURED=0;
      /*Injured or not*/
      END;
      RUN;

/* Calculate two year total estimates */
PROC CROSSTAB DATA=CRASH_SEV_F DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
      CLASS YEAR INJURED;
      TABLES YEAR*INJURED;
      SETENV ROWWIDTH=15 COLWIDTH=15 LABWIDTH=25;
      PRINT  NSUM="Sample Size" WSUM="Population Size"
             SEWGT="Pop Size SE" ROWPER="Percent"
      / NSUMFMT=F8.0 WSUMFMT=F10.0 SEWGTFMT=F9.0 ROWPERFMT=F6.4;
      RFORMAT INJURED INJURED.;
      RUN;

/* Compare multiple year totals */
PROC DESCRIPT DATA=CRASH_SEV_F DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
```

```
      CLASS YEAR INJURED;
      TABLES INJURED;
      VAR _ONE_;
      PAIRWISE YEAR / NAME="Year to Year Comparison";
      SETENV ROWWIDTH=15 COLWIDTH=15 LABWIDTH=30;
      PRINT NSUM="Sample Size" TOTAL="Difference"
            SETOTAL="Difference SE" LOWTOTAL UPTOTAL
            / NSUMFMT=F6.0 TOTALFMT=F12.0 SETOTALFMT=F12.0
              LOWTOTALFMT=F12.0 UPTOTALFMT=F12.0;
      RFORMAT INJURED INJURED.;
      RUN;

/* Compare multiple year percentages */
PROC DESCRIPT DATA=CRASH_SEV_F DESIGN=WOR NOTSORTED;
      NEST PSUSTRAT PSU_VAR;
      TOTCNT PSU_CNT CRASH_CNT;
      WEIGHT WEIGHT;
      SUBPOPN DOMAIN_FLAG=1 / NAME="FARS & CRSS Non-Fatal";
      CLASS YEAR INJURED;
      VAR INJURED;
      CATLEVEL 1;
      PAIRWISE YEAR / NAME="Year to Year Comparison";
      SETENV ROWWIDTH=20 COLWIDTH=40 LABWIDTH=30;
      PRINT NSUM="Sample Size" PERCENT="Percent Difference"
            SEPERCENT="Percent Diff SE" LOWPCT UPPCT
            / NSUMFMT=F20.0 PERCENTFMT=F6.4 SEPERCENTFMT=F6.4
              LOWPCTFMT=F6.4 UPPCTFMT=F6.4;
      RFORMAT INJURED INJURED.;
      RUN;
```

Appendix B: Example Programs and Output in R

Load libraries and global options.

```
library(haven)
library(survey)
library(srvyr)

options(scipen = 999) #disable scientific notation
```

Load SAS file and variables needed for analysis.

```
FARS2018 <- read_sas("L:/FARS/2018/accident.sas7bdat", col_select = c(PERMVIT
, VE_FORMS, LGT_COND))

FARS2019 <- read_sas("L:/FARS/2019/accident.sas7bdat", col_select = c(PERMVIT
, VE_FORMS, LGT_COND))

CRSS2018 <- read_sas("R:/CRSS/2018/accident.sas7bdat", col_select = c(PSUSTRA
T, PSU_VAR, WEIGHT, MAXSEV_IM, PERMVIT, VE_FORMS, LGT_COND))

CRSS2019 <- read_sas("R:/CRSS/2019/accident.sas7bdat", col_select = c(PSUSTRA
T, PSU_VAR, WEIGHT, MAXSEV_IM, PERMVIT, VE_FORMS, LGT_COND))
```

Example B-1: Variance Estimation of Linear Composite Estimator

Example B-1: Linear composite estimator.

PERMVIT: Number of People in Motor Vehicle in Transport.

FARS total number of people in motor vehicles in transport.

```
tfars <- sum(FARS2018$PERMVIT)
tfars
```

```
## [1] 76036
```

CRSS estimated total number of people in motor vehicles in transport.

```
PERMVIT <- CRSS2018

PERMVIT$FATAL_FLAG <- as.numeric(as.character(ifelse(PERMVIT$MAXSEV_IM == 4,
0, 1)))

keep <- c("PSUSTRAT", "PSU_VAR", "WEIGHT", "PERMVIT", "FATAL_FLAG")
PERMVIT <- PERMVIT[keep]

PERMVIT_sd <- svydesign(
  ids = ~ PSU_VAR,
  strata = ~ PSUSTRAT,
  weights = ~ WEIGHT,
  data = PERMVIT,
  nest = TRUE
)

PERMVIT_dom_sd <- subset(PERMVIT_sd, FATAL_FLAG)

tcrss <- svytotal(~PERMVIT, PERMVIT_dom_sd)
tcrss
```

```
##            total     SE
## PERMVIT 15924248 857870
```

Estimated total number of people in motor vehicles in transport.

```
tc <- tfars + tcrss[1]
tc
```

```
##  PERMVIT
## 16000284
```

Example B-2: Use the Variance Estimate of the CRSS Crash Estimate

Example B-2: Use Variance Estimate of the All-CRSS-CRASH Estimate.

Injury severity

```
INJURED <- CRSS2018
INJURED$INJURED <- as.factor(ifelse(INJURED$MAXSEV_IM %in% c(1:5),1, 0))
INJURED$INJURED <- factor(INJURED$INJURED, levels = c(1, 0), labels = c("Inju
red or Fatal", "No Injury"))

keep <- c("PSUSTRAT", "PSU_VAR", "WEIGHT", "INJURED")
INJURED <- INJURED[keep]

INJURED_sd <- svydesign(
  ids = ~ PSU_VAR,
  strata = ~ PSUSTRAT,
  weights = ~ WEIGHT,
  data = INJURED,
  nest = TRUE
)

svymean(~INJURED, design = INJURED_sd)
```

```
##                             mean     SE
## INJUREDInjured or Fatal 0.2862 0.0075
## INJUREDNo Injury        0.7138 0.0075
```

```
INJURED_sd %>%
  as_survey_design() %>%
  summarise(Total = survey_total())
```

```
##     Total Total_se
## 1 6734416 333709.3
```

Example B-3: Use the Variance Estimate of the Non-Fatal CRSS Estimate

Example B-3: Use Variance Estimate of CRSS Non-fatal Crash Estimate.

Injury severity

```
SEVERITY <- CRSS2018
SEVERITY$SEVERITY <- as.factor(ifelse(SEVERITY$MAXSEV_IM == 4, 1,
                              ifelse(SEVERITY$MAXSEV_IM %in% c(1:3, 5), 2, 3)))
SEVERITY$SEVERITY <- factor(SEVERITY$SEVERITY, levels = c(1:3), labels = c("F
atal", "Injury", "PDO"))

keep <- c("PSUSTRAT", "PSU_VAR", "WEIGHT", "SEVERITY")
SEVERITY <- SEVERITY[keep]

SEVERITY_sd <- svydesign(
  ids = ~ PSU_VAR,
  strata = ~ PSUSTRAT,
  weights = ~ WEIGHT,
  data = SEVERITY,
  nest = TRUE
)

svymean(~SEVERITY, SEVERITY_sd)
```

```
##                       mean     SE
## SEVERITYFatal  0.0049973 0.0004
## SEVERITYInjury 0.2811980 0.0074
## SEVERITYPDO    0.7138047 0.0075
```

Example B-4: Use CRSS GVF

Example B-4: Use CRSS GVF.

Injury severity

```
CRASH_SEV_B <- CRSS2018
CRASH_SEV_B$CRASH_SEV <- ifelse(CRASH_SEV_B$MAXSEV_IM == 4, 1,
                                ifelse(CRASH_SEV_B$MAXSEV_IM %in% c(1:3, 5),
2, 3))
CRASH_SEV_B$CRASH_SEV <- factor(CRASH_SEV_B$CRASH_SEV, levels = c(1:3), label
s = c("Fatal", "Injury", "PDO"))

CRASH_SEV_B$FATAL_FLAG <- ifelse(CRASH_SEV_B$MAXSEV_IM == 4, 1, 0)
CRASH_SEV_B$FATAL_FLAG <- factor(CRASH_SEV_B$FATAL_FLAG, levels = c(1, 0), la
bels = c("Fatal", "Non-Fatal"))

CRASH_SEV_B$INJURED <- ifelse(CRASH_SEV_B$MAXSEV_IM %in% c(1:5), 1, 0)
CRASH_SEV_B$INJURED <- factor(CRASH_SEV_B$INJURED, levels = c(1, 0), labels =
c("Injured or Fatal", "No Injury"))

keep <- c("PSUSTRAT", "PSU_VAR", "WEIGHT", "CRASH_SEV", "FATAL_FLAG", "INJURE
D")
CRASH_SEV_B <- CRASH_SEV_B[keep]

CRASH_SEV_B_sd <- svydesign(
  ids = ~ PSU_VAR,
  strata = ~ PSUSTRAT,
  weights = ~ WEIGHT,
  data = CRASH_SEV_B,
  nest = TRUE
)
svytotal(~CRASH_SEV, CRASH_SEV_B_sd)
```

```
##                   total     SE
## CRASH_SEVFatal    33654   2180
## CRASH_SEVInjury 1893704  88083
## CRASH_SEVPDO    4807058 262531
```

```
svymean(~CRASH_SEV, CRASH_SEV_B_sd)
```

```
##                       mean     SE
## CRASH_SEVFatal  0.0049973 0.0004
## CRASH_SEVInjury 0.2811980 0.0074
## CRASH_SEVPDO    0.7138047 0.0075
```

```
svytotal(~FATAL_FLAG, CRASH_SEV_B_sd)
```

```
##                         total     SE
## FATAL_FLAGFatal        33654   2180
## FATAL_FLAGNon-Fatal 6700762 333302
```

```
svymean(~FATAL_FLAG, CRASH_SEV_B_sd)
```

```
##                            mean     SE
## FATAL_FLAGFatal     0.0049973 0.0004
## FATAL_FLAGNon-Fatal 0.9950027 0.0004
```

```
svytotal(~INJURED, CRASH_SEV_B_sd)
```

```
##                             total     SE
## INJUREDInjured or Fatal 1927358  88421
## INJUREDNo Injury        4807058 262531
```

```
svymean(~INJURED, CRASH_SEV_B_sd)
```

```
##                             mean     SE
## INJUREDInjured or Fatal 0.2862 0.0075
## INJUREDNo Injury        0.7138 0.0075
```

```
CRASH_SEV_B_sd %>%
  as_survey_design() %>%
  summarise(Total = survey_total())
```

```
##     Total Total_se
## 1 6734416 333709.3
```

GVF Calculation

```
linear <- 2.33242 + (0.31521 * log(15924248)) + (0.02258 * ((log(15924248))^2
))
ste <- exp(linear)
ste
```

```
## [1] 954870.3
```

```
R <- 0.286195
X_d <- 1927358
X_p <- 6734416
ste_X_d <- 88083
ste_X_p <- 333302
under_sqrt <- (ste_X_d / X_d)^2 - (ste_X_p / X_p)^2
under_sqrt
```

```
## [1] -0.0003608717
```

```
#ste_R <- R * sqrt(under_sqrt)
```

Example B-5: Use the Proposed Method for Simple Estimates

Example B-5: Use concatenated FARS and CRSS data for simple estimates.

Prepare all variables

```
fars18 <- FARS2018

n <- dim(fars18)[1]

fars18$PSUSTRAT <- rep(99, n) #Pseudo PSU stratum for FARS crashes
fars18$PSU_VAR <- rep(999, n) #Pseudo PSU for FARS crashes
fars18$WEIGHT <- rep(1, n) #FARS cases were selected with certainty
fars18$DOMAIN_FLAG <- rep(1, n) #Domain flag for FARS cases
fars18$CRASH_SEV <- rep(1, n) #Crash severity: Fatal crash
fars18$FATAL_FLAG <- rep(1, n) #Fatal flag
fars18$INJURED <- ifelse(fars18$CRASH_SEV %in% c(1:2), 1, 0) #Injured or not
fars18$DAYLIGHT <- ifelse(fars18$LGT_COND == 1, 1, 2) #1 = Daylight, 2 = othe
rwise

crss18 <- CRSS2018

n <- dim(crss18)[1]

crss18$DOMAIN_FLAG <- ifelse(crss18$MAXSEV_IM == 4, 0, 1) #Domain flag for CR
SS cases
crss18$CRASH_SEV <- ifelse(crss18$MAXSEV_IM == 4, 1,
                           ifelse(crss18$MAXSEV_IM %in% c(1:3, 5), 2, 3)) #1
= Fatal crash, 2 = Injury crash, 3 = PDO crash
crss18$FATAL_FLAG <- ifelse(crss18$MAXSEV_IM == 4, 1, 0) #Fatal flag
crss18$INJURED <- ifelse(crss18$CRASH_SEV %in% c(1:2), 1, 0) #Injured or not
crss18$DAYLIGHT <- ifelse(crss18$LGT_COND == 1, 1, 2) #1 = Daylight, 2 = othe
rwise

keep <- c("PSUSTRAT", "PSU_VAR", "WEIGHT", "DOMAIN_FLAG", "PERMVIT", "CRASH_S
EV", "INJURED", "FATAL_FLAG", "DAYLIGHT", "VE_FORMS")
fars18 <- fars18[keep]
crss18 <- crss18[keep]

CRASH_SEV_C <- rbind(fars18, crss18)
```

Reproduce Example 1 using R

```
options(survey.lonely.psu = "remove")

CRASH_SEV_C_sd <- svydesign(
  ids = ~ PSU_VAR,
  strata = ~ PSUSTRAT,
  weights = ~ WEIGHT,
  data = CRASH_SEV_C,
  nest = TRUE
)
```

```
CRASH_SEV_C_dom_sd <- subset(CRASH_SEV_C_sd, DOMAIN_FLAG)

svytotal(~PERMVIT, CRASH_SEV_C_dom_sd)
##            total     SE
## PERMVIT 16000284 857870
```

Reproduce Example 2 using R

```
svymean(~INJURED, design = CRASH_SEV_C_dom_sd)
##            mean     SE
## INJURED 0.28622 0.0075
```

Example B-6: Use the Proposed Method in Logistic Regression Analysis

Example B-6: Logistic regression using the concatenated dataset.

R procedure using the concatenated FARS and CRSS data set.

```
confarscrss_glm <- summary(svyglm(I(INJURED == 1) ~ I(DAYLIGHT == 1), CRASH_S
EV_C_dom_sd, family = quasibinomial))

confarscrss_glm
```

```
##
## Call:
## svyglm(formula = I(INJURED == 1) ~ I(DAYLIGHT == 1), design = CRASH_SEV_C_
dom_sd,
##     family = quasibinomial)
##
## Survey design:
## subset(CRASH_SEV_C_sd, DOMAIN_FLAG)
##
## Coefficients:
##                       Estimate Std. Error t value            Pr(>|t|)
## (Intercept)           -0.86954    0.04507 -19.294 <0.0000000000000002 ***
## I(DAYLIGHT == 1)TRUE -0.06454    0.04107  -1.572               0.124
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasibinomial family taken to be 1.000012)
##
## Number of Fisher Scoring iterations: 4
```

```
# Odds ratio
exp(coef(confarscrss_glm)[2])
```

```
## [1] 0.9374982
```

R procedure using only CRSS data set.

```
CRASH_SEV_C_CRSS_sd <- svydesign(
  ids = ~ PSU_VAR,
  strata = ~ PSUSTRAT,
  weights = ~ WEIGHT,
  data = crss18,
  nest = TRUE
)

crss_glm <- summary(svyglm(I(INJURED == 1) ~ I(DAYLIGHT == 1), CRASH_SEV_C_CR
SS_sd, family = quasibinomial))

crss_glm
```

```
##
## Call:
## svyglm(formula = I(INJURED == 1) ~ I(DAYLIGHT == 1), design = CRASH_SEV_C_
```

```
CRSS_sd,
##     family = quasibinomial)
##
## Survey design:
## svydesign(ids = ~PSU_VAR, strata = ~PSUSTRAT, weights = ~WEIGHT,
##     data = crss18, nest = TRUE)
##
## Coefficients:
##                       Estimate Std. Error t value            Pr(>|t|)
## (Intercept)           -0.87083    0.04516 -19.281 <0.0000000000000002 ***
## I(DAYLIGHT == 1)TRUE -0.06284    0.04131  -1.521               0.136
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasibinomial family taken to be 1.000021)
##
## Number of Fisher Scoring iterations: 4
```

```
# Odds ratio
exp(coef(crss_glm)[2])
```

```
## [1] 0.9390932
```

Example B-7: Use the Proposed Method for Multiple Year Comparison

Example B-7: Year-to-year comparison using concatenated data set.

Prepare all variables

```
f18 <- FARS2018
n <- dim(f18)[1]
f18$YEAR <- rep(2018, n)
f18$PSUSTRAT <- rep(99, n) #Pseudo PSU stratum for FARS
f18$PSU_VAR <- rep(999, n) #Pseudo PSU for FARS crashes
f18$PSU_CNT <- rep(0, n) #Pseudo PSU contributes zero variance
f18$WEIGHT <- rep(1, n) #FARS cases were selected with certainty
f18$DOMAIN_FLAG <- rep(1, n) #Domain flag for FARS cases
f18$INJURED <- rep(1, n) #Injury crashes
f18$INJURED <- factor(f18$INJURED, levels = 1, labels = c("Injured or Fatal")
)


f19 <-FARS2019
n <- dim(f19)[1]
f19$YEAR <- rep(2019, n)
f19$PSUSTRAT <- rep(99, n) #Pseudo PSU stratum for FARS
f19$PSU_VAR <- rep(999, n) #Pseudo PSU for FARS crashes
f19$PSU_CNT <- rep(0, n) #Pseduo PSU contributes zero variance
f19$WEIGHT <- rep(1, n) #FARS cases were selected with certainty
f19$DOMAIN_FLAG <- rep(1, n) #Domain flage for FARS cases
f19$INJURED <- rep(1, n) #Injury crashes
f19$INJURED <- factor(f19$INJURED, levels = 1, labels = c("Injured or Fatal")
)


c18 <- CRSS2018
n <- dim(c18)[1]
c18$YEAR <- rep(2018, n)
c18$PSU_CNT <- rep(-1, n) #Invoke with-replacement option at CRSS PSU level
c18$DOMAIN_FLAG <- ifelse(c18$MAXSEV_IM == 4, 0, 1) #Domain flag for CRSS cas
es
c18$INJURED <- ifelse(c18$MAXSEV_IM %in% c(1:5), 1, 0)
c18$INJURED <- factor(c18$INJURED, levels = c(1, 0), labels = c("Injured or F
atal", "No Injury"))


c19 <- CRSS2019
n <- dim(c19)[1]
c19$YEAR <- rep(2019, n)
c19$PSU_CNT <- rep(-1, n) #Invoke with-replacement option at CRSS PSU level
c19$DOMAIN_FLAG <- ifelse(c19$MAXSEV_IM == 4, 0, 1) #Domain flag for CRSS cas
es
c19$INJURED <- ifelse(c19$MAXSEV_IM %in% c(1:5), 1, 0)
c19$INJURED <- factor(c19$INJURED, levels = c(1, 0), labels = c("Injured or F
```

```
atal", "No Injury"))

keep <- c("PSUSTRAT", "PSU_VAR", "PSU_CNT", "WEIGHT", "DOMAIN_FLAG", "YEAR",
"INJURED")

datasets <- list(f18, f19, c18, c19)

f18 <- f18[keep]
f19 <- f19[keep]
c18 <- c18[keep]
c19 <- c19[keep]

CRASH_SEV_E <- rbind(f18, f19, c18, c19)

CRASH_CNT <- as.data.frame(table(CRASH_SEV_E$PSU_VAR))
colnames(CRASH_CNT) <- c("PSU_VAR", "CRASH_CNT")

CRASH_SEV_F <- merge(CRASH_SEV_E, CRASH_CNT)
```

Calculate two year total estimates

```
CRASH_SEV_F_sd <- svydesign(
  ids = ~ PSU_VAR,
  strata = ~ PSUSTRAT,
  weights = ~ WEIGHT,
  data = CRASH_SEV_F,
  nest = TRUE
)

CRASH_SEV_F_dom_sd <- subset(CRASH_SEV_F_sd, DOMAIN_FLAG)

# Frequency and standard error for YEAR and INJURED interaction
a <- svytotal(~ interaction(YEAR, factor(INJURED)), CRASH_SEV_F_dom_sd)
a
```

```
##                                                              total     SE
## interaction(YEAR, factor(INJURED))2018.Injured or Fatal 1927623  88083
## interaction(YEAR, factor(INJURED))2019.Injured or Fatal 1949588  88907
## interaction(YEAR, factor(INJURED))2018.No Injury        4807058 262531
## interaction(YEAR, factor(INJURED))2019.No Injury        4806253 272166
```

```
# Overall frequency and standard error by INJURED
b <- svytotal(~ factor(INJURED), CRASH_SEV_F_dom_sd)
b
```

```
##                                    total     SE
## factor(INJURED)Injured or Fatal 3877211 175301
## factor(INJURED)No Injury        9613311 531143
```

```
# Row percents for YEAR and INJURED interaction
c <- svyby(~ factor(INJURED), by = ~ YEAR, design = CRASH_SEV_F_dom_sd, FUN =
```

```
svymean, covmat = TRUE)
c
```

```
##      YEAR factor(INJURED)Injured or Fatal factor(INJURED)No Injury
## 2018 2018                       0.2862234                0.7137766
## 2019 2019                       0.2885781                0.7114219
##      se.factor(INJURED)Injured or Fatal se.factor(INJURED)No Injury
## 2018                        0.007526611                0.007526611
## 2019                        0.008424254                0.008424254
```

```
# Overall percent by INJURED
N <- b[1] + b[2]
## INJURED = 0: No Injury
b[1] / N
```

```
## factor(INJURED)Injured or Fatal
##                       0.2874026
```

```
## INJURED  1: Injured
b[2] / N
```

```
## factor(INJURED)No Injury
##                 0.7125974
```

```
# Total Frequency and standard error by YEAR
d <- svytotal(~ factor(YEAR), CRASH_SEV_F_dom_sd)
d
```

```
##                     total     SE
## factor(YEAR)2018 6734681 333302
## factor(YEAR)2019 6755841 340199
```

```
# Total Frequency and standard error
CRASH_SEV_F_dom_sd %>%
  as_survey_design() %>%
  summarise(Total = survey_total())
```

```
##      Total Total_se
## 1 13490522 669283.6
```

Compare multiple year totals

```
#Calculate df: Number of PSUs - Number of strata
df <- length(unique(CRASH_SEV_F$PSU_VAR)) - length(unique(CRASH_SEV_F$PSUSTRA
T))

#Total
nii <- svycontrast(d, list(diff = c(1, -1)))
nii
```

```
##      contrast    SE
## diff   -21160 75569
```

```
confint(nii, df = df)
```

```
##        2.5 % 97.5 %
## diff -173664 131345
```

```r
#No Injury
ni <- svycontrast(a, list(diff = c(0, 0, 1, -1)))
ni
```

```
##      contrast    SE
## diff   805.14 62292
```

```r
confint(ni, df = df)
```

```
##          2.5 %   97.5 %
## diff -124905.6 126515.9
```

```r
#Injured or fatal
i <- svycontrast(a, list(diff = c(1, -1, 0, 0)))
i
```

```
##      contrast    SE
## diff   -21965 24405
```

```r
confint(i, df = df)
```

```
##          2.5 %   97.5 %
## diff -71215.22 27285.99
```

Compare multiple year percentages

```r
e <- svycontrast(c, list(diff = c(0, 0, -1, 1)))
e
```

```
##        contrast     SE
## diff -0.0023547 0.0029
```

```r
confint(e, df = df)
```

```
##             2.5 %      97.5 %
## diff -0.008195041 0.003485552
```

DOT HS 813 225
December 2021

U.S. Department
of Transportation

**National Highway
Traffic Safety
Administration**

15444-120221-v2