

---

# Data analysis on the CT Crash Dataset

---

Jianmin Chen, Jun Jin

December 5, 2019

## CONTENTS

<b>1 Abstract</b>	<b>2</b>
<b>2 Introduction and Description of Data</b>	<b>2</b>
2.1 Research proposing . . . . .	2
2.2 Qualitative and Quantitative . . . . .	2
<b>3 Exploratory Data Analysis</b>	<b>3</b>
3.1 Overall Distribution . . . . .	3
3.2 Crash vs Qualitative Variables . . . . .	5
3.2.1 Lighting . . . . .	5
3.2.2 Approach Left Turn and Approach Right Turn . . . . .	5
3.3 Crash vs Quantitative Variables . . . . .	6
3.3.1 AADT-MAJOR and AADT-MINOR . . . . .	6
3.3.2 Skew Angle . . . . .	7
3.4 EDA Conclusion . . . . .	8
<b>4 Statistical Inference</b>	<b>8</b>
4.1 One v.s. PDO crashing independence test . . . . .	8
4.2 One v.s. BC and KA crashing independence test . . . . .	10
4.3 Two v.s. crashing independence test . . . . .	12
4.3.1 Crossing type + Skewed Angle v.s. Crashing . . . . .	12
4.3.2 Crossing type + Lighting vs Crashing . . . . .	13
4.3.3 Crossing type + turns for Turn vs Crashing . . . . .	13
4.3.4 Crossing type + AADT major VS Crashing . . . . .	14
4.3.5 Crossing type + AADT minor vs Crashing . . . . .	15
<b>5 Discussion of Results and Concluding Remarks</b>	<b>16</b>
5.1 Conclusions . . . . .	16
5.2 Further Discussion . . . . .	16
5.2.1 Logistic regression . . . . .	16
5.2.2 Sufficient dimension reduction . . . . .	17

<b>6 Appendix</b>	<b>18</b>
6.1 Extra Plots . . . . .	18
6.2 R codes . . . . .	22
6.2.1 EDA . . . . .	22
6.2.2 Inference . . . . .	30
6.2.3 Python - PSVM code (Implementing file) . . . . .	59
6.2.4 Python - PSVM code (Support file) . . . . .	60

## 1 ABSTRACT

In this paper, the group conducts the exploratory data analysis and statistical inferences on the CT crash data, judging that whether the variables such as type of intersections, presence of lighting, presence of a left-turn lane, presence of a right-turn lane and others could post great effects on the crash counts and exploring that how they work. The research might be useful for the government as a reference when designing the intersections in Connecticut.

## 2 INTRODUCTION AND DESCRIPTION OF DATA

In this paper, the group will take advantage of "CT Crash Data", which is the crash data for rural two-lane intersections and segments in Connecticut are collected in order to study highway safety. There 3 types of crash categories: PDO (property damage only), or B+C (possible injury or nonincapacitating injury), or K+A (fatal or incapacitating injury) crashes. And the intersection is also aggregated into 3 types with one dataset for each. Dataset 3ST Intersections CT.csv consists of  $n = 385$  cases for 3ST (three-way stop controlled) intersections. Dataset 4ST Intersections CT.csv consists of  $n = 61$  cases for 4ST (four-way stop controlled) intersections. Dataset 4SG Intersections CT.csv consists of  $n = 102$  cases for 4SG (four-way signalized) intersections. Each dataset includes the following variables: Intersection ID, Annual Average Daily Traffic (AADT) for major roads and minor roads, three dummy variables to indicate presence of lighting, presence of a left-turn lane, and presence of a right-turn lane, followed by skew angle, and then the counts of PDO, KA, and BC crashes. We combine the three data sets together for our analysis.

### 2.1 Research proposing

In this paper, we are going to explore and confirm the following questions:

- Do these different types of intersection affect the number of crashes? (Regardless of other features.)
- If we control the types of intersection, whether other features (e.g. skewed angle, lighting, approaching turns, AADT major and minor) affect the number of crashes?
- And we may want to explore what kind of relationship (negative or positive) between the features we mentioned previously with the number of crashes.

### 2.2 Qualitative and Quantitative

Qualitative: Lighting, Approach-Left, Approach-Right.

Quantitative: AADT major, AADT minor, Skew Angle, PDO crashing number, KA crashing number, BC crashing number.

### 3 EXPLORATORY DATA ANALYSIS

We first rearranged the 3 data set and combined them into 1 single data set. One column is added to specify the intersection type, denoted as INTERTYPE. It has 3 values, "3ST", "4ST" and "4SG". Then we have 548 observations in total.

A new dummy variable is constructed with "1" standing for there exists crash and "0" standing for no crash. This variable is denoted as "crash" and it is used as a candidate response variable when we want to analyze the data with mixed types of crash.

In this part, we will do exploratory data analysis to first find some insight of the data.

#### 3.1 Overall Distribution

First, let us see overall distribution of the 3 types of crashes and the distribution across intersection types.

For the PDO(property damage only) crashes, overall, around 38% of the intersections have 0 PDO crash. The overall distribution of PDO is spread out with some extreme values larger than 15. The distribution across different intersections shows some variations. The distribution is similar to the overall distribution in the 3ST type, while in 4SG and 4ST types, there are fewer zero counts and the data has larger variance with more non-zero counts had hence heavier tails. These infer that 4-way intersections tend to have more PDO crashes, especially the 4-way signalized intersection.

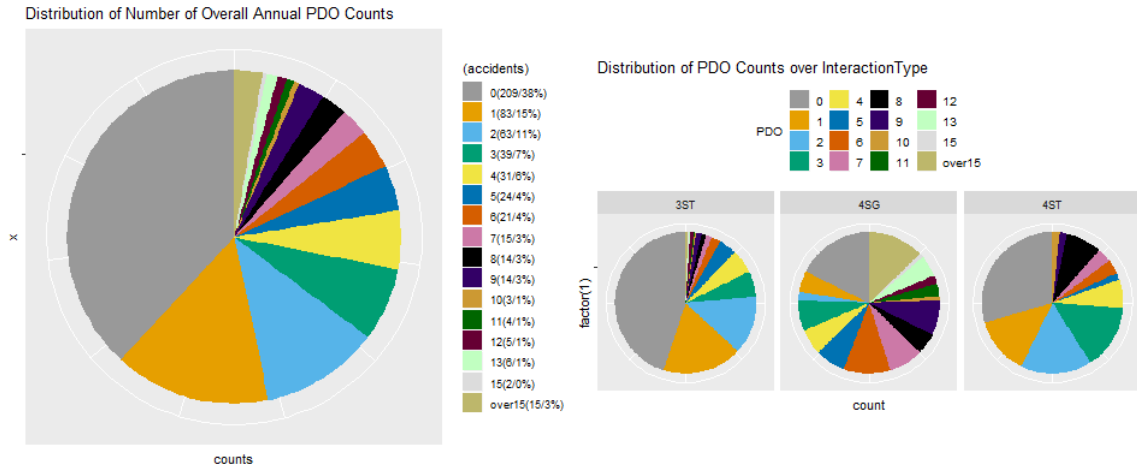


Figure 3.1: Pie chart of PDO

For the BC(possible injury or non-incapacitating injury) crashes, over one-half(59%) of the intersections have 0 crash and 76% of all intersections have less than 2 crashes. Like PDO, 4-way intersections tend to have larger number of BC crashes.

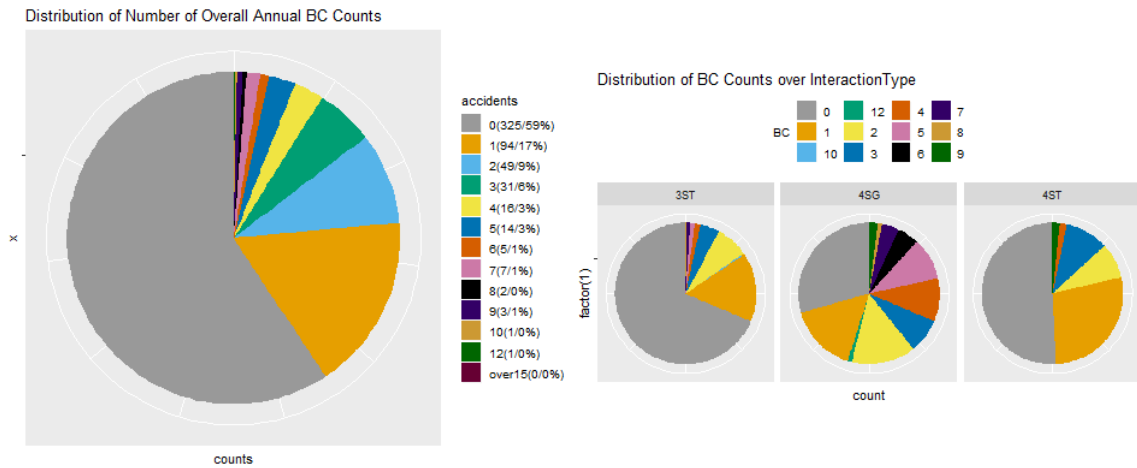


Figure 3.2: Pie chart of BC

For the KA(fatal or incapacitating injury) crashes, only 0,1 and 2 crashes are observed. We can infer that Poisson or Negative Binomial distribution may not be suitable to fit the KA data. Quite similar to the conclusion above, 4SG and 4ST types have larger proportion of intersection with crashes.

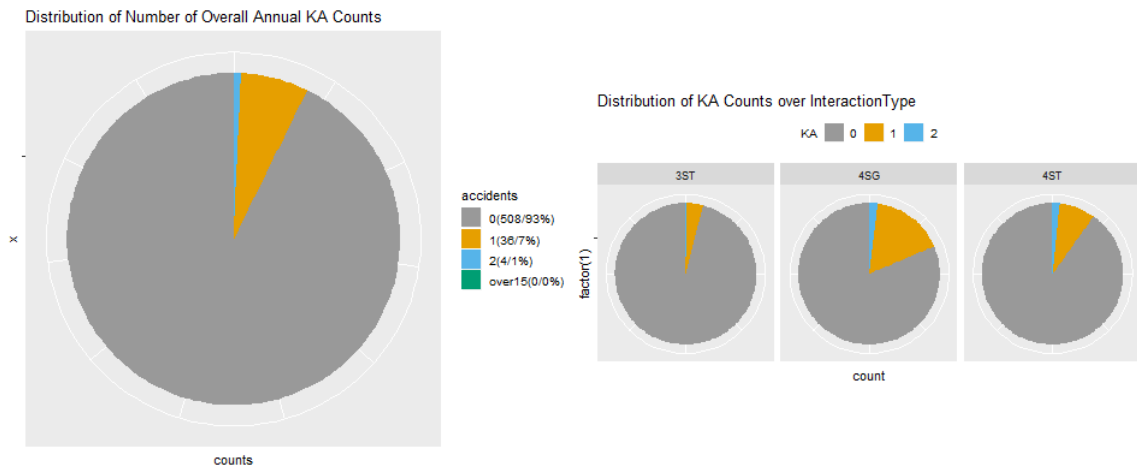


Figure 3.3: Pie chart of KA

Comparing the overall distribution of 3 types of crashes, we can see that for all 3 types, 0 crash has the maximum proportion in all 548 intersection. PDO and BC have more possible count values with larger dispersions, which is a hint that it is not plausible to mix PDO, KA and BC observations together when we want to model the distribution of crashes. Also, variance exists between types of intersections. Since intersection type is a more fixed factor that is hard to change in real traffic, in the following parts, we will analyze the data controlling intersection type to get some plausible suggestions to reduce crashes based on other variables.

### 3.2 Crash vs Qualitative Variables

Next, we find some interesting relationship between crash and the 3 categorical variables, LIGHTING, APPROACH-LEFTTURN, and APPROACH-RIGHTTURN.

#### 3.2.1 Lighting

First, let's see the relationship between LIGHTING and crashes. There are 197 intersections without lighting in all 548(around 36%) intersections. Overall, when not controlling intersection type, intersection without lighting tends to have fewer crashes than intersection with lighting in all 3 types of crashes. We can compare the proportion of 0 counts in the following bar chart of PDO. The bar chart of BC and KA has similar tendency which are contained in Appendix1.

When controlling intersection type, there are some interesting findings.

For PDO, in 3ST and 4SG types, partitioned data has same tendency with whole data while for 4ST group, there is lower proportion of observations with crash when there is lighting, which is opposite to the whole data. For BC and KA, partitioning with intersection type generates same result in each group as the whole set.

The plots infer that there might be a relationship between lighting and crash, but the effect is not very obvious from the plots. We will not take 4ST PDO group out for special consideration as these group has a quite small number of observations which may cause larger randomness.

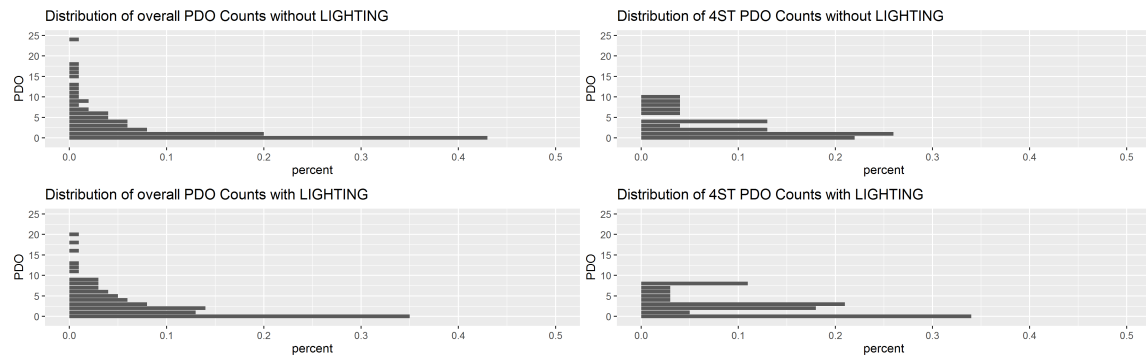


Figure 3.4: Bar Chart of Lighting

#### 3.2.2 Approach Left Turn and Approach Right Turn

There are only 65 intersections with approach left-turn. When partitioning with intersection type, we get the following contingency table. From the table we can see that the counts get even smaller when separated into 3 type, there is not much sense to analyze with data further separated by crash type. There is similar problem with approach right-turn. Hence, we will not control crash type here and only care whether there is crash or not. So later discussion will be based on the generated binary columns "crash".

We observed similar association between these 2 variables and crash. Overall, when not controlling intersection type, intersection with approach left-turn or right turn lane has a higher crash rate, which infers that approach turn lane may has a positive relationship with car crashes. However, when we look into data grouped by intersection type, it is shown that approach turn lane does not have big effect with 3ST and 4SG intersections, even shown slightly negative relationship. The effect is only obvious in 4ST groups, where the number of observations with left or right turn lane are quite small. Hence, the result of overall data and 4ST data may not be reliable.

Further, as same patterns are observed with right-turn and left-turn, we combine these 2 variables into a new binary variable, denoted as "TURN", where "1" stands for there exists an approaching turn and "0" else. The follow graph is made with combined column. (Separated result is in Appendix.)

Table 3.1: Intersection vs Turns

Observations	3ST	4ST	4SG
Without Left-turn	368	58	57
With Left-turn	17	44	4

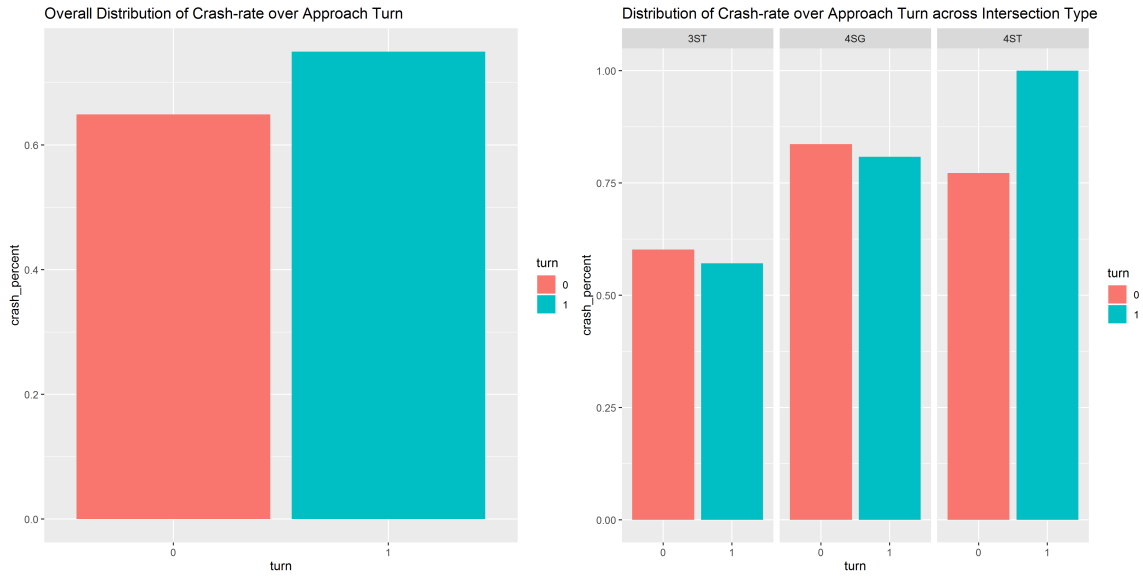


Figure 3.5: Bar Chart of Approach Left Turn

### 3.3 Crash vs Quantitative Variables

Also, there is likely to exist some association between AADT-MAJOR, AADT-MINOR, SKEW-ANGLE and crash.

#### 3.3.1 AADT-MAJOR and AADT-MINOR

Selected graphs are shown in this part.

We mainly apply the notched box-plot here to compare the distribution of AADT in intersection of crashes with intersection with no crash. The discussion is separated for each crash type. The first 3 box plots below show significant difference on median of overall distribution as the notches do not overlap. This infers that generally, the intersections with crash has higher level of AADT-Major in all 3 types of crashes.

After controlling the intersection type, all partitioned data shows same pattern that higher AADT-Major appears in group with crash. However, for PDO and BC, the difference is only significant in 3ST group and for KA, no difference is significant.

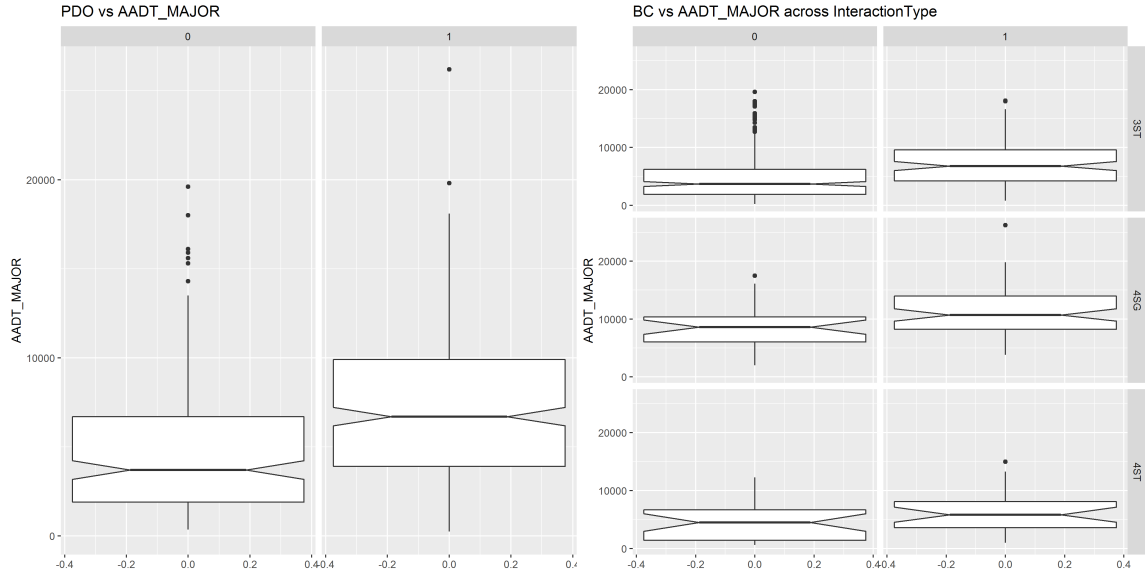


Figure 3.6: Boxplot of AADT-Major

For AADT-Minor, AADT-Minor is significantly larger in intersection with crash when exploring overall data for all of PDO, KA and BC. When taking intersection type into consideration, for 3ST group, same patterns are observed as overall data that the difference are all significant. However, opposite pattern occurs in 4SG group that intersection with crash has a lower AADT-MINOR value. The opposite difference of sample median between the with-crash and without-crash group is significant in PDO crash. For 4ST group, all difference in median is not significant.

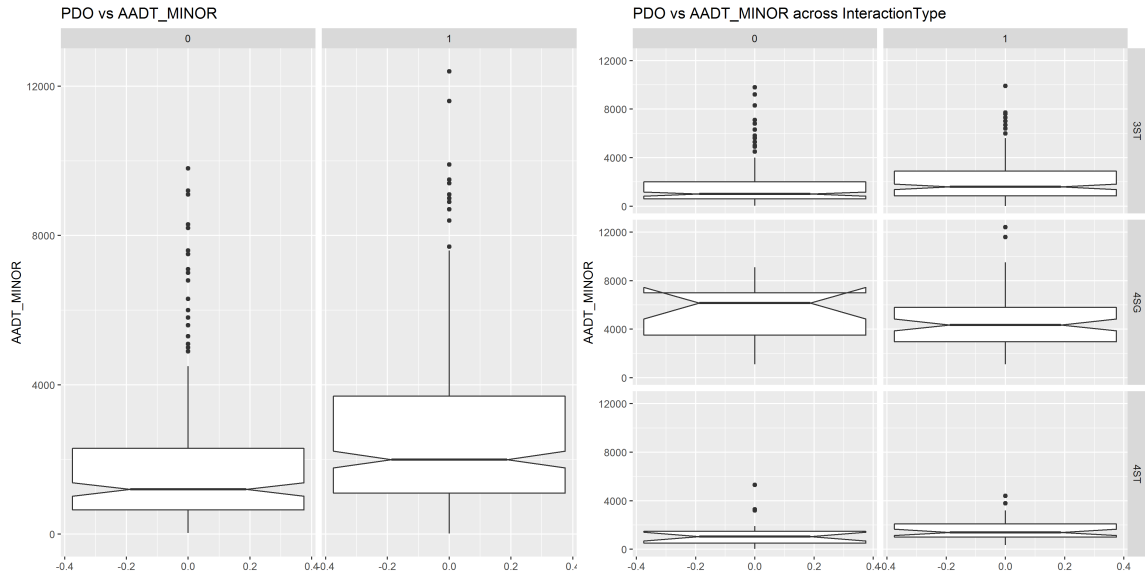


Figure 3.7: Boxplot of AADT-Minor

### 3.3.2 Skew Angle

For Skew Angle, we apply notched box plot also. In all 3 types of crash, groups with crash have smaller median for skew angle while the difference is only significant in KA crash type. When controlling intersection type, only the 3ST intersection and PDO crash group shows significant

larger skew angle median in no crash group. All other are insignificant. 4SG group get insignificant opposite pattern in PDO and BC crash.

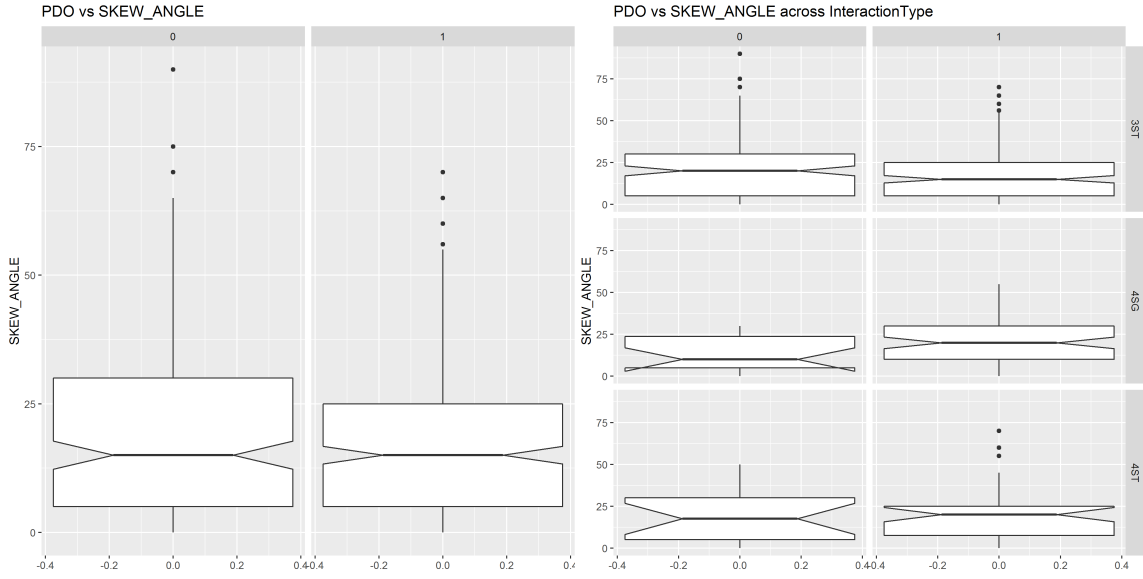


Figure 3.8: Boxplot of Skew Angle

### 3.4 EDA Conclusion

We get the following guess from the EDA part and the verification will be shown in the next inference part.

1. PDO, BC and KA has different distribution features.
2. Intersection type is a vital factor to crash counts. Four-way intersections have a higher probability for all 3 types of crash.
3. Intersection without lighting has a lower rate for all 3 types of crash. Controlling intersection type, same results are yielded except the PDO, 4ST group which has a lower crash rate when there is lighting. These results are kind of opposite to common sense.
4. Approach (left or right) turn lane has negative effect on overall crash rate, which infers that when there is an approach turn lane, the crash rate goes down. The result is reasonable only for 3ST and 4ST group individually, as 4SG group has a small number of observations.
5. AADT-Major and AADT-Minor is at a higher level in intersections with crash for whole data set. When controlling intersection, However, for AADT-Major, the tendency is only significant in 3ST group with PDO and BC. For AADT-Minor, opposite significant pattern occurs in 4SG group of PDO crash.
6. Skew angle may not act as a vital factor to crash on overall data, but may has negative effect for crash in 3ST intersections.

## 4 STATISTICAL INFERENCE

### 4.1 One v.s. PDO crashing independence test

We first test the relationship between crash and intersection type for 3 types of crash separately. We mainly use 3 types of methods here. When treating crash count as qualitative, normality assumption and same dispersion assumption are checked in each paired sample. Since none of the pairs satisfies normal assumption, we use Mann-Whitney test when paired sample has



same dispersion. When we get different dispersion, we calculate Bootstrap confidence interval for difference of sample mean to check whether the interval contains 0 or not. If the interval contains 0, we infer that there is not obvious difference on the sample mean. (Since there is still randomness in Bootstrap, the Bootstrap CI cannot be related to test significance seriously.) Re-sampling time is set as 1000 in Bootstrap method. The same dispersion tests are carried out by Ansari-Bradley nonparametric test here. Whether the impact is positive or negative can be directly get from the tests above.

When treating turning crash counts into binary variable as 1: crash, 0: nocrash, Pearson Chi-square test is applied to check independence.

For Intersection Type vs PDO crash, only the 4ST vs 4SG group pass the same dispersion test. For 3ST vs 4ST and 3ST vs 4SG, 95% Bootstrap Confidence Interval of difference of sample mean is calculate instead of testing. The result is shown in the table below. It infers that there is strong evidence that 4SG intersections have more PDO crash than 3ST. 4ST also has a higher level of crash than 3ST but not that strong.

Table 4.1: Bootstrap Confidence Interval for PDO crash counts mean difference

Difference of Mean	Lower Bound	Upper Bound	Contain 0
3ST-4ST	-1.652	-0.019	F(not obvious)
3ST-4SG	-6.670	-4.112	F

Although some of the paired does not pass same dispersion test, we still conduct the Mann-Whitney tests on all pairs to see the results. the test may lose some power when the assumptions are violated. The test procedure is listed below. For two data sequence  $X_A$  and  $X_B$ , the null hypothesis and alternative will be

$$H_0 : q_A = q_B$$

$$H_1 : q_A > q_B \text{ or } q_B > q_A$$

where  $q$  is the median, and the statistic will be

$$\frac{W - E(W)}{\sqrt{Var(W)}} \quad (4.1)$$

where

$$W = \sum_{j=1}^n R_j$$

$$E(W) = \frac{n(n+m+1)}{2}$$

$$Var(W) = \frac{mn}{12} \left\{ (n+m+1) - \frac{1}{(m+n)(m+n-1)} \sum_{j=1}^g t_j (t_j^2 - 1) \right\} \quad (4.2)$$

For PDO crashing between 3ST intersections and 4ST intersections, the  $q_A$  represents the median PDO crashing for 3ST intersections, AND  $q_B$  represents the median PDO crashing for 4ST intersections. Plus, in the statistic,  $n$  is PDO crashing sample size for 4ST intersection;  $m$  is PDO crashing sample size for 3ST intersection;  $R_j$  is the  $j$ -th sample rank for PDO crashing rank of 4ST intersection samples;  $g$  is the tie group numbers and  $t_j$  is the tie group size for  $j$ -th tie group. According to the code in Appendix 2, our conclusion is  $q_B > q_A$ , i.e. the 4ST intersection is more likely to happen PDO crash than 3ST intersection.

For PDO crashing between 4ST intersections and 4ST intersections, which the paired sample pass the same dispersion test, the  $q_A$  represents the median PDO crashing for 4ST intersections,  $q_B$  represents the median PDO crashing for 4SG intersections. Plus, in the statistic,  $n$  is

PDO crashing sample size for 4SG intersection;  $m$  is PDO crashing sample size for 4ST intersection;  $R_j$  is the  $j$ -th sample rank for PDO crashing rank of 4SG intersection samples;  $g$  is the tie group numbers and  $t_j$  is the tie group size for  $j$ -th tie group. According to the code in Appendix 2, our conclusion is  $q_B > q_A$ , i.e. the 4SG intersection is more likely to happen PDO crash than 4ST intersection.

For PDO crashing between 3ST intersections and 4SG intersections, and the  $q_A$  represents the median PDO crashing for 3ST intersections,  $q_B$  represents the median PDO crashing for 4SG intersections. Plus, in the statistic,  $n$  is PDO crashing sample size for 4SG intersection;  $m$  is PDO crashing sample size for 3ST intersection;  $R_j$  is the  $j$ -th sample rank for PDO crashing rank of 4SG intersection samples;  $g$  is the tie group numbers and  $t_j$  is the tie group size for  $j$ -th tie group. According to the code in Appendix 2, our conclusion is  $q_B > q_A$ , i.e. the 4SG intersection is more likely to happen PDO crash than 3ST intersection. According to the conclusions above, we have:

$$q_A < q_B < q_C$$

where  $q_A$  represents the median number of PDO crashes of 3ST intersection,  $q_B$  represents the median number of PDO crashes of 4ST intersection,  $q_C$  represents the median number of PDO crashes of 4SG intersection.

We can use another method - Pearson test in categorical analysis - to analyze this dependence. We firstly, transform the PDO crashing number into dummy variable. If the crashing number equals to 0, then we denote it as "noncrash", while if the crashing number is greater than 0, then we denote it as "crash". Thus, we can form the following table:

Table 4.2: Intersection VS Crash

	3ST	4ST	4SG	Total
Nocrash	173	18	18	209
Crash	212	43	84	339
Total	385	61	102	548

Then our hypothesis will be

$$H_0 : p_{ij} = p_i p_j$$

$$H_1 : p_{ij} \neq p_i p_j$$

and our statistic is:

$$Q_P = \frac{\sum_{i=1}^2 \sum_{j=1}^3 (n_{ij} - n_{i.} n_{.j} / n)^2}{n_{i.} n_{.j} / n}$$

and it approximately obeys a  $\chi^2(1)$  distribution. Thus, according to the code in Appendix 2, our conclusion is: they are dependence.

## 4.2 One v.s. BC and KA crashing independence test

We next conduct the independence test for BC crashes and KA crashes similar with the way we conduct for PDO test, so the testing procedures are omitted (codes are in Appendix 2).

For 3 paired sample in KA crash, none of the pairs pass the Ansari-Bradley test. Hence, we construct Bootstrap Confidence Interval here.

Table 4.3: Bootstrap Confidence Interval for KA crash counts mean difference

Difference of Mean	Lower Bound	Upper Bound	Contain 0
3ST-4ST	-1.180	0.014	Y(not obvious)
3ST-4SG	-0.267	-0.078	F(not obvious)
4ST-4SG	-0.209	0.047	Y(not obvious)

For 3 paired sample in BC crash, none of the pairs pass the Ansari-Bradley test. Hence, we also construct Bootstrap Confidence Interval here.

Table 4.4: Bootstrap Confidence Interval for BC crash counts mean difference

Difference of Mean	Lower Bound	Upper Bound	Contain 0
3ST-4ST	-0.716	0.070	Y(not obvious)
3ST-4SG	-2.376	-1.361	F
4ST-4SG	-2.164	-0.930	F

The Mann Whitney tests are still done and generate the result below. The order of median is the same as the shown in the Bootstrap confidence interval. However, the Bootstrap interval show some evidence that some of the inequality signs are not strictly holds.

- $q_A < q_B < q_C$ , where  $q_A$  represents the median number of BC crashes of 3st intersection,  $q_B$  represents the median number of BC crashes of 4st intersection,  $q_C$  represents the median number of BC crashes of 4sg intersection.
- $q_A < q_B = q_C$ , where  $q_A$  represents the median number of KA crashes of 3st intersection,  $q_B$  represents the median number of KA crashes of 4st intersection,  $q_C$  represents the median number of KA crashes of 4sg intersection.

The Pearson tests are also applied. And finally both of these two Pearson tests show dependence. Two categorical tables are:

Table 4.5: Categorical table for BC crash

	3ST	4ST	4SG	Total
Nocrash	264	31	30	325
Crash	121	30	72	223
Total	385	61	102	548

Table 4.6: Categorical table for KA crash

	3ST	4ST	4SG	Total
Nocrash	370	55	83	508
Crash	15	6	19	50
Total	385	61	102	548

From the discussion between intersection type and crash counts, we can draw the follow conclusion:

- (1) There are dependent between intersection type and crash counts for all 3 types of crashes by Pearson tests.
- (2) By Mann-whitney test, for PDO, 4SG intersection is more likely to happen PDO crash than 4ST intersection.

(3) As there is still randomness in Bootstrap method, some intervals constructed above have bounds close to 0, which may change sign in other realizations. Hence, here we take  $|x| > 0.1$  as an obvious difference from 0. From the tables above and combine with Mann-whitney test results, we can claim that for PDO, 4SG has obvious more crash than 3ST; for BC, 4SG has obvious more crash than 3ST and 4ST; for KA, there is likely to be no obvious pattern. However, all intervals constructed have the tendency to be at the left side of 0, which infers that there might be a tendency that the unit number of crash in all 3 types of crash follow the order  $3ST < 4ST < 4SG$ .

### 4.3 Two v.s. crashing independence test

#### 4.3.1 Crossing type + Skewed Angle v.s. Crashing

As we the skewed angle holds little choices of values, we decide to transform skewed angle into categorical data and also transform PDO crashing number as our previous method, forming the categorical table below. We settle a standard: let  $s$  denote the skewed angle, if  $0 \leq s \leq 30$ , then we record it as “1”; If  $31 \leq s \leq 60$ , then we record it as “2”; If  $61 \leq s \leq 90$ , then we record it as “3”. The effect is tested with interaction category across intersection types and skew angle. The 2-by-n tables are shown below.

Table 4.7: Discrete skew v.s. PDO crash

	3ST_1	3ST_2	3ST_3	4ST_1	4ST_2	4ST_3	4SG_1	4SG_2	Total
Nocrash	131	31	11	14	4	0	18	0	209
Crash	168	39	5	36	6	1	71	13	339
Total	299	70	16	50	10	1	89	13	548

Table 4.8: Discrete skew v.s. BC crash

	3ST_1	3ST_2	3ST_3	4ST_1	4ST_2	4ST_3	4SG_1	4SG_2	Total
Nocrash	203	46	15	25	6	0	29	1	325
Crash	96	24	1	25	4	1	60	12	223
Total	299	70	16	50	10	1	89	13	548

Table 4.9: Discrete skew v.s. KA crash

	3ST_1	3ST_2	3ST_3	4ST_1	4ST_2	4ST_3	4SG_1	4SG_2	Total
Nocrash	285	69	16	44	10	1	71	12	508
Crash	14	1	0	6	0	0	18	1	40
Total	299	70	16	50	10	1	89	13	548

Then, the Pearson test shows that there exists a dependence for all three kinds of crashes (codes are in Appendix 2.) and we have also conducted the association test for separate single one vs one group as following, the p-value table shows:

Table 4.10: P-value for skew angle vs Crash

p-value	3ST	4ST	4SG
PDO	0.1472	0.6058	0.2026
BC	0.0806	0.5006	0.1840
KA	0.3197	0.4809	0.5556
Mixed	0.0496	0.2618	0.2026

The one vs one test show no significance except 3ST with mixed data, which is opposite to the overall test.

#### 4.3.2 Crossing type + Lighting vs Crashing

As the lighting itself is a dummy variable, we can form the following categorical table:

Table 4.11: Crossing type + Lighting v.s. PDO crash

	3ST_0	3ST_1	4ST_0	4ST_1	4SG_0	4SG_1	Total
Nocrash	74	99	5	13	6	12	209
Crash	76	136	18	25	28	66	339
Total	150	235	23	38	24	78	548

Then, the Pearson test shows that there exists a dependence (codes are in Appendix 2.)

One vs one tests are also done here.

We also use the three 2-by-2 contingency tables between Lighting and crash with PDO crash for each intersection type, which can directly get from table above. Same table are generated for KA and BC. For all 2-by-2 table, only table of KA, 4ST has cell counts less than 5. For this one, we apply Fisher-exact test. For others, we conduct individual Pearson test on each table. None of the above tests reject the Null hypothesis. P-value are shown in table below. When testing the overall effect across table, in the Mantel-Haenszel test cannot reject the Null in all 3 types of crashes. Hence, no dependence of Lighting and crash are observed here.

When we mix 3 types of crash together and still construct 2-by-2 table across intersections, same results are generated.

In all, we conclude that there is no effect of Lighting on crash when controlling intersection type.

Table 4.12: P-value for Lighting vs Crash

p-value	3ST	4ST	4SG
PDO	0.1657	0.3006	0.7644
KA	0.6484	0.0746	0.7509
BC	0.247	0.4883	0.976
Mixed	0.0879	0.9494	0.5732

Again, the one vs one test show opposite result to the overall test.

#### 4.3.3 Crossing type + turns for Turn vs Crashing

Based on the results from the EDA part, we combine Approach Left Turn and Approach Right Turn as one dummy variable where 1 stands for there is an approach turn and 0 stands for not. Even then, approach turn existence is still a rare event in all intersection types. Also, if we

continuing partitioning the data by crash type to form contingency table, we will have cells with extremely small counts equal to 0. Hence, we only analyses with mixed data with all crashes types mixed. We can form the following contingency table:

Table 4.13: Overall Crossing type + turns v.s. crash

	3ST_00	3ST_01	3ST_10	3ST_11	4ST_00	4ST_10	4ST_11	4SG_00	4SG_01	4SG_10	4SG_11	Total
Nocrash	164	2	3	4	18	0	0	9	1	4	4	209
Crash	200	2	7	3	39	3	1	46	2	18	18	339
Total	364	4	10	7	57	3	1	55	3	22	22	548

where “00” represents no left approach or right approach; “01” represents no left approach but a right approach; “10” represents no right approach but left approach; “11” represents a left approach and right approach. Then, the Pearson test shows that there exists a dependence (codes are in Appendix 2.)

One vs one tests are also applied with 2-by-2 tables from above larger table for each intersection type separately. We do both Mantel-Haenszel test and 3 individual tests to test for independence between column and row.

For the 3 individual test, for 3ST and 4SG, we have all cell counts larger than 5. We apply Pearson Chi-square tests and get p-value as 0.7834 and 0.713, which infers there is no dependency. For 4ST, we use the Fisher-exact test as counts are rather small and get p-value of 0.3729. The individual tests show no relationship between Crash and Turns.

For the Mantel-Haenszel test, although we get the result that there is no overall effect of Turn on crash when controlling intersection type, the assumptions are not all satisfied due to small cell counts. But, it can still be some kind of evidence that there is no relationship.

Again, we get difference result between overall test and one vs one test.

#### 4.3.4 Crossing type + AADT major VS Crashing

We firstly partition the full samples into 6 parts: 3ST intersection + crash; 3ST intersection + noncrash; 4ST intersection + crash; 4ST intersection + noncrash; 4SG intersection + crash; 4SG intersection + noncrash. And we carry out a Shapiro normality test on these 6 sample pieces, however, few of them pass the test. Hence, once again, we need to turn to the Mann-Whitney test.

Comparing the AADT major and controlling the intersection type in 3st, we have the following hypotheses:

$$H_0 : q_A = q_B$$

$$H_1 : q_A < q_B \text{ or } q_A > q_B$$

where  $q_A$  represents the median of AADT major in PDO noncrash situation in 3ST intersection, while  $q_B$  represents the median of AADT major with PDO crash situation in 3ST intersection. And the statistic is similar with (4.2). And our final conclusion (codes are in Appendix 2) is:

$$q_A < q_B.$$

Comparing the AADT major and controlling the intersection type in 4ST, we have the following hypotheses:

$$H_0 : q_A = q_B$$

$$H_1 : q_A < q_B \text{ or } q_A > q_B$$

where  $q_A$  represents the median of AADT major in PDO noncrash situation in 4ST intersection, while  $q_B$  represents the median of AADT major with PDO crash situation in 4ST intersection. And the statistic is similar with (4.2). And our final conclusion (codes are in Appendix 2) is:

$$q_A = q_B.$$

Comparing the AADT major and controlling the intersection type in 4SG, we have the following hypotheses:

$$\begin{aligned} H_0 : q_A &= q_B \\ H_1 : q_A &< q_B \text{ or } q_A > q_B \end{aligned}$$

where  $q_A$  represents the median of AADT major in PDO noncrash situation in 4SG intersection, while  $q_B$  represents the median of AADT major with PDO crash situation in 4SG intersection. And the statistic is similar with (4.2). And our final conclusion (codes are in Appendix 2) is:

$$q_A = q_B.$$

By same Mann-Whitney tests, for KA crash, AADT-Major only has significant effect at 3ST intersection. Larger AADT-Major intersection has higher crash rate. For BC crash, significant impact is observed both in 3ST and 4ST. In both case, larger AADT-Major is related to higher crash rate.

#### 4.3.5 Crossing type + AADT minor vs Crashing

With the same partition made for AADT major, we also carry out a Shapiro normality test on these 6 sample pieces for AADT minor, however, few of them pass the test. Hence, once again, we need to turn to the Mann-Whitney test.

Comparing the AADT minor and controlling the intersection type in 3ST, we have the following hypotheses:

$$\begin{aligned} H_0 : q_A &= q_B \\ H_1 : q_A &< q_B \text{ or } q_A > q_B \end{aligned}$$

where  $q_A$  represents the median of AADT minor in PDO noncrash situation in 3st intersection, while  $q_B$  represents the median of AADT minor with PDO crash situation in 3ST intersection. And the statistic is similar with (4.2). And our final conclusion (codes are in Appendix 2) is:

$$q_A < q_B.$$

Comparing the AADT minor and controlling the intersection type in 4ST, we have the following hypotheses:

$$\begin{aligned} H_0 : q_A &= q_B \\ H_1 : q_A &< q_B \text{ or } q_A > q_B \end{aligned}$$

where  $q_A$  represents the median of AADT minor in PDO noncrash situation in 4ST intersection, while  $q_B$  represents the median of AADT minor with PDO crash situation in 4ST intersection. And the statistic is similar with (4.2). And our final conclusion (codes are in Appendix 2) is:

$$q_A = q_B.$$

Comparing the AADT minor and controlling the intersection type in 4SG, we have the following hypotheses:

$$\begin{aligned} H_0 : q_A &= q_B \\ H_1 : q_A &< q_B \text{ or } q_A > q_B \end{aligned}$$

where  $q_A$  represents the median of AADT minor in PDO noncrash situation in 4SG intersection, while  $q_B$  represents the median of AADT minor with PDO crash situation in 4SG intersection. And the statistic is similar with (4.2). And our final conclusion (codes are in Appendix 2) is:

$$q_A = q_B.$$

By same Mann-Whitney tests, for KA crash, no significant effects is observed. For BC crash, positive effect on crash rate is observed in 3ST intersection. For 4ST group, the AADT-Minor in crash and noncrash group do not pass the same dispersion test. We again carry out the Bootstrap Confidence Interval method on the difference of sample mean. The 95% interval for mean(crash)-mean(noncrash) is (100,1051.25). Hence, there is obvious positive effect of AADT-Minor on crash rate in 4ST.

## 5 DISCUSSION OF RESULTS AND CONCLUDING REMARKS

### 5.1 Conclusions

From all the discussion above, we can draw the conclusion below:

For Intersection Type vs crash counts only: Intersection type affects the probability of all 3 types of crash. There is a tendency that the crash probability in 3 types of intersection follows that order 3ST<4ST<4SG. The tendency is obvious in PDO,4SG vs 4ST and BC, 3ST vs 4ST vs 4SG. We may recommend to pay much more attention when coming across a 4SG intersection.

Controlling intersection type:

(1) Skew angle and Lighting: The effect of skew angle is not significant when controlling intersection type, while the interaction effect between intersection and skew angle is tested to be significant.

(2) Turns: Discussions only focus on data with 3 types of crashes mixed together due to lack of observations. Again, we have insignificant effect of turns with significant interaction effect between turns and intersection types.

(3) AADT-Major and AADT-Minor: Only positive impact of AADT-Major and AADT-Minor is observed on crash rate. For PDO, 3ST intersection with higher AADT-Major level or higher AADT-Minor level tends to has higher crash rate. For KA, same impact observed in 3ST. For BC, such positive effect occurs in 3ST and 4ST. The result infers that controlling AADT at a lower level might be a good method to reduce crash rate at 3ST and 4ST intersection, but might not help for 4SG intersections.

The results in (1) and (2) need further discussion as the existence of contradict between main effect and interaction effect. Overall, there is no common suggestion based on skew angle, lighting and turns towards crash.

### 5.2 Further Discussion

For all the analysis above, we mainly apply 3 types of technique: data visualization, confidence interval construction and hypothesis testing on location and scale. Here are some other probable methods that can be applied to explore the data set.

#### 5.2.1 Logistic regression

All the methods used can hardly predict the crash rate or crash counts. Also, the impact of certain variable when controlling 2 or more other variables cannot be discusses with current technique. Since we have counts as response variable and also many 0 in the data set, logistic models are a good option here to fit the data, either with normal family(predict crash rate) or with poisson/negative-binomial family(predict crash counts). Then we can get prediction of crash and also further explore the effect of regressors in the full model.



### 5.2.2 Sufficient dimension reduction

In statistical fields, there is a special topic - sufficient dimension reduction - which is used to select most relative variable to the response. Mathematically, for  $Y$  as response and  $X$  as a  $p$ -dimension predictor, we'd like to find a  $p \times d$  ( $d < p$ ) vector  $\eta$  such that

$$Y \perp X | \eta^T X.$$

where  $\eta^T X$  is denoted as the central subspace  $S_{Y|X}$ . Proposed by Li et. al.[1], the principal support vector machines is a popular sufficient dimension reduction method. The estimation procedure can be divided into three steps:

- 1. Compute the sample mean  $\bar{X}$  and sample variance matrix  $\hat{\Sigma}$ .
- 2. (Optimization) Let  $q_r, r = 1, \dots, h-1$  be  $h-1$  dividing points, equally spaced sample percentiles of  $\{Y_1, \dots, Y_n\}$ . Let

$$\tilde{Y}_i^r = I(Y_i > q_r) - I(Y_i \leq q_r).$$

Then let  $(\hat{\beta}_r, \hat{t}_r)$  be the minimizer of

$$\beta^T \hat{\Sigma} \beta + \lambda E_n \left\{ 1 - \tilde{Y}^r \left[ (X - \bar{X})^T \beta - t \right] \right\}^+.$$

- 3. (Estimation) Let  $\hat{v}_1, \dots, \hat{v}_d$  be the  $d$  leading eigenvectors of matrix

$$\hat{M}_n = \sum_{r=1}^{h-1} \hat{\beta}_r \hat{\beta}_r^T.$$

Then we use subspace spanned by  $\hat{v} = (\hat{v}_1, \dots, \hat{v}_d)$  to estimate  $S_{Y|X}$ .

By the way, proposed by Xia et. al.[2], the eigenvector number  $d$  can be selected by the jump ratio of eigenvalues. That is, if  $\lambda_1, \dots, \lambda_p$  is  $p$  eigenvalues of the matrix in descending order, then we forming the  $p-1$  jump ratio by

$$\{J_1, \dots, J_{p-1}\}, J_i = \frac{\lambda_i}{\lambda_{i+1}},$$

then we select  $d$  as

$$d = \arg \max_i J_i.$$

Geometrically, the following figure shows the principal.

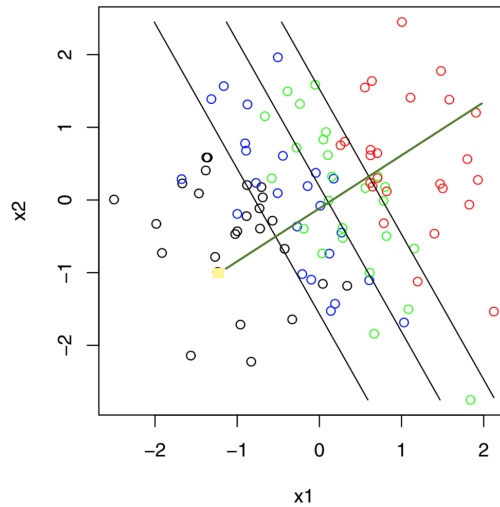


Figure 5.1: 2 dimension SVM surfaces and their normal line

In the figure, if the colors represent the value, from red (4th quantile part) to black (1st quantile part), the value  $Y$  descends. Then, three black surface is the SVM boundary for its four quantile parts. Finally, the green line - the normal direction of its SVM boundary - can perfectly delineate a path (forming by the weights of  $x_1$  and  $x_2$ ) where the  $Y$  value grows fastest.

Implementing this algorithm by Python (code is in Appendix 2), we get the central subspace for PDO crashing number as response:

```
array([[ 0.00000000e+00, -6.28336625e-03, -3.44987939e-04, -1.54542490e-04],
       [ 9.99807242e-01, -1.96296091e-02, -3.32648777e-04, -1.63439760e-04],
       [-9.52640689e-04, -2.96027569e-02, -9.24611159e-01, -3.78741274e-01],
       [-1.67939885e-04, -7.34446402e-03, -2.37055602e-01, 6.33323237e-01],
       [-7.19757840e-05, -6.52082837e-03, -2.96534802e-01, 6.74871829e-01],
       [ 1.96096296e-02,  9.99300962e-01, -3.10761591e-02, -2.16533571e-03]])
```

It shows that for PDO crashing number, there are totally 4 factors which may significantly affects it. The AADT major holds a positive effect ( 9.99807242e-01) to PDO crash; The skewed angle holds a positive effect (9.99300962e-01) to PDO crash; The lighting holds a negative effect (-9.24611159e-01) to PDO crash; The right-turn approach holds positive effect (6.74871829e-01) to PDO crash.

Similarly, for BC crashing number as response, the central subspace will be:

```
array([[ 4.00994989e-05,  1.88638392e-05,  2.29261861e-04],
       [-2.92592349e-05, -2.48004596e-06, -4.20608685e-05],
       [-9.87266764e-01, -1.21357177e-02, -1.58483896e-01],
       [-1.53212888e-01,  3.38178083e-01,  9.28526411e-01],
       [ 4.23293850e-02,  9.41003883e-01, -3.35737815e-01],
       [-6.19423106e-03,  1.07378434e-05, -1.27455844e-03]])
```

It can also reveal three key and relative factor: lighting for negative effect, left-turn and right-turn approaches for positive effect. And certainly we can do the same procedure on the KA crashing easily, so we omit it here.

## 6 APPENDIX

### 6.1 Extra Plots

Extra support plots for AADT-Major:

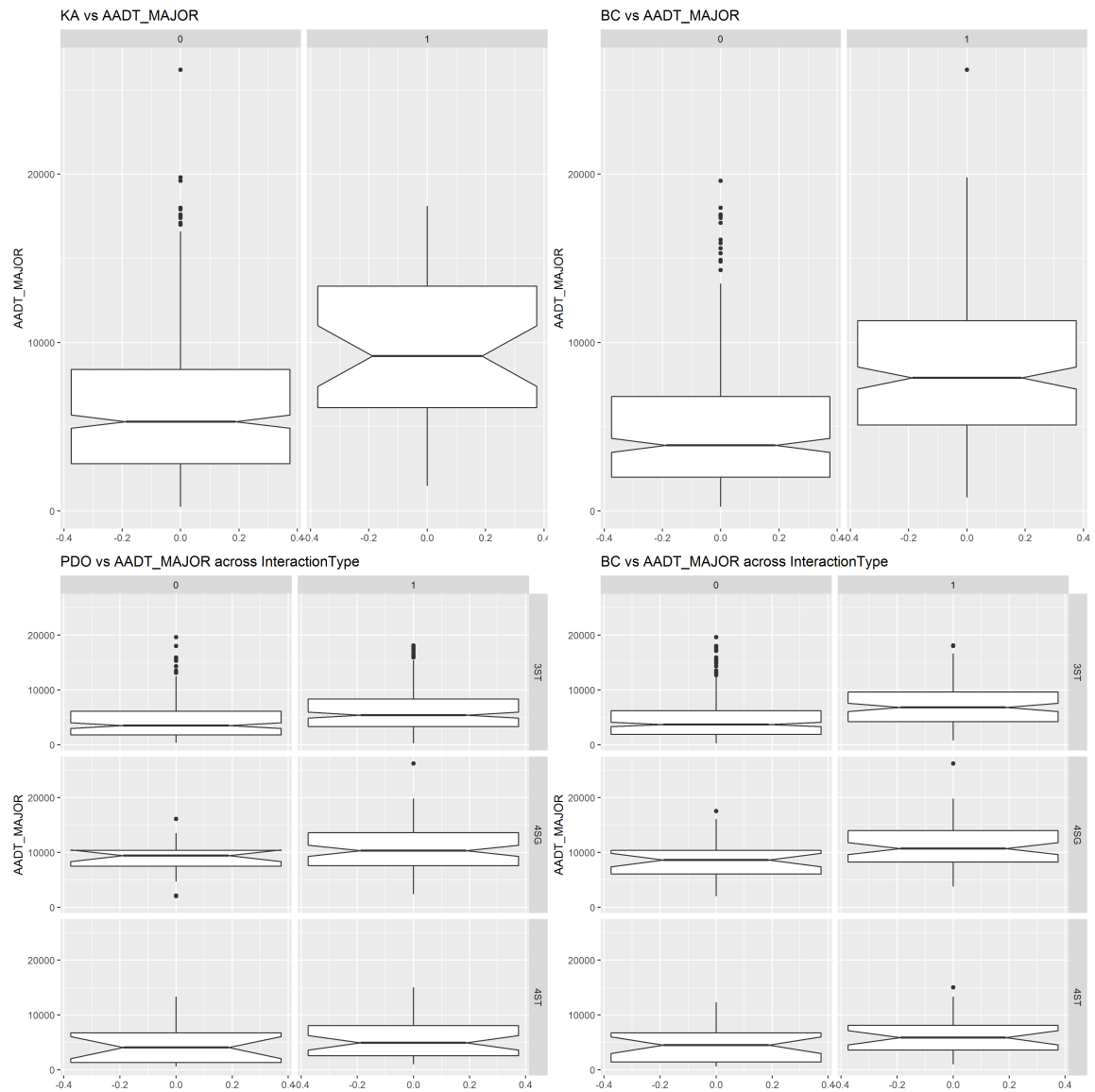


Figure 6.1: Extra plots for AADT-Major

Extra support plots for AADT-Minor:

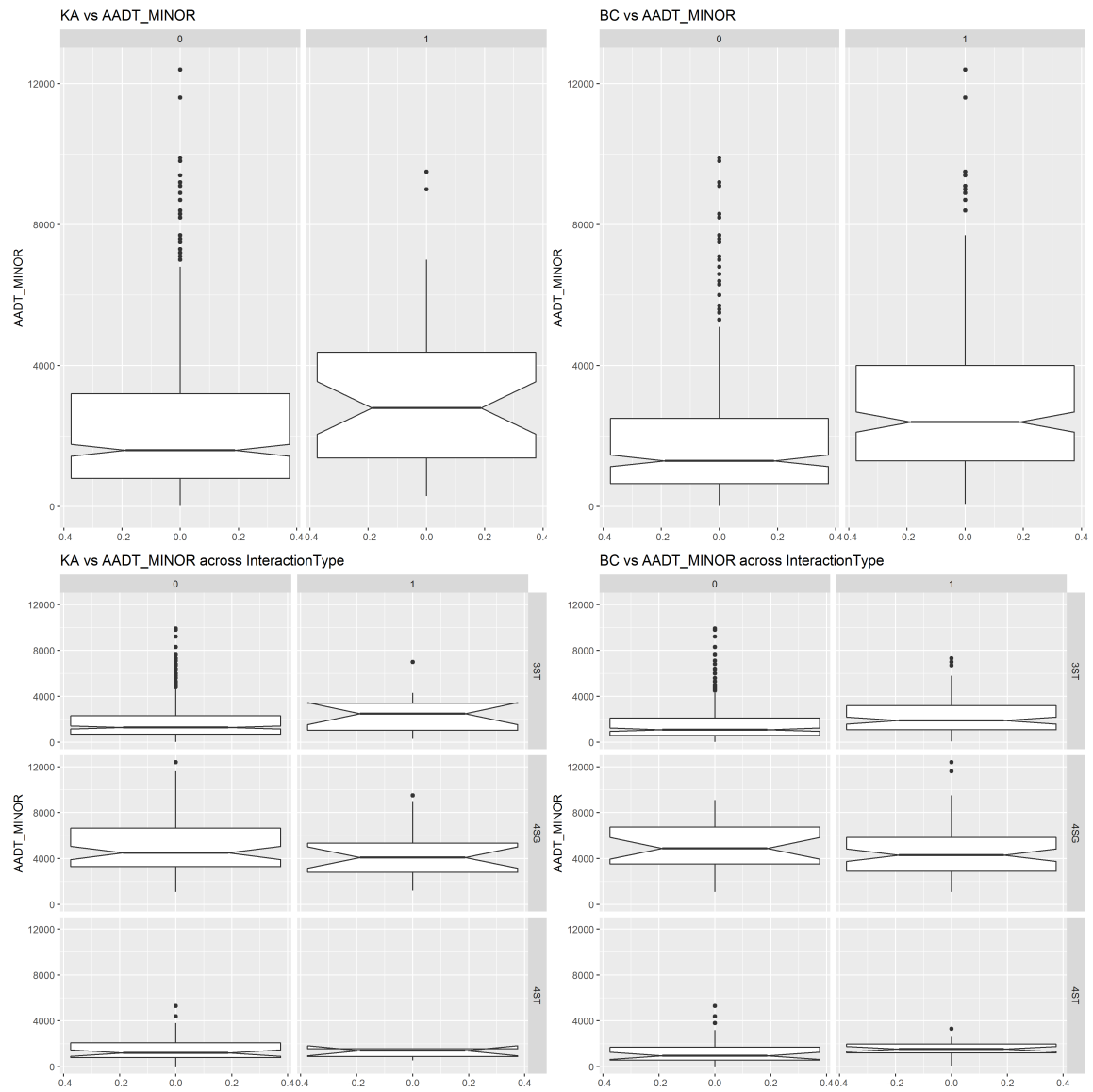


Figure 6.2: Extra plots for AADT-Minor

Extra support plots for Skew Angle:

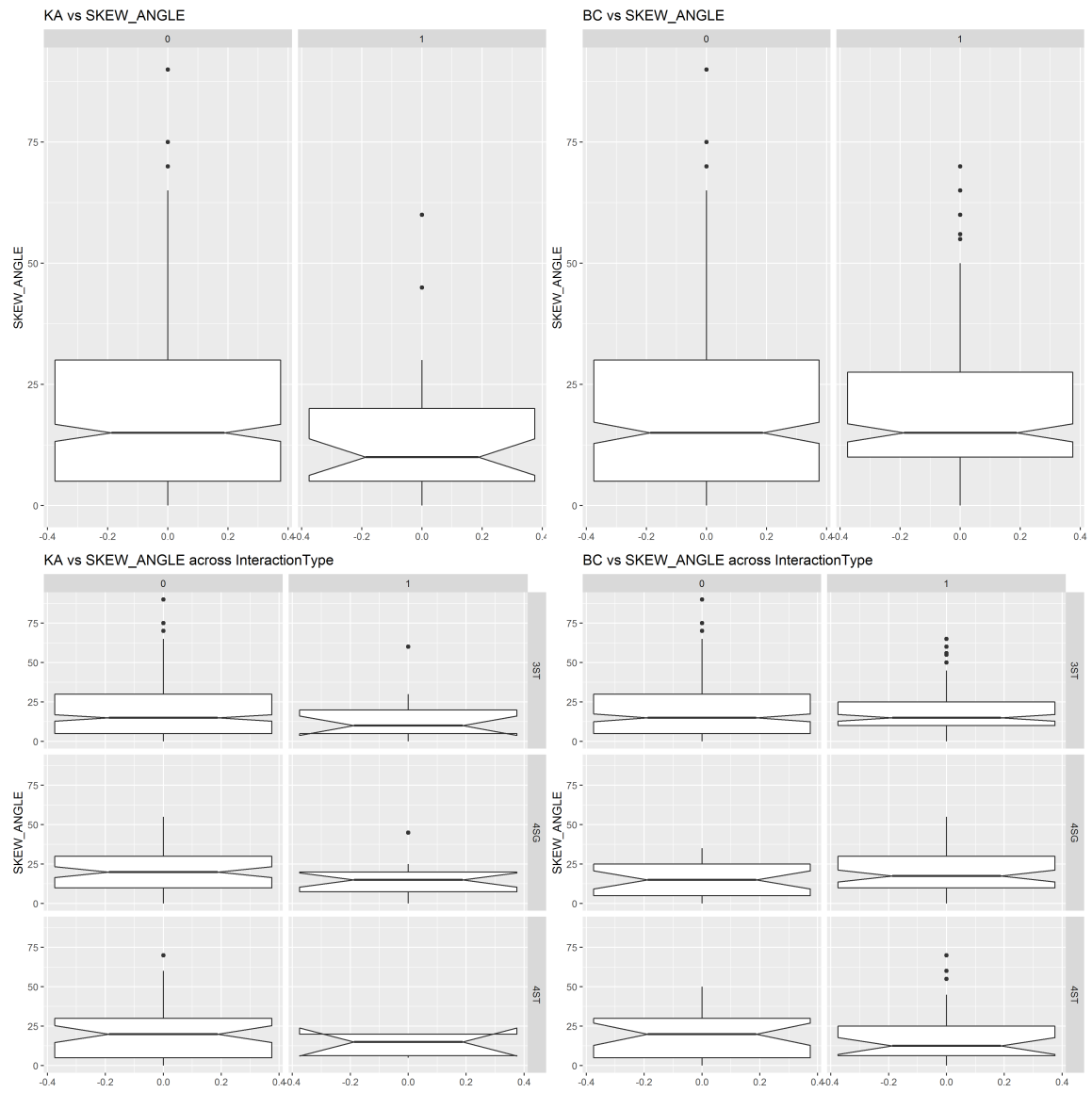


Figure 6.3: Extra plots for Skew-Angle

Extra support plots for Lighting:

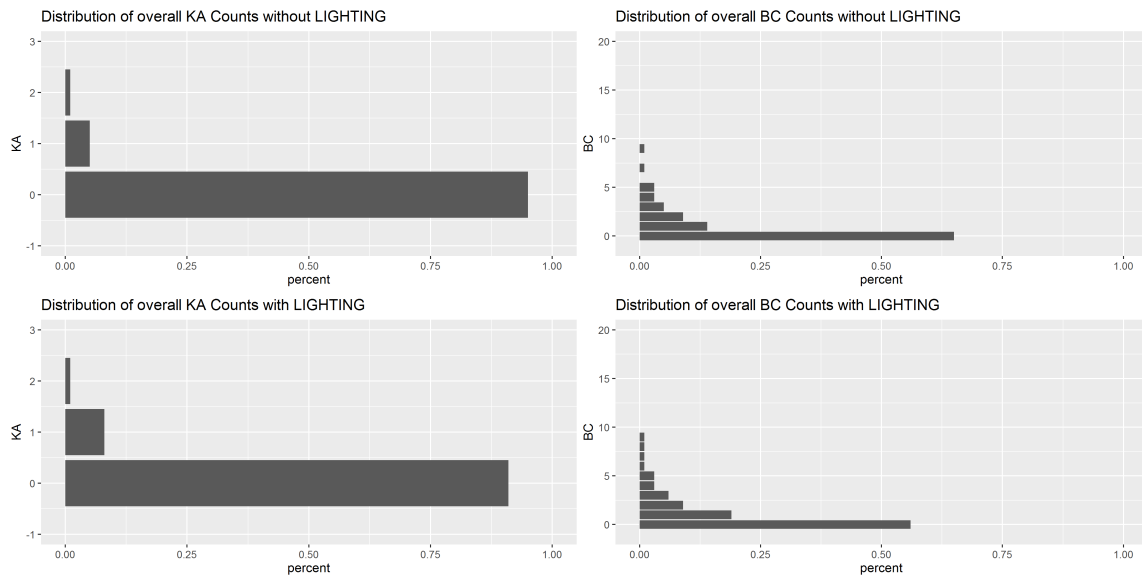


Figure 6.4: Extra plots for Lighting

## 6.2 R codes

### 6.2.1 EDA

```
library(tidyverse)
library(gridExtra)
dat_1 = read.csv('./../data/3ST.csv')
dat_2 = read.csv('./../data/4ST.csv')
dat_3 = read.csv('./../data/4SG.csv')
dat_all = rbind(dat_1, dat_2, dat_3)
dat_all$INTERTYPE = c(rep('3ST', nrow(dat_1)), rep('4ST', nrow(dat_2)), rep('4SG', nrow(dat_3)))
```

*## 1.1 Overall*

*## Overall distribution of 3 types of accidents: Use original counts, and group counts larger than 10 to be a single group. Make pie plots*

*### 1-1-1 PDO*

```
cbPalette <- c("#999999", "#E69F00", "#56B4E9", "#009E73",
               "#F0E442", "#0072B2", "#D55E00", "#CC79A7",
               "#000000", "#330066", "#CC9933", "#006600",
               "#660033", "#C1FFC1", "#DCDCDC", "#BDB76B", "#FFD39B"
)
```

```
tab1 = table(dat_all$PDO)
over15 = sum(tab1[which(as.numeric(names(tab1))>15)])
tab1 = tab1[which(as.numeric(names(tab1))<=15)]
accidents = c(names(tab1), 'over15')
accidents = factor(accidents, levels = c("0", "1", "2", "3", "4", "5", "6", "7",
    "8", "9", "10", "11", "12", "13", "15", "over15"))
tab1 = data.frame(counts = c(as.vector(tab1), over15), accidents = accidents)
tab1 = tab1 %>% mutate(labels =
```

```

      paste(accidents, "(", counts, " / ", round(counts/sum(
        counts), 2)*100, "%", ") ", sep=""))
p111 = ggplot(tab1, aes(x="", y=counts, fill=(accidents)))+
  geom_bar(stat='identity')+coord_polar("y")+
  scale_fill_manual(values =cbPalette, labels=tab1$labels)+
  theme(legend.text = element_text(size = 9), axis.text = element_blank())
  +
  ggtitle("Distribution_of_Number_of_Overall_Annual_PDO_Counts")
png(file="../../report/image/p1.png")
p111
dev.off()
### 1-1-2 KA
tab1 = table(dat_all$KA)
over15 = sum(tab1[which(as.numeric(names(tab1))>15)])
tab1 = tab1[which(as.numeric(names(tab1))<=15)]
accidents = c(names(tab1), 'over15')
accidents=factor(accidents, levels=c(names(tab1), 'over15'))
tab1 = data.frame(counts=c(as.vector(tab1), over15), accidents)
tab1 = tab1 %>% mutate(labels=
  paste(accidents, "(", counts, " / ", round(counts/sum(
    counts), 2)*100, "%", ") ", sep=""))
p112 = ggplot(tab1, aes(x="factor(1)", y=counts, fill=accidents))+
  geom_bar(stat='identity')+coord_polar("y")+
  scale_fill_manual(values =cbPalette, labels=tab1$labels)+
  theme(legend.text = element_text(size = 10), axis.text = element_blank()
  )+
  ggtitle("Distribution_of_Number_of_Overall_Annual_KA_Counts")
png(file="../../report/image/p112.png")
p112
dev.off()
### 1-1-3 BC
tab1 = table(dat_all$BC)
over15 = sum(tab1[which(as.numeric(names(tab1))>15)])
tab1 = tab1[which(as.numeric(names(tab1))<=15)]
accidents = c(names(tab1), 'over15')
accidents=factor(accidents, levels=c(names(tab1), 'over15'))
tab1 = data.frame(counts=c(as.vector(tab1), over15), accidents)
tab1 = tab1 %>% mutate(labels=
  paste(accidents, "(", counts, " / ", round(counts/sum(
    counts), 2)*100, "%", ") ", sep=""))
p113 = ggplot(tab1, aes(x="factor(1)", y=counts, fill=accidents))+
  geom_bar(stat='identity')+coord_polar("y")+
  scale_fill_manual(values =cbPalette, labels=tab1$labels)+
  theme(legend.text = element_text(size = 10), axis.text = element_blank()
  )+
  ggtitle("Distribution_of_Number_of_Overall_Annual_BC_Counts")
png(file="../../report/image/p113.png")
p113
dev.off()

```

## ## 1.2 Accident counts VS variables

### ### 1-2-1 Accident across interaction type

```
datplot = dat_all
datplot$PDO[which(datplot$PDO>15)] = 'over15'
datplot$PDO=factor(datplot$PDO, levels=c(as.character(0:15), 'over15'))
p121_PDO = ggplot(datplot, aes(x=factor(1), fill=PDO))+
  geom_bar(position = 'fill')+
  coord_polar("y")+
  scale_fill_manual(values =cbPalette )+
  facet_wrap(~INTERTYPE)+
  theme(legend.text = element_text(size = 10),
        axis.text = element_blank(),
        legend.position = 'top',
        legend.title=element_text(size=10))+
  ggtitle("Distribution_of_PDO_Counts_over_InteractionType")
png(file="./../report/image/p121_PDO.png")
p121_PDO
dev.off()
#KA
datplot = dat_all
datplot$KA[which(datplot$KA>15)] = 'over15'
datplot$PDO=factor(datplot$PDO, levels=c(as.character(0:15), 'over15'))
p121_KA = ggplot(datplot, aes(x=factor(1), fill=KA))+
  geom_bar(position = 'fill')+
  coord_polar("y")+
  scale_fill_manual(values =cbPalette )+
  facet_wrap(~INTERTYPE)+
  theme(legend.text = element_text(size = 10),
        axis.text = element_blank(),
        legend.position = 'top',
        legend.title=element_text(size=10))+
  ggtitle("Distribution_of_KA_Counts_over_InteractionType")
png(file="./../report/image/p121_KA.png")
p121_KA
dev.off()
#BC
datplot = dat_all
datplot$BC[which(datplot$BC>15)] = 'over15'
datplot$PDO=factor(datplot$PDO, levels=c(as.character(0:15), 'over15'))
p121_BC = ggplot(datplot, aes(x=factor(1), fill=BC))+
  geom_bar(position = 'fill')+
  coord_polar("y")+
  scale_fill_manual(values =cbPalette )+
  facet_wrap(~INTERTYPE)+
  theme(legend.text = element_text(size = 10),
        axis.text = element_blank(),
        legend.position = 'top',
        legend.title=element_text(size=10))+
  ggtitle("Distribution_of_BC_Counts_over_InteractionType")
```



```

png( file = "../.. /report/image/p121_BC.png")
p121_BC
dev.off()
### 1-2-2 Accident VS AADT-MAJOR
g1 = dat_all %>% ggplot(aes(AADT_MAJOR,AADT_MINOR))+geom_point()+
  geom_smooth(method='lm')+ggtitle(' Relationship_between_AADT-Major_and_
  AADT-Minor')+facet_grid(~INTERTYPE)
ggsave( file = "../.. /report/image/major_minor.png" , g1)
cor( dat_all$AADT_MAJOR, dat_all$AADT_MINOR)
dat_all%>% group_by(INTERTYPE) %>% summarise(r=cor(AADT_MAJOR,AADT_MINOR)
)
##overall
p122_overall_pdo = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(PDO>0)) %>%
  ggplot(aes(y=AADT_MAJOR))+geom_boxplot(notch=TRUE)+facet_grid(.~PDO_
  ACCIDENT)+ggtitle("PDO_vs_AADT_MAJOR")
ggsave( file = "../.. /report/image/major_all_pdo.png" , p122_overall_pdo)

p122_overall_ka = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(KA>0)) %>%
  ggplot(aes(y=AADT_MAJOR))+geom_boxplot(notch=TRUE)+facet_grid(.~PDO_
  ACCIDENT)+ggtitle("KA_vs_AADT_MAJOR")

p122_overall_bc = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(BC>0)) %>%
  ggplot(aes(y=AADT_MAJOR))+geom_boxplot(notch=TRUE)+facet_grid(.~PDO_
  ACCIDENT)+ggtitle("BC_vs_AADT_MAJOR")

```

Across **interaction** type

```

p122_pdo = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(PDO>0)) %>%
  ggplot(aes(y=AADT_MAJOR))+geom_boxplot(notch=TRUE)+facet_grid(INTERTYPE
  ~PDO_ACCIDENT)+ggtitle("PDO_vs_AADT_MAJOR_across_InteractionType")

p122_ka = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(KA>0)) %>%
  ggplot(aes(y=AADT_MAJOR))+geom_boxplot(notch=TRUE)+facet_grid(INTERTYPE
  ~PDO_ACCIDENT)+ggtitle("KA_vs_AADT_MAJOR_across_InteractionType")
ggsave( file = "../.. /report/image/major_ka.png" , p122_ka)

p122_bc = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(BC>0)) %>%
  ggplot(aes(y=AADT_MAJOR))+geom_boxplot(notch=TRUE)+facet_grid(INTERTYPE
  ~PDO_ACCIDENT)+ggtitle("BC_vs_AADT_MAJOR_across_InteractionType")
ggsave( file = "../.. /report/image/major_bc.png" , p122_bc)
### 1-2-3 Accident VS AADT-MINOR
p123_overall_pdo = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(PDO>0)) %>%
  ggplot(aes(y=AADT_MINOR))+geom_boxplot(notch=TRUE)+facet_grid(.~PDO_
  ACCIDENT)+ggtitle("PDO_vs_AADT_MINOR")
ggsave( file = "../.. /report/image/minor_all_pdo.png" , p123_overall_pdo)

p123_overall_ka = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(KA>0)) %>%
  ggplot(aes(y=AADT_MINOR))+geom_boxplot(notch=TRUE)+facet_grid(.~PDO_
  ACCIDENT)+ggtitle("KA_vs_AADT_MINOR")

```

```

p123_overall_bc = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(BC>0)) %>%
  ggplot(aes(y=AADT_MINOR))+geom_boxplot(notch=TRUE)+facet_grid(.~PDO_
    ACCIDENT)+ggtitle("BC_vs_AADT_MINOR")

#Across interaction type

P123_pdo = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(PDO>0)) %>%
  ggplot(aes(y=AADT_MINOR))+geom_boxplot(notch=TRUE)+facet_grid(INTERTYPE
    ~PDO_ACCIDENT)+ggtitle("PDO_vs_AADT_MINOR_across_InteractionType")
ggsave(file="./../report/image/minor_pdo.png", P123_pdo)

P123_ka = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(KA>0)) %>%
  ggplot(aes(y=AADT_MINOR))+geom_boxplot(notch=TRUE)+facet_grid(INTERTYPE
    ~PDO_ACCIDENT)+ggtitle("KA_vs_AADT_MINOR_across_InteractionType")

P123_bc = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(BC>0)) %>%
  ggplot(aes(y=AADT_MINOR))+geom_boxplot(notch=TRUE)+facet_grid(INTERTYPE
    ~PDO_ACCIDENT)+ggtitle("BC_vs_AADT_MINOR_across_InteractionType")

### 1-2-4 Accident VS SKEW-ANGLE
#overall
p123_overall_pdo = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(PDO>0)) %>%
  ggplot(aes(y=SKEW_ANGLE))+geom_boxplot(notch=TRUE)+facet_grid(.~PDO_
    ACCIDENT)+ggtitle("PDO_vs_SKEW_ANGLE")
ggsave(file="./../report/image/angle_all_pdo.png", p123_overall_pdo)

p123_overall_ka = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(KA>0)) %>%
  ggplot(aes(y=SKEW_ANGLE))+geom_boxplot(notch=TRUE)+facet_grid(.~PDO_
    ACCIDENT)+ggtitle("KA_vs_SKEW_ANGLE")

p123_overall_bc = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(BC>0)) %>%
  ggplot(aes(y=SKEW_ANGLE))+geom_boxplot(notch=TRUE)+facet_grid(.~PDO_
    ACCIDENT)+ggtitle("BC_vs_SKEW_ANGLE")

#Across interaction type
p123_pdo = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(PDO>0)) %>%
  ggplot(aes(y=SKEW_ANGLE))+geom_boxplot(notch=TRUE)+facet_grid(INTERTYPE
    ~PDO_ACCIDENT)+ggtitle("PDO_vs_SKEW_ANGLE_across_InteractionType")
ggsave(file="./../report/image/angle_pdo.png", p123_pdo)

p123_ka = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(KA>0)) %>%
  ggplot(aes(y=SKEW_ANGLE))+geom_boxplot(notch=TRUE)+facet_grid(INTERTYPE
    ~PDO_ACCIDENT)+ggtitle("KA_vs_SKEW_ANGLE_across_InteractionType")

p123_bc = dat_all %>% mutate(PDO_ACCIDENT=as.numeric(BC>0)) %>%
  ggplot(aes(y=SKEW_ANGLE))+geom_boxplot(notch=TRUE)+facet_grid(INTERTYPE
    ~PDO_ACCIDENT)+ggtitle("BC_vs_SKEW_ANGLE_across_InteractionType")

```

### 1-2-5 Accident VS lighting

```
dat_all$crash = as.numeric((dat_all$PDO+dat_all$BC+dat_all$KA)>0)
getcatgraph <- function(v, crash) {
  if (crash=='PDO') {a=25;b=0.5}
  if (crash=='KA') {a=3;b=1}
  if (crash=='BC') {a=20;b=1}
  if (crash=='crash') {a=2;b=1}
  g1 = dat_all %>% filter(!sym(v)==0) %>% group_by(!sym(crash)) %>%
    summarise(count=n()) %>% mutate(percent=round(count/sum(count),2))
    %>% ggplot(aes(x=!sym(crash),y=percent))+
    geom_bar(stat='identity')+coord_flip()+xlim(c(-1,a))+ylim(c(0,b))+
    ggtitle(c(paste("Distribution_of_overall",as.character(crash),"Counts_
      _without",as.character(v))))
  g2 = dat_all %>% filter(!sym(v)==1) %>% group_by(!sym(crash)) %>%
    summarise(count=n()) %>% mutate(percent=round(count/sum(count),2))
    %>% ggplot(aes(x=!sym(crash),y=percent))+
    geom_bar(stat='identity')+coord_flip()+xlim(c(-1,a))+ylim(c(0,b))+
    ggtitle(c(paste("Distribution_of_overall",as.character(crash),"Counts_
      _with",as.character(v))))

  g <- arrangeGrob(g1,g2,nrow=2)
  ggsave(file=paste("../report/image/",as.character(v),"_all_",as.
    character(crash),".png",sep=""), g)
```

```
g1 = dat_all %>% filter(!sym(v)==0,INTERTYPE=='3ST') %>% group_by(!
  sym(crash)) %>% summarise(count=n()) %>% mutate(percent=round(count/
    sum(count),2)) %>% ggplot(aes(x=!sym(crash),y=percent))+
    geom_bar(stat='identity')+coord_flip()+xlim(c(-1,a))+ylim(c(0,b))+
    ggtitle(paste("Distribution_of_3ST",as.character(crash),"Counts_
      without",as.character(v)))
g2 = dat_all %>% filter(!sym(v)==1,INTERTYPE=='3ST') %>% group_by(!
  sym(crash)) %>% summarise(count=n()) %>% mutate(percent=round(count/
    sum(count),2)) %>% ggplot(aes(x=!sym(crash),y=percent))+
    geom_bar(stat='identity')+coord_flip()+xlim(c(-1,a))+ylim(c(0,b))+
    ggtitle(paste("Distribution_of_3ST",as.character(crash),"Counts_with"
      ,as.character(v)))
```

```
g <- arrangeGrob(g1,g2,nrow=2)
ggsave(file=paste("../report/image/",v,"_3ST_",crash,".png",sep="")
  ), g)
```

```
g1 = dat_all %>% filter(!sym(v)==0,INTERTYPE=='4ST') %>% group_by(!
  sym(crash)) %>% summarise(count=n()) %>% mutate(percent=round(count/
    sum(count),2)) %>% ggplot(aes(x=!sym(crash),y=percent))+
    geom_bar(stat='identity')+coord_flip()+xlim(c(-1,a))+ylim(c(0,b))+
    ggtitle(paste("Distribution_of_4ST",crash,"Counts_without",v))
g2 = dat_all %>% filter(!sym(v)==1,INTERTYPE=='4ST') %>% group_by(!
  sym(crash)) %>% summarise(count=n()) %>% mutate(percent=round(count/
    sum(count),2)) %>% ggplot(aes(x=!sym(crash),y=percent))+
```

```

geom_bar(stat='identity')+coord_flip()+xlim(c(-1,a))+ylim(c(0,b))+
ggtitle(paste("Distribution_of_4ST", crash, "Counts_with", v))

g <- arrangeGrob(g1,g2,nrow=2)
ggsave(file=paste("../.. /report/image/", v, "_4ST_", crash, ".png", sep=""
), g)

g1 = dat_all %>% filter(!sym(v)==0,INTERTYPE=='4SG') %>% group_by(!
  sym(crash)) %>% summarise(count=n()) %>% mutate(percent=round(count/
  sum(count),2)) %>% ggplot(aes(x=!sym(crash),y=percent))+
  geom_bar(stat='identity')+coord_flip()+xlim(c(-1,a))+ylim(c(0,b))+
  ggtitle(paste("Distribution_of_4SG", crash, "Counts_without", v))
g2 = dat_all %>% filter(!sym(v)==1,INTERTYPE=='4SG') %>% group_by(!
  sym(crash)) %>% summarise(count=n()) %>% mutate(percent=round(count/
  sum(count),2)) %>% ggplot(aes(x=!sym(crash),y=percent))+
  geom_bar(stat='identity')+coord_flip()+xlim(c(-1,a))+ylim(c(0,b))+
  ggtitle(paste("Distribution_of_4SG", crash, "Counts_with", v))

g <- arrangeGrob(g1,g2,nrow=2)
ggsave(file=paste("../.. /report/image/", v, "_4SG_", crash, ".png", sep=""
), g)

}
getcatgraph("LIGHTING", "PDO")
getcatgraph("LIGHTING", "KA")
getcatgraph("LIGHTING", "BC")
table(dat_all$LIGHTING)
### 1-2-6 Accident VS APPROACHLEFT_TURN

table(dat_all$APPROACH_LEFTTURN, dat_all$INTERTYPE)
getcatgraph("APPROACH_LEFTTURN", "PDO")
# too less obs in PDO 4ST
getcatgraph("APPROACH_LEFTTURN", "KA")
getcatgraph("APPROACH_LEFTTURN", "BC")

# not partitioned by crashtype
g1 = dat_all %>% mutate(APPROACH_LEFTTURN=as.character(APPROACH_LEFTTURN)
) %>%
  group_by(APPROACH_LEFTTURN, INTERTYPE) %>%
  summarise(crash_percent = mean(crash)) %>%
  ggplot(aes(x=APPROACH_LEFTTURN, y=crash_percent, fill=APPROACH_LEFTTURN))
  +geom_bar(stat='identity')+
  facet_grid(~INTERTYPE)+ggtitle('Distribution_of_Crash-rate_over_
  Approach_Left-Turn_across_Intersection_Type')
ggsave(file="../.. /report/image/approach-left-turn.png", g1)

g1 = dat_all %>% mutate(APPROACH_LEFTTURN=as.character(APPROACH_LEFTTURN)
) %>%
  group_by(APPROACH_LEFTTURN) %>%
  summarise(crash_percent = mean(crash)) %>%

```

```

ggplot(aes(x=APPROACH_LEFTTURN,y=crash_percent,fill=APPROACH_LEFTTURN))
+geom_bar(stat='identity')+ggtitle('Overall_Distribution_of_Crash-
rate_over_Approach_Left-Turn')
ggsave(file="/.../report/image/approach-left-turn-all.png", g1)

### 1-2-6 Accident VS APPROACHLEFTTRUE
table(dat_all$APPROACH_RIGHTTURN,dat_all$INTERTYPE)
getcatgraph("APPROACH_RIGHTTURN","PDO")
# too less obs in PDO 4ST
getcatgraph("APPROACH_RIGHTTURN","KA")
getcatgraph("APPROACH_RIGHTTURN","BC")

# not partitioned by crashtype
g1 = dat_all %>% mutate(APPROACH_RIGHTTURN=as.character(APPROACH_
RIGHTTURN)) %>%
group_by(APPROACH_RIGHTTURN,INTERTYPE) %>%
summarise(crash_percent = mean(crash)) %>%
ggplot(aes(x=APPROACH_RIGHTTURN,y=crash_percent,fill=APPROACH_RIGHTTURN
))+geom_bar(stat='identity')+
facet_grid(~INTERTYPE)+ggtitle('Distribution_of_Crash-rate_over_
Approach_Right-Turn_across_Intersection_Type')
ggsave(file="/.../report/image/approach-right-turn.png", g1)

g1 = dat_all %>% mutate(APPROACH_RIGHTTURN=as.character(APPROACH_
RIGHTTURN)) %>%
group_by(APPROACH_RIGHTTURN) %>%
summarise(crash_percent = mean(crash)) %>%
ggplot(aes(x=APPROACH_RIGHTTURN,y=crash_percent,fill=APPROACH_RIGHTTURN
))+geom_bar(stat='identity')+ggtitle('Overall_Distribution_of_Crash-
rate_over_Approach_Right-Turn')
ggsave(file="/.../report/image/approach-right-turn-all.png", g1)
### 1-2-7 Accident VS APPROATRUE
dat_all$turn = as.numeric(dat_all$APPROACH_LEFTTURN+dat_all$APPROACH_
RIGHTTURN>0)
table(dat_all$turn,dat_all$INTERTYPE)

# not partitioned by crashtype
g1 = dat_all %>% mutate(turn=as.character(turn)) %>%
group_by(turn,INTERTYPE) %>%
summarise(crash_percent = mean(crash)) %>%
ggplot(aes(x=turn,y=crash_percent,fill=turn))+geom_bar(stat='identity')
+
facet_grid(~INTERTYPE)+ggtitle('Distribution_of_Crash-rate_over_
Approach_Turn_across_Intersection_Type')
ggsave(file="/.../report/image/approach-turn.png", g1)

g1 = dat_all %>% mutate(turn=as.character(turn)) %>%
group_by(turn) %>%
summarise(crash_percent = mean(crash)) %>%
ggplot(aes(x=turn,y=crash_percent,fill=turn))+geom_bar(stat='identity')

```

```
+ggtitle('Overall_Distribution_of_Crash-rate_over_Approach_Turn')
ggsave(file = "../report/image/approach-turn-all.png", g1)
```

### 6.2.2 Inference

```
\lstset{language=R, breaklines=true}
##Data import
dat_1 = read.csv('../data/3ST.csv')
dat_2 = read.csv('../data/4ST.csv')
dat_3 = read.csv('../data/4SG.csv')

### Question 1: Simplest question, do these different types of
intersection holds different number of crashes? (Measured by mean or
median.)

### About PDO

st3_pdo = dat_1$PDO
st4_pdo = dat_2$PDO
sg4_pdo = dat_3$PDO

### Check same dispersion assumption for Mann Whitney test: only st4 vs
sg4 group satisfied

ansari.test(st3_pdo, st4_pdo)
ansari.test(st4_pdo, sg4_pdo)
ansari.test(st3_pdo, sg4_pdo)

### Check st3_pdo with st4_pdo:

library(magrittr)
X <- data.frame(value = data.matrix(st3_pdo), label = "st3")
Y <- data.frame(value = data.matrix(st4_pdo), label = "st4")
data_XY <- rbind(X, Y)[order(rbind(X, Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)
W <- data_XY[data_XY$label == 'st4',] %>% dplyr::select("rank") %>% sum
n = length(st4_pdo)
m = length(st3_pdo)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n / 12) * ((m+n+1) - 1 / ((m+n) * (m+n-1)) * (191 * (191^2 - 1) + 78 * (78^2 - 1)
+ 61 * (61^2 - 1) + 32 * (32^2 - 1) + 25 * (25^2 - 1) + 17 * (17^2 - 1) + 10 * (10^2 - 1) + 7 * (7^2 - 1)
+ 9 * (9^2 - 1) + 6 * (6^2 - 1) + 2 * (2^2 - 1) + 3 * (3^2 - 1)))
T_ <- abs((W - E_0_W) / sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

### Check by 95% bootstrap CI. 95% confident CI does not contain 0 and
both bounds are negative
bootci <- function(x, y) {
  nl = length(x)
```

```

n2 = length(y)
dif = NULL
for(i in 1:1000){
  x1 = base::sample(x,n1,replace=TRUE)
  x2 = base::sample(y,n2,replace=TRUE)
  dif = c(dif, mean(x1)-mean(x2))
}
return(quantile(dif,prob=c(0.025,0.975)))
}
set.seed(1)
bootci(st3_pdo,st4_pdo)

### It returns true, so we think there is difference.

### Check st4_pdo with sg4_pdo:

X <- data.frame(value = data.matrix(st4_pdo), label = "st4")
Y <- data.frame(value = data.matrix(sg4_pdo), label = "sg4")
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)
W <- data_XY[data_XY$label=='sg4',] %>% dplyr::select("rank") %>% sum
n = length(sg4_pdo)
m = length(st4_pdo)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n/12)*((m+n+1)-1/((m+n)*(m+n-1)))*(36*(36^2-1)+13*(13^2-1)
+12*(12^2-1)+16*(16^2-1)+10*(10^2-1)+8*(8^2-1)+13*(13^2-1)+10*(10^2-1)
+9*(9^2-1)+2*(2^2-1)+3*(3^2-1)+2*(2^2-1)+5*(5^2-1)+2*3*(3^2-1)+2*
(2^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

### Also return true, so we think there is difference.

### Check st3_pdo with sg4_pdo:

X <- data.frame(value = data.matrix(st3_pdo), label = "st3")
Y <- data.frame(value = data.matrix(sg4_pdo), label = "sg4")
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)
W <- data_XY[data_XY$label=='sg4',] %>% dplyr::select("rank") %>% sum
n = length(sg4_pdo)
m = length(st3_pdo)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n/12)*((m+n+1)-1/((m+n)*(m+n-1)))*(191*(191^2-1)+75*(75^2-1)
+53*(53^2-1)+30*(30^2-1)+27*(27^2-1)+23*(23^2-1)+19*(19^2-1)+2*13*
(13^2-1)+9*(9^2-1)+3*2*(2^2-1)+4*(4^2-1)+5*(5^2-1)+6*(6^2-1)+2*3*
(3^2-1)))

```

```

T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

### Also return true, so we think there is difference.

### Check by 95% bootstrap CI. 95% confident CI does not contain 0 and
  both bounds are negative
set.seed(1)
bootci(st3_pdo, sg4_pdo)

### next, we use the categorical analysis perspective to verify the
  independence holds false.

st3_crash = c(nrow(dat_1[dat_1$PDO==0,]), nrow(dat_1[dat_1$PDO>0,]), nrow(
  dat_1[dat_1$PDO==0,])+nrow(dat_1[dat_1$PDO>0,]))
st4_crash = c(nrow(dat_2[dat_2$PDO==0,]), nrow(dat_2[dat_2$PDO>0,]), nrow(
  dat_2[dat_2$PDO==0,])+nrow(dat_2[dat_2$PDO>0,]))
sg4_crash = c(nrow(dat_3[dat_3$PDO==0,]), nrow(dat_3[dat_3$PDO>0,]), nrow(
  dat_3[dat_3$PDO==0,])+nrow(dat_3[dat_3$PDO>0,]))
total = st3_crash + st4_crash + sg4_crash
tbl <- data.frame(st3_crash, st4_crash, sg4_crash, total)
rownames(tbl) <- c('Safe', "Crash", "total")
tbl
Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i, j] - (tbl[i, 4] * tbl[3, j] / tbl[3, 4]))^2 / (tbl[i, 4] * tbl[3, j] /
      tbl[3, 4])
  }
}
Q >= qchisq(0.95, df=(2*1))

### We get the same result: intersection type and DPO crash number are
  relative.

### About KA

st3_ka = dat_1$KA
st4_ka = dat_2$KA
sg4_ka = dat_3$KA

### Check same dispersion assumption for Mann Whitney test

ansari.test(st3_ka, st4_ka)
ansari.test(st4_ka, sg4_ka)
ansari.test(st3_ka, sg4_ka)

### Check st3_ka with st4_ka:

```



```

X <- data.frame(value = data.matrix(st3_ka), label = "st3")
Y <- data.frame(value = data.matrix(st4_ka), label = "st4")
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)
W <- data_XY[data_XY$label=='st4',] %>% dplyr::select("rank") %>% sum
n = length(st4_ka)
m = length(st3_ka)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n/12) * ((m+n+1)-1 / ((m+n) * (m+n-1)) * (425*(425^2-1)+19*(19^2-1)
+2*(2^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

```

*### It returns true, so we think there is difference.*

*### Check by 95% bootstrap CI. 95% confident CI does contain*

```

set.seed(1)
bootci(st3_ka, st4_ka)

```

*### Check st4\_ka with sg4\_ka:*

```

X <- data.frame(value = data.matrix(st4_ka), label = "st4")
Y <- data.frame(value = data.matrix(sg4_ka), label = "sg4")
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)
W <- data_XY[data_XY$label=='sg4',] %>% dplyr::select("rank") %>% sum
n = length(sg4_ka)
m = length(st4_ka)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n/12) * ((m+n+1)-1 / ((m+n) * (m+n-1)) * (138*(138^2-1)+22*(22^2-1)
+3*(3^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

```

*### Then, this time, we found that the st4 and sg4 doesn't pose a strong effect on the ka crash number.*

*### Check by 95% bootstrap CI. 95% confident CI does contain 0.*

```

set.seed(1)
bootci(st4_ka, sg4_ka)

```

*### Check st3\_pdo with sg4\_pdo:*

```

X <- data.frame(value = data.matrix(st3_ka), label = "st3")
Y <- data.frame(value = data.matrix(sg4_ka), label = "sg4")

```

```

data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)
W <- data_XY[data_XY$label=='sg4',] %>% dplyr::select("rank") %>% sum
n = length(sg4_ka)
m = length(st3_ka)
E_0_W <- n * (m+n+1)/2
var_0_W <- (m*n/12)*((m+n+1)-1/((m+n)*(m+n-1))*(453*(453^2-1)+31*(31^2-1)
+3*(3^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

```

*### Returns true, so we think there is difference.*

*### Check by 95% bootstrap CI. 95% confident CI does not contain 0.*

```

set.seed(1)
bootci(st3_ka, sg4_ka)

```

*### next, we use the categorical analysis perspective to verify the independence holds false.*

```

st3_crash = c(nrow(dat_1[dat_1$KA==0,]),nrow(dat_1[dat_1$KA>0,]),nrow(dat_1[dat_1$KA==0,])+nrow(dat_1[dat_1$KA>0,]))
st4_crash = c(nrow(dat_2[dat_2$KA==0,]),nrow(dat_2[dat_2$KA>0,]),nrow(dat_2[dat_2$KA==0,])+nrow(dat_2[dat_2$KA>0,]))
sg4_crash = c(nrow(dat_3[dat_3$KA==0,]),nrow(dat_3[dat_3$KA>0,]),nrow(dat_3[dat_3$KA==0,])+nrow(dat_3[dat_3$KA>0,]))
total = st3_crash + st4_crash + sg4_crash
tbl <- data.frame(st3_crash, st4_crash, sg4_crash, total)
rownames(tbl) <- c('Safe', "Crash", "total")
tbl
Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i,j]-(tbl[i,4]*tbl[3,j]/tbl[3,4]))^2/(tbl[i,4]*tbl[3,j]/tbl[3,4])
  }
}
Q >= qchisq(0.95, df=(2*1))

```

*### About BC*

```

st3_bc = dat_1$BC
st4_bc = dat_2$BC
sg4_bc = dat_3$BC

```

*### Check same dispersion assumption:*

```

ansari.test(st3_bc, st4_bc)
ansari.test(st4_bc, sg4_bc)
ansari.test(st3_bc, sg4_bc)

### Check st3_bc with st4_bc:

X <- data.frame(value = data.matrix(st3_bc), label = "st3")
Y <- data.frame(value = data.matrix(st4_bc), label = "st4")
data_XY <- rbind(X, Y)[order(rbind(X, Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)
W <- data_XY[data_XY$label == 'st4',] %>% dplyr::select("rank") %>% sum
n = length(st4_bc)
m = length(st3_bc)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n/12) * ((m+n+1)-1 / ((m+n)*(m+n-1)) * (295*(295^2-1)+78*(78^2-1)
+34*(34^2-1)+23*(23^2-1)+6*(6^2-1)+4*(4^2-1)+3*(3^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

### It returns true, so we think there is difference.

### Check by 95% bootstrap CI. 95% confident CI does not contain 0.
set.seed(1)
bootci(st3_bc, st4_bc)

### Check st4_bc with sg4_bc:

X <- data.frame(value = data.matrix(st4_bc), label = "st4")
Y <- data.frame(value = data.matrix(sg4_bc), label = "sg4")
data_XY <- rbind(X, Y)[order(rbind(X, Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)
W <- data_XY[data_XY$label == 'sg4',] %>% dplyr::select("rank") %>% sum
n = length(sg4_bc)
m = length(st4_bc)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n/12) * ((m+n+1)-1 / ((m+n)*(m+n-1)) * (61*(61^2-1)+33*(33^2-1)
+20*(20^2-1)+14*(14^2-1)+11*(11^2-1)+10*(10^2-1)+5*(5^2-1)+4*(4^2-1)+3
*(3^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

### Also return true, so we think there is difference.
### Check by 95% bootstrap CI. 95% confident CI does not contain 0.
set.seed(1)
bootci(st4_bc, sg4_bc)

```

```
### Check st3_bc with sg4_bc:
```

```
X <- data.frame(value = data.matrix(st3_bc), label = "st3")
Y <- data.frame(value = data.matrix(sg4_bc), label = "sg4")
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)
W <- data_XY[data_XY$label=='sg4',] %>% dplyr::select("rank") %>% sum
n = length(sg4_bc)
m = length(st3_bc)
E_0_W <- n * (m+n+1)/2
var_0_W <- (m*n/12)*((m+n+1)-1/((m+n)*(m+n-1))*(297*(297^2-1)+77*(77^2-1)
+44*(44^2-1)+25*(25^2-1)+15*(15^2-1)+14*(14^2-1)+5*(5^2-1)+7*(7^2-1)+2
*2*(2^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit
```

```
### Returns true, so we think there is difference.
```

```
### Check by 95% bootstrap CI. 95% confident CI does not contain 0.
```

```
set.seed(1)
bootci(st3_bc, sg4_bc)
```

```
### next, we use the categorical analysis perspective to verify the
independence holds false.
```

```
st3_crash = c(nrow(dat_1[dat_1$BC==0,]),nrow(dat_1[dat_1$BC>0,]),nrow(dat
_1[dat_1$BC==0,])+nrow(dat_1[dat_1$BC>0,]))
st4_crash = c(nrow(dat_2[dat_2$BC==0,]),nrow(dat_2[dat_2$BC>0,]),nrow(dat
_2[dat_2$BC==0,])+nrow(dat_2[dat_2$BC>0,]))
sg4_crash = c(nrow(dat_3[dat_3$BC==0,]),nrow(dat_3[dat_3$BC>0,]),nrow(dat
_3[dat_3$BC==0,])+nrow(dat_3[dat_3$BC>0,]))
total = st3_crash + st4_crash + sg4_crash
tbl <- data.frame(st3_crash, st4_crash, sg4_crash, total)
rownames(tbl) <- c('Safe', "Crash", "total")
tbl
Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i,j]-(tbl[i,4]*tbl[3,j]/tbl[3,4]))^2/(tbl[i,4]*tbl[3,j]/
tbl[3,4])
  }
}
Q >= qchisq(0.95, df=(2*1))
#
```

```
#####
```

### Question 2: We have already known the crossing type affects the crash number, so we now explore whether cross type + another property may effects the crash number.

## Cross type + discrete skewed angle

#

#####

## Firstly, we make skewed angle discrete:

```
dat_1$SKEW_ANGLE = cut(dat_1$SKEW_ANGLE, breaks=c(-1,31,61,91), labels=c(1,2,3))
```

```
dat_2$SKEW_ANGLE = cut(dat_2$SKEW_ANGLE, breaks=c(-1,31,61,91), labels=c(1,2,3))
```

```
dat_3$SKEW_ANGLE = cut(dat_3$SKEW_ANGLE, breaks=c(-1,31,61,91), labels=c(1,2,3))
```

## Forming the table (PDO):

```
st3_1_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==1)]),
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==1)]),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==1)]) +
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==1)]))
```

```
st3_2_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==2)]),
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==2)]),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==2)]) +
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==2)]))
```

```
st3_3_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==3)]),
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==3)]),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==3)]) +
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==3)]))
```

```
st4_1_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==1)]),
               nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==1)]),
               nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==1)]) +
               nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==1)]))
```

```
st4_2_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==2)]),
               nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==2)]),
               nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==2)]) +
               nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==2)]))
```

```
st4_3_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==3)]),
               nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==3)]),
               nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==3)]) +
               nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==3)]))
```

```
sg4_1_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==1)]),
               nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==1)]),
               nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==1)]) +
               nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==1)]))
```

```
sg4_2_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==2)]),
               nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==2)]),
```

```

      nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==2),]) +
      nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==2),]) )
sg4_3_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==3),]),
      nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==3),]),
      nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==3),]) +
      nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==3),]) )
total = st3_1_crash + st3_2_crash + st3_3_crash + st4_1_crash + st4_2_
      crash + st4_3_crash + sg4_1_crash + sg4_2_crash + sg4_3_crash
tbl <- data.frame(st3_1_crash, st3_2_crash, st3_3_crash, st4_1_crash, st4
      _2_crash, st4_3_crash, sg4_1_crash,
      sg4_2_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe','Crash','total')
tbl

```

### Conduct chi-square test:

```

Q = 0
for (i in 1:2){
  for (j in 1:8){
    Q = Q + (tbl[i,j]-(tbl[i,9]*tbl[3,j]/tbl[3,9]))^2/(tbl[i,9]*tbl[3,j]/
      tbl[3,9])
  }
}
Q >= qchisq(0.95,df=(7*1))
1-pchisq(Q,df=(7*1))

```

### It returns true, so there exist correlation.

```

#
#####

```

#For BC

```

dat_1$SKEW_ANGLE = cut(dat_1$SKEW_ANGLE, breaks=c(-1,31,61,91), labels=c
      (1,2,3))
dat_2$SKEW_ANGLE = cut(dat_2$SKEW_ANGLE, breaks=c(-1,31,61,91), labels=c
      (1,2,3))
dat_3$SKEW_ANGLE = cut(dat_3$SKEW_ANGLE, breaks=c(-1,31,61,91), labels=c
      (1,2,3))

```

## Forming the table (BC):

```

st3_1_crash = c(nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==1),]),
      nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==1),]),
      nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==1),]) +
      nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==1),]) )
st3_2_crash = c(nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==2),]),
      nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==2),]),
      nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==2),]) +
      nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==2),]) )

```

```

st3_3_crash = c(nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==3)],),
               nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==3)],),
               nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==3)],) +
               nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==3)],))
st4_1_crash = c(nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==1)],),
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==1)],),
               nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==1)],) +
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==1)],))
st4_2_crash = c(nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==2)],),
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==2)],),
               nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==2)],) +
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==2)],))
st4_3_crash = c(nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==3)],),
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==3)],),
               nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==3)],) +
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==3)],))
sg4_1_crash = c(nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==1)],),
               nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==1)],),
               nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==1)],) +
               nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==1)],))
sg4_2_crash = c(nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==2)],),
               nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==2)],),
               nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==2)],) +
               nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==2)],))
sg4_3_crash = c(nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==3)],),
               nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==3)],),
               nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==3)],) +
               nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==3)],))
total = st3_1_crash + st3_2_crash + st3_3_crash + st4_1_crash + st4_2_
        crash + st4_3_crash + sg4_1_crash + sg4_2_crash + sg4_3_crash
tbl <- data.frame(st3_1_crash, st3_2_crash, st3_3_crash, st4_1_crash, st4
        _2_crash, st4_3_crash, sg4_1_crash,
        sg4_2_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

```

### Conduct chi-square test:

```

Q = 0
for (i in 1:2){
  for (j in 1:8){
    Q = Q + (tbl[i,j]-(tbl[i,9]*tbl[3,j]/tbl[3,9]))^2/(tbl[i,9]*tbl[3,j]/
    tbl[3,9])
  }
}
Q >= qchisq(0.95, df=(7*1))
1-pchisq(Q, df=(7*1))

```

### It returns true, so there exist correlation.

*## Forming the table (KA):*

```

st3_1_crash = c(nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==1)],),
               nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==1)],),
               nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==1)],) +
               nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==1)],))
st3_2_crash = c(nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==2)],),
               nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==2)],),
               nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==2)],) +
               nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==2)],))
st3_3_crash = c(nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==3)],),
               nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==3)],),
               nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==3)],) +
               nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==3)],))
st4_1_crash = c(nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==1)],),
               nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==1)],),
               nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==1)],) +
               nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==1)],))
st4_2_crash = c(nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==2)],),
               nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==2)],),
               nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==2)],) +
               nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==2)],))
st4_3_crash = c(nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==3)],),
               nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==3)],),
               nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==3)],) +
               nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==3)],))
sg4_1_crash = c(nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==1)],),
               nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==1)],),
               nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==1)],) +
               nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==1)],))
sg4_2_crash = c(nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==2)],),
               nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==2)],),
               nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==2)],) +
               nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==2)],))
sg4_3_crash = c(nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==3)],),
               nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==3)],),
               nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==3)],) +
               nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==3)],))
total = st3_1_crash + st3_2_crash + st3_3_crash + st4_1_crash + st4_2_
        crash + st4_3_crash + sg4_1_crash + sg4_2_crash + sg4_3_crash
tbl <- data.frame(st3_1_crash, st3_2_crash, st3_3_crash, st4_1_crash, st4_
        _2_crash, st4_3_crash, sg4_1_crash,
                  sg4_2_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

```



```
### Conduct chi-square test:
```

```
Q = 0
for (i in 1:2){
  for (j in 1:8){
    Q = Q + (tbl[i,j]-(tbl[i,9]*tbl[3,j]/tbl[3,9]))^2/(tbl[i,9]*tbl[3,j]/
    tbl[3,9])
  }
}
Q >= qchisq(0.95,df=(7*1))
1-pchisq(Q,df=(7*1))
```

```
### It returns true, so there exist correlation.
```

```
#
```

```
#####
```

```
## Seperate check for skew + intersection v.s. crash
```

```
 #(pdo + 3st)
st3_1_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==1)],),
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==1)],),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==1)],) +
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==1)],))
st3_2_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==2)],),
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==2)],),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==2)],) +
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==2)],))
st3_3_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==3)],),
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==3)],),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$SKEW_ANGLE==3)],) +
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$SKEW_ANGLE==3)],))
total = st3_1_crash + st3_2_crash + st3_3_crash
tbl <- data.frame(st3_1_crash, st3_2_crash, st3_3_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe','Crash','total')
tbl
```

```
Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i,j]-(tbl[i,4]*tbl[3,j]/tbl[3,4]))^2/(tbl[i,4]*tbl[3,j]/
    tbl[3,4])
  }
}
1-pchisq(Q,df=2)
```

```
 #(pdo+4st)
```

```
st4_1_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==1)],),
```

```

      nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==1)],),
      nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==1)],) +
      nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==1)],))
st4_2_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==2)],),
      nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==2)],),
      nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==2)],) +
      nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==2)],))
st4_3_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==3)],),
      nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==3)],),
      nrow(dat_2[(dat_2$PDO==0) & (dat_2$SKEW_ANGLE==3)],) +
      nrow(dat_2[(dat_2$PDO>0) & (dat_2$SKEW_ANGLE==3)],))
total = st4_1_crash + st4_2_crash + st4_3_crash
tbl <- data.frame(st4_1_crash, st4_2_crash, st4_3_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i, j]-(tbl[i, 4]*tbl[3, j]/tbl[3, 4]))^2/(tbl[i, 4]*tbl[3, j]/
      tbl[3, 4])
  }
}
1-pchisq(Q, df=2)

#(pdo+4sg)
sg4_1_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==1)],),
      nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==1)],),
      nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==1)],) +
      nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==1)],))
sg4_2_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==2)],),
      nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==2)],),
      nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==2)],) +
      nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==2)],))
sg4_3_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==3)],),
      nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==3)],),
      nrow(dat_3[(dat_3$PDO==0) & (dat_3$SKEW_ANGLE==3)],) +
      nrow(dat_3[(dat_3$PDO>0) & (dat_3$SKEW_ANGLE==3)],))
total = sg4_1_crash + sg4_2_crash + sg4_3_crash
tbl <- data.frame(sg4_1_crash, sg4_2_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

Q = 0
for (i in 1:2){
  for (j in 1:2){
    Q = Q + (tbl[i, j]-(tbl[i, 3]*tbl[3, j]/tbl[3, 3]))^2/(tbl[i, 3]*tbl[3, j]/
      tbl[3, 3])
  }
}

```

```

    }
  }
  1-pchisq(Q, df=2)

#(bc + 3st)
st3_1_crash = c(nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==1)],),
               nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==1)],),
               nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==1)],) +
               nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==1)],))
st3_2_crash = c(nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==2)],),
               nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==2)],),
               nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==2)],) +
               nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==2)],))
st3_3_crash = c(nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==3)],),
               nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==3)],),
               nrow(dat_1[(dat_1$BC==0) & (dat_1$SKEW_ANGLE==3)],) +
               nrow(dat_1[(dat_1$BC>0) & (dat_1$SKEW_ANGLE==3)],))
total = st3_1_crash + st3_2_crash + st3_3_crash
tbl <- data.frame(st3_1_crash, st3_2_crash, st3_3_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i, j]-(tbl[i, 4]*tbl[3, j]/tbl[3, 4]))^2/(tbl[i, 4]*tbl[3, j]/
    tbl[3, 4])
  }
}
1-pchisq(Q, df=2)

#(bc+4st)
st4_1_crash = c(nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==1)],),
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==1)],),
               nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==1)],) +
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==1)],))
st4_2_crash = c(nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==2)],),
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==2)],),
               nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==2)],) +
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==2)],))
st4_3_crash = c(nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==3)],),
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==3)],),
               nrow(dat_2[(dat_2$BC==0) & (dat_2$SKEW_ANGLE==3)],) +
               nrow(dat_2[(dat_2$BC>0) & (dat_2$SKEW_ANGLE==3)],))
total = st4_1_crash + st4_2_crash + st4_3_crash
tbl <- data.frame(st4_1_crash, st4_2_crash, st4_3_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

```

```

Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i,j]-(tbl[i,4]*tbl[3,j]/tbl[3,4]))^2/(tbl[i,4]*tbl[3,j]/
      tbl[3,4])
  }
}
1-pchisq(Q, df=2)

```

```

#(bc+4sg)
sg4_1_crash = c(nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==1)],),
  nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==1)],),
  nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==1)],) +
  nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==1)],))
sg4_2_crash = c(nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==2)],),
  nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==2)],),
  nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==2)],) +
  nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==2)],))
sg4_3_crash = c(nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==3)],),
  nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==3)],),
  nrow(dat_3[(dat_3$BC==0) & (dat_3$SKEW_ANGLE==3)],) +
  nrow(dat_3[(dat_3$BC>0) & (dat_3$SKEW_ANGLE==3)],))
total = sg4_1_crash + sg4_2_crash + sg4_3_crash
tbl <- data.frame(sg4_1_crash, sg4_2_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

```

```

Q = 0
for (i in 1:2){
  for (j in 1:2){
    Q = Q + (tbl[i,j]-(tbl[i,3]*tbl[3,j]/tbl[3,3]))^2/(tbl[i,3]*tbl[3,j]/
      tbl[3,3])
  }
}
1-pchisq(Q, df=2)

```

```

#(KA + 3st)
st3_1_crash = c(nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==1)],),
  nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==1)],),
  nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==1)],) +
  nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==1)],))
st3_2_crash = c(nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==2)],),
  nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==2)],),
  nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==2)],) +
  nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==2)],))
st3_3_crash = c(nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==3)],),
  nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==3)],),
  nrow(dat_1[(dat_1$KA==0) & (dat_1$SKEW_ANGLE==3)],) +

```

```

      nrow(dat_1[(dat_1$KA>0) & (dat_1$SKEW_ANGLE==3),]))
total = st3_1_crash + st3_2_crash + st3_3_crash
tbl <- data.frame(st3_1_crash, st3_2_crash, st3_3_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i, j] - (tbl[i, 4] * tbl[3, j] / tbl[3, 4]))^2 / (tbl[i, 4] * tbl[3, j] /
      tbl[3, 4])
  }
}
1-pchisq(Q, df=2)

#(ka+4st)
st4_1_crash = c(nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==1),]),
  nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==1),]),
  nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==1),]) +
  nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==1),]))
st4_2_crash = c(nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==2),]),
  nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==2),]),
  nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==2),]) +
  nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==2),]))
st4_3_crash = c(nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==3),]),
  nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==3),]),
  nrow(dat_2[(dat_2$KA==0) & (dat_2$SKEW_ANGLE==3),]) +
  nrow(dat_2[(dat_2$KA>0) & (dat_2$SKEW_ANGLE==3),]))
total = st4_1_crash + st4_2_crash + st4_3_crash
tbl <- data.frame(st4_1_crash, st4_2_crash, st4_3_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i, j] - (tbl[i, 4] * tbl[3, j] / tbl[3, 4]))^2 / (tbl[i, 4] * tbl[3, j] /
      tbl[3, 4])
  }
}
1-pchisq(Q, df=2)

#(ka+4sg)
sg4_1_crash = c(nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==1),]),
  nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==1),]),
  nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==1),]) +
  nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==1),]))
sg4_2_crash = c(nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==2),]),

```

```

      nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==2)],),
      nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==2)],) +
      nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==2)],))
sg4_3_crash = c(nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==3)],),
      nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==3)],),
      nrow(dat_3[(dat_3$KA==0) & (dat_3$SKEW_ANGLE==3)],) +
      nrow(dat_3[(dat_3$KA>0) & (dat_3$SKEW_ANGLE==3)],))
total = sg4_1_crash + sg4_2_crash + sg4_3_crash
tbl <- data.frame(sg4_1_crash, sg4_2_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

```

```

Q = 0
for (i in 1:2){
  for (j in 1:2){
    Q = Q + (tbl[i,j]-(tbl[i,3]*tbl[3,j]/tbl[3,3]))^2/(tbl[i,3]*tbl[3,j]/
      tbl[3,3])
  }
}
1-pchisq(Q, df=2)

```

```

#### mixed 3st
st3_1_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$KA==0) & (dat_1$BC==0)
  & (dat_1$SKEW_ANGLE==1)],),
  nrow(dat_1[(dat_1$SKEW_ANGLE==1)],) - nrow(dat_1[(dat_1$
    PDO==0) & (dat_1$KA==0) & (dat_1$BC==0) & (dat_1$SKEW_
    ANGLE==1)],),
  nrow(dat_1[(dat_1$SKEW_ANGLE==1)],))
st3_2_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$KA==0) & (dat_1$BC==0)
  & (dat_1$SKEW_ANGLE==2)],),
  nrow(dat_1[(dat_1$SKEW_ANGLE==2)],) - nrow(dat_1[(dat_1$
    PDO==0) & (dat_1$KA==0) & (dat_1$BC==0) & (dat_1$SKEW_
    ANGLE==2)],),
  nrow(dat_1[(dat_1$SKEW_ANGLE==2)],))
st3_3_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$KA==0) & (dat_1$BC==0)
  & (dat_1$SKEW_ANGLE==3)],),
  nrow(dat_1[(dat_1$SKEW_ANGLE==3)],) - nrow(dat_1[(dat_1$
    PDO==0) & (dat_1$KA==0) & (dat_1$BC==0) & (dat_1$SKEW_
    ANGLE==3)],),
  nrow(dat_1[(dat_1$SKEW_ANGLE==3)],))
total = st3_1_crash + st3_2_crash + st3_3_crash
tbl <- data.frame(st3_1_crash, st3_2_crash, st3_3_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

```

```

Q = 0
for (i in 1:2){

```

```

    for (j in 1:3){
      Q = Q + (tbl[i,j]-(tbl[i,4]*tbl[3,j]/tbl[3,4]))^2/(tbl[i,4]*tbl[3,j]/
        tbl[3,4])
    }
  }
  1-pchisq(Q, df=2)

```

```

### Mixed 4t
st4_1_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$KA==0) & (dat_2$BC==0)
  & (dat_2$SKEW_ANGLE==1) ,]) ,
  nrow(dat_2[(dat_2$SKEW_ANGLE==1) ,]) - nrow(dat_2[(dat_2$
    PDO==0) & (dat_2$KA==0) & (dat_2$BC==0) & (dat_2$SKEW_
    ANGLE==1) ,]) ,
  nrow(dat_2[(dat_2$SKEW_ANGLE==1) ,]))
st4_2_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$KA==0) & (dat_2$BC==0)
  & (dat_2$SKEW_ANGLE==2) ,]) ,
  nrow(dat_2[(dat_2$SKEW_ANGLE==2) ,]) - nrow(dat_2[(dat_2$
    PDO==0) & (dat_2$KA==0) & (dat_2$BC==0) & (dat_2$SKEW_
    ANGLE==2) ,]) ,
  nrow(dat_2[(dat_2$SKEW_ANGLE==2) ,]))
st4_3_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$KA==0) & (dat_2$BC==0)
  & (dat_2$SKEW_ANGLE==3) ,]) ,
  nrow(dat_2[(dat_2$SKEW_ANGLE==3) ,]) - nrow(dat_2[(dat_2$
    PDO==0) & (dat_2$KA==0) & (dat_2$BC==0) & (dat_2$SKEW_
    ANGLE==3) ,]) ,
  nrow(dat_2[(dat_2$SKEW_ANGLE==3) ,]))
total = st4_1_crash + st4_2_crash + st4_3_crash
tbl <- data.frame(st4_1_crash, st4_2_crash, st4_3_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

```

```

Q = 0
for (i in 1:2){
  for (j in 1:3){
    Q = Q + (tbl[i,j]-(tbl[i,4]*tbl[3,j]/tbl[3,4]))^2/(tbl[i,4]*tbl[3,j]/
      tbl[3,4])
  }
}
1-pchisq(Q, df=2)

```

```

### Mixed 4sg
sg4_1_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$KA==0) & (dat_3$BC==0)
  & (dat_3$SKEW_ANGLE==1) ,]) ,
  nrow(dat_3[(dat_3$SKEW_ANGLE==1) ,]) - nrow(dat_3[(dat_3$
    PDO==0) & (dat_3$KA==0) & (dat_3$BC==0) & (dat_3$SKEW_
    ANGLE==1) ,]) ,
  nrow(dat_3[(dat_3$SKEW_ANGLE==1) ,]))
sg4_2_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$KA==0) & (dat_3$BC==0)

```

```

    & (dat_3$SKEW_ANGLE==2),)) ,
      nrow(dat_3[(dat_3$SKEW_ANGLE==2),]) - nrow(dat_3[(dat_3$
        PDO==0) & (dat_3$KA==0) & (dat_3$BC==0) & (dat_3$SKEW_
        ANGLE==2),]) ,
      nrow(dat_3[(dat_3$SKEW_ANGLE==2),]))
sg4_3_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$KA==0) & (dat_3$BC==0)
  & (dat_3$SKEW_ANGLE==3),]) ,
  nrow(dat_3[(dat_3$SKEW_ANGLE==3),]) - nrow(dat_3[(dat_3$
    PDO==0) & (dat_3$KA==0) & (dat_3$BC==0) & (dat_3$SKEW_
    ANGLE==3),]) ,
  nrow(dat_3[(dat_3$SKEW_ANGLE==3),]))
total = sg4_1_crash + sg4_2_crash + sg4_3_crash
tbl <- data.frame(sg4_1_crash, sg4_2_crash, total)
# we've drop sg4_3_crash, as its size = 0.
rownames(tbl) <- c('Safe', "Crash", "total")
tbl

```

```

Q = 0
for (i in 1:2){
  for (j in 1:2){
    Q = Q + (tbl[i,j]-(tbl[i,3]*tbl[3,j]/tbl[3,3]))^2/(tbl[i,3]*tbl[3,j]/
      tbl[3,3])
  }
}
1-pchisq(Q, df=2)

```

```

## Cross type + light (PDO)
#

```

```

#####

```

```

st3_0_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$LIGHTING==0),]) ,
  nrow(dat_1[(dat_1$PDO>0) & (dat_1$LIGHTING==0),]) ,
  nrow(dat_1[(dat_1$PDO==0) & (dat_1$LIGHTING==0),]) + nrow
  (dat_1[(dat_1$PDO>0) & (dat_1$LIGHTING==0),]))
st3_1_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$LIGHTING==1),]) ,
  nrow(dat_1[(dat_1$PDO>0) & (dat_1$LIGHTING==1),]) ,
  nrow(dat_1[(dat_1$PDO==0) & (dat_1$LIGHTING==1),]) + nrow
  (dat_1[(dat_1$PDO>0) & (dat_1$LIGHTING==1),]))
st4_0_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$LIGHTING==0),]) ,
  nrow(dat_2[(dat_2$PDO>0) & (dat_2$LIGHTING==0),]) ,
  nrow(dat_2[(dat_2$PDO==0) & (dat_2$LIGHTING==0),]) + nrow
  (dat_2[(dat_2$PDO>0) & (dat_2$LIGHTING==0),]))
st4_1_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$LIGHTING==1),]) ,
  nrow(dat_2[(dat_2$PDO>0) & (dat_2$LIGHTING==1),]) ,
  nrow(dat_2[(dat_2$PDO==0) & (dat_2$LIGHTING==1),]) + nrow
  (dat_2[(dat_2$PDO>0) & (dat_2$LIGHTING==1),]))
sg4_0_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$LIGHTING==0),]) ,
  nrow(dat_3[(dat_3$PDO>0) & (dat_3$LIGHTING==0),]) ,
  nrow(dat_3[(dat_3$PDO==0) & (dat_3$LIGHTING==0),]) + nrow

```



```

      (dat_3[(dat_3$PDO>0) & (dat_3$LIGHTING==0),]))
sg4_1_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$LIGHTING==1),]),
               nrow(dat_3[(dat_3$PDO>0) & (dat_3$LIGHTING==1),]),
               nrow(dat_3[(dat_3$PDO==0) & (dat_3$LIGHTING==1),]) + nrow
               (dat_3[(dat_3$PDO>0) & (dat_3$LIGHTING==1),]))
total = st3_0_crash + st3_1_crash + st4_0_crash + st4_1_crash + sg4_0_
      crash + sg4_1_crash
tbl <- data.frame(st3_0_crash, st3_1_crash, st4_0_crash, st4_1_crash, sg4
      _0_crash, sg4_1_crash, total)
rownames(tbl) <- c('Safe', "Crash", "total")
tbl
Q = 0
for (i in 1:2){
  for (j in 1:6){
    Q = Q + (tbl[i,j]-(tbl[i,7]*tbl[3,j]/tbl[3,7]))^2/(tbl[i,7]*tbl[3,j]/
      tbl[3,7])
  }
}
Q >= qchisq(0.95,df=(2*1))

```

### Cross type + turn

#

#####

```

st3_00_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$APPROACH_LEFTTURN==0)
      & (dat_1$APPROACH_RIGHTTURN==0),]),
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$APPROACH_LEFTTURN==0) &
               (dat_1$APPROACH_RIGHTTURN==0),]),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$APPROACH_LEFTTURN==0)
               & (dat_1$APPROACH_RIGHTTURN==0),]) + nrow(dat_1[(dat_1
               $PDO>0) & (dat_1$APPROACH_LEFTTURN==0) & (dat_1$
               APPROACH_RIGHTTURN==0),]))
st3_10_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$APPROACH_LEFTTURN==1)
      & (dat_1$APPROACH_RIGHTTURN==0),]),
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$APPROACH_LEFTTURN==1)
               & (dat_1$APPROACH_RIGHTTURN==0),]),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$APPROACH_LEFTTURN==1)
               & (dat_1$APPROACH_RIGHTTURN==0),]) + nrow(dat_1[(dat
               _1$PDO>0) & (dat_1$APPROACH_LEFTTURN==1) & (dat_1$
               APPROACH_RIGHTTURN==0),]))
st3_01_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$APPROACH_LEFTTURN==0)
      & (dat_1$APPROACH_RIGHTTURN==1),]),
               nrow(dat_1[(dat_1$PDO>0) & (dat_1$APPROACH_LEFTTURN==0)
               & (dat_1$APPROACH_RIGHTTURN==1),]),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$APPROACH_LEFTTURN==0)
               & (dat_1$APPROACH_RIGHTTURN==1),]) + nrow(dat_1[(dat
               _1$PDO>0) & (dat_1$APPROACH_LEFTTURN==0) & (dat_1$
               APPROACH_RIGHTTURN==1),]))

```

```

st3_11_crash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$APPROACH_LEFTTURN==1)
& (dat_1$APPROACH_RIGHTTURN==1) ,]) ,
nrow(dat_1[(dat_1$PDO>0) & (dat_1$APPROACH_LEFTTURN==1)
& (dat_1$APPROACH_RIGHTTURN==1) ,]) ,
nrow(dat_1[(dat_1$PDO==0) & (dat_1$APPROACH_LEFTTURN==1)
& (dat_1$APPROACH_RIGHTTURN==1) ,]) + nrow(dat_1[(dat
_1$PDO>0) & (dat_1$APPROACH_LEFTTURN==1) & (dat_1$
APPROACH_RIGHTTURN==1) ,]))

st4_00_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$APPROACH_LEFTTURN==0)
& (dat_2$APPROACH_RIGHTTURN==0) ,]) ,
nrow(dat_2[(dat_2$PDO>0) & (dat_2$APPROACH_LEFTTURN==0)
& (dat_2$APPROACH_RIGHTTURN==0) ,]) ,
nrow(dat_2[(dat_2$PDO==0) & (dat_2$APPROACH_LEFTTURN==0)
& (dat_2$APPROACH_RIGHTTURN==0) ,]) + nrow(dat_2[(dat
_2$PDO>0) & (dat_2$APPROACH_LEFTTURN==0) & (dat_2$
APPROACH_RIGHTTURN==0) ,]))

st4_10_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$APPROACH_LEFTTURN==1)
& (dat_2$APPROACH_RIGHTTURN==0) ,]) ,
nrow(dat_2[(dat_2$PDO>0) & (dat_2$APPROACH_LEFTTURN==1)
& (dat_2$APPROACH_RIGHTTURN==0) ,]) ,
nrow(dat_2[(dat_2$PDO==0) & (dat_2$APPROACH_LEFTTURN==1)
& (dat_2$APPROACH_RIGHTTURN==0) ,]) + nrow(dat_2[(dat
_2$PDO>0) & (dat_2$APPROACH_LEFTTURN==1) & (dat_2$
APPROACH_RIGHTTURN==0) ,]))

st4_01_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$APPROACH_LEFTTURN==0)
& (dat_2$APPROACH_RIGHTTURN==1) ,]) ,
nrow(dat_2[(dat_2$PDO>0) & (dat_2$APPROACH_LEFTTURN==0)
& (dat_2$APPROACH_RIGHTTURN==1) ,]) ,
nrow(dat_2[(dat_2$PDO==0) & (dat_2$APPROACH_LEFTTURN==0)
& (dat_2$APPROACH_RIGHTTURN==1) ,]) + nrow(dat_2[(dat
_2$PDO>0) & (dat_2$APPROACH_LEFTTURN==0) & (dat_2$
APPROACH_RIGHTTURN==1) ,]))

st4_11_crash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$APPROACH_LEFTTURN==1)
& (dat_2$APPROACH_RIGHTTURN==1) ,]) ,
nrow(dat_2[(dat_2$PDO>0) & (dat_2$APPROACH_LEFTTURN==1)
& (dat_2$APPROACH_RIGHTTURN==1) ,]) ,
nrow(dat_2[(dat_2$PDO==0) & (dat_2$APPROACH_LEFTTURN==1)
& (dat_2$APPROACH_RIGHTTURN==1) ,]) + nrow(dat_2[(dat
_2$PDO>0) & (dat_2$APPROACH_LEFTTURN==1) & (dat_2$
APPROACH_RIGHTTURN==1) ,]))

sg4_00_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$APPROACH_LEFTTURN==0)
& (dat_3$APPROACH_RIGHTTURN==0) ,]) ,
nrow(dat_3[(dat_3$PDO>0) & (dat_3$APPROACH_LEFTTURN==0)
& (dat_3$APPROACH_RIGHTTURN==0) ,]) ,
nrow(dat_3[(dat_3$PDO==0) & (dat_3$APPROACH_LEFTTURN==0)
& (dat_3$APPROACH_RIGHTTURN==0) ,]) + nrow(dat_3[(dat
_3$PDO>0) & (dat_3$APPROACH_LEFTTURN==0) & (dat_3$
APPROACH_RIGHTTURN==0) ,]))

```

```

sg4_10_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$APPROACH_LEFTTURN==1)
& (dat_3$APPROACH_RIGHTTURN==0) ,)] ,
nrow(dat_3[(dat_3$PDO>0) & (dat_3$APPROACH_LEFTTURN==1)
& (dat_3$APPROACH_RIGHTTURN==0) ,)] ,
nrow(dat_3[(dat_3$PDO==0) & (dat_3$APPROACH_LEFTTURN==1)
& (dat_3$APPROACH_RIGHTTURN==0) ,)] + nrow(dat_3[(dat
_3$PDO>0) & (dat_3$APPROACH_LEFTTURN==1) & (dat_3$
APPROACH_RIGHTTURN==0) ,]))
sg4_01_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$APPROACH_LEFTTURN==0)
& (dat_3$APPROACH_RIGHTTURN==1) ,)] ,
nrow(dat_3[(dat_3$PDO>0) & (dat_3$APPROACH_LEFTTURN==0)
& (dat_3$APPROACH_RIGHTTURN==1) ,)] ,
nrow(dat_3[(dat_3$PDO==0) & (dat_3$APPROACH_LEFTTURN==0)
& (dat_3$APPROACH_RIGHTTURN==1) ,)] + nrow(dat_3[(dat
_3$PDO>0) & (dat_3$APPROACH_LEFTTURN==0) & (dat_3$
APPROACH_RIGHTTURN==1) ,]))
sg4_11_crash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$APPROACH_LEFTTURN==1)
& (dat_3$APPROACH_RIGHTTURN==1) ,)] ,
nrow(dat_3[(dat_3$PDO>0) & (dat_3$APPROACH_LEFTTURN==1)
& (dat_3$APPROACH_RIGHTTURN==1) ,)] ,
nrow(dat_3[(dat_3$PDO==0) & (dat_3$APPROACH_LEFTTURN==1)
& (dat_3$APPROACH_RIGHTTURN==1) ,)] + nrow(dat_3[(dat
_3$PDO>0) & (dat_3$APPROACH_LEFTTURN==1) & (dat_3$
APPROACH_RIGHTTURN==1) ,]))

```

```

tbl <- data.frame(st3_00_crash, st3_01_crash, st3_10_crash, st3_11_crash, st4
_00_crash, st4_10_crash, st4_11_crash, sg4_00_crash, sg4_01_crash, sg4_10_
crash, sg4_11_crash, total)

```

```

rownames(tbl) <- c('Safe', "Crash", "total")

```

```

tbl

```

```

Q = 0

```

```

for (i in 1:2){

```

```

  for (j in 1:11){

```

```

    Q = Q + (tbl[i, j] - (tbl[i, 12] * tbl[3, j] / tbl[3, 12]))^2 / (tbl[i, 12] * tbl[3,
j] / tbl[3, 12])

```

```

  }

```

```

}

```

```

Q >= qchisq(0.95, df=(2*1))

```

```

### Cross type + AADT major

```

```

##PDO

```

```

#####

```

```

st3_aadt_m_pdocrash = dat_1[dat_1$PDO>0, c("AADT_MAJOR")]

```

```

st3_aadt_m_pdonocrash = dat_1[dat_1$PDO==0, c("AADT_MAJOR")]

```

```

st4_aadt_m_pdocrash = dat_2[dat_2$PDO>0, c("AADT_MAJOR")]

```

```

st4_aadt_m_pdonocrash = dat_2[dat_2$PDO==0, c("AADT_MAJOR")]

```

```

sg4_aadt_m_pdocrash = dat_3[dat_3$PDO>0, c("AADT_MAJOR")]

```

```

sg4_aadt_m_pdonocrash = dat_3[dat_3$PDO==0, c("AADT_MAJOR")]

```

```

### Comparing st3:
shapiro.test(st3_aadt_m_pdocrash)
shapiro.test(st3_aadt_m_pdonocrash)
shapiro.test(st4_aadt_m_pdocrash)
shapiro.test(st4_aadt_m_pdonocrash)
shapiro.test(sg4_aadt_m_pdocrash)
shapiro.test(sg4_aadt_m_pdonocrash)

### Only sg4_aadt_m_pdonocrash passes the normality test, while their
dispersions are similar, so we decide to conduct w-rank test.

### Check st3 (crash v.s. non crash)

X <- data.frame(value = data.matrix(st3_aadt_m_pdonocrash), label = "
nocrash")
Y <- data.frame(value = data.matrix(st3_aadt_m_pdocrash), label = "crash"
)
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)[table(data_XY$rank) != 1]
W <- data_XY[data_XY$label=='crash',] %>% dplyr::select("rank") %>% sum
n = length(st3_aadt_m_pdocrash)
m = length(st3_aadt_m_pdonocrash)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n/12)*((m+n+1)-1/((m+n)*(m+n-1))*(35*2*(2^2-1)+18*3*(3^2-1)
+9*4*(4^2-1)+10*5*(5^2-1)+6*6*(6^2-1)+3*7*(7^2-1)+2*9*(9^2-1)+4*10*
(10^2-1)+12*(12^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

### Yes, controlling crosstype, the major addt affects pdo crashing
number.

### Check st4 (crash v.s. non crash)

X <- data.frame(value = data.matrix(st4_aadt_m_pdonocrash), label = "
nocrash")
Y <- data.frame(value = data.matrix(st4_aadt_m_pdocrash), label = "crash"
)
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)[table(data_XY$rank) != 1]
W <- data_XY[data_XY$label=='crash',] %>% dplyr::select("rank") %>% sum
n = length(st4_aadt_m_pdocrash)
m = length(st4_aadt_m_pdonocrash)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n/12)*((m+n+1)-1/((m+n)*(m+n-1))*(13*2*(2^2-1)+2*3*(3^2-1))
)

```

```

T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

### This time, we think major addt doesn't affects pdo crashing number in
    st4 crossing.

### Check sg4 (crash v.s. non crash)

X <- data.frame(value = data.matrix(sg4_aadt_m_pdonocrash), label = "
    nocrash")
Y <- data.frame(value = data.matrix(sg4_aadt_m_pdocrash), label = "crash"
    )
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)[table(data_XY$rank)!= 1]
W <- data_XY[data_XY$label=='crash',] %>% dplyr::select("rank") %>% sum
n = length(sg4_aadt_m_pdocrash)
m = length(sg4_aadt_m_pdonocrash)
E_0_W <- n * (m+n+1)/2
var_0_W <- (m*n/12)*((m+n+1)-1/((m+n)*(m+n-1))*(21*2*(2^2-1)+3*(3^2-1)+4*
    (4^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

### And also, we don't have sufficient clue to show that major addt
    affects pdo crashing number in sg4 crossing.

### Cross type + AADT minor
#
#####

st3_aadt_m_pdocrash = dat_1[dat_1$PDO>0,c("AADT_MINOR")]
st3_aadt_m_pdonocrash = dat_1[dat_1$PDO==0,c("AADT_MINOR")]
st4_aadt_m_pdocrash = dat_2[dat_2$PDO>0,c("AADT_MINOR")]
st4_aadt_m_pdonocrash = dat_2[dat_2$PDO==0,c("AADT_MINOR")]
sg4_aadt_m_pdocrash = dat_3[dat_3$PDO>0,c("AADT_MINOR")]
sg4_aadt_m_pdonocrash = dat_3[dat_3$PDO==0,c("AADT_MINOR")]

### Comparing st3:
shapiro.test(st3_aadt_m_pdocrash)
shapiro.test(st3_aadt_m_pdonocrash)
shapiro.test(st4_aadt_m_pdocrash)
shapiro.test(st4_aadt_m_pdonocrash)
shapiro.test(sg4_aadt_m_pdocrash)
shapiro.test(sg4_aadt_m_pdonocrash)

```

*### Only sg4\_aadt\_m\_pdonocrash passes the normality test, while their dispersions are similar, so we decide to conduct w-rank test.*

*### Check st3 (crash v.s. non crash)*

```
X <- data.frame(value = data.matrix(st3_aadt_m_pdonocrash), label = "
  nocrash")
Y <- data.frame(value = data.matrix(st3_aadt_m_pdocrash), label = "crash"
  )
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)[table(data_XY$rank) != 1]
W <- data_XY[data_XY$label=='crash',] %>% dplyr::select("rank") %>% sum
n = length(st3_aadt_m_pdocrash)
m = length(st3_aadt_m_pdonocrash)
E_0_W <- n * (m+n+1)/2
var_0_W <- (m*n/12)*((m+n+1)-1/((m+n)*(m+n-1))*(6*2*(2^2-1)+9*3*(3^2-1)+8
  *4*(4^2-1)+4*5*(5^2-1)+4*6*(6^2-1)+4*7*(7^2-1)+5*8*(8^2-1)+3*9*(9^2-1)
  +2*10*(10^2-1)+2*11*(11^2-1)+2*12*(12^2-1)+13*(13^2-1)+3*14*(14^2-1)
  +15*(15^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit
```

*### Yes, controlling crosstype, the major addt affects pdo crashing number.*

*### Chech st4 (crash v.s. non crash)*

```
X <- data.frame(value = data.matrix(st4_aadt_m_pdonocrash), label = "
  nocrash")
Y <- data.frame(value = data.matrix(st4_aadt_m_pdocrash), label = "crash"
  )
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)[table(data_XY$rank) != 1]
W <- data_XY[data_XY$label=='crash',] %>% dplyr::select("rank") %>% sum
n = length(st4_aadt_m_pdocrash)
m = length(st4_aadt_m_pdonocrash)
E_0_W <- n * (m+n+1)/2
var_0_W <- (m*n/12)*((m+n+1)-1/((m+n)*(m+n-1))*(8*2*(2^2-1)+6*3*(3^2-1)+2
  *4*(4^2-1)+5*(5^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit
```

*### This time, we think major addt doesn't affects pdo crashing number in st4 crossing.*

*### Check sg4 (crash v.s. non crash)*

```

X <- data.frame(value = data.matrix(sg4_aadt_m_pdonocrash), label = "
  nocrash")
Y <- data.frame(value = data.matrix(sg4_aadt_m_pdocrash), label = "crash"
  )
data_XY <- rbind(X,Y)[order(rbind(X,Y)$value),]
data_XY$rank = rank(data_XY$value, ties.method = "average")
table(data_XY$rank)[table(data_XY$rank) != 1]
W <- data_XY[data_XY$label=='crash',] %>% dplyr::select("rank") %>% sum
n = length(sg4_aadt_m_pdocrash)
m = length(sg4_aadt_m_pdonocrash)
E_0_W <- n * (m+n+1) / 2
var_0_W <- (m*n/12) * ((m+n+1)-1 / ((m+n) * (m+n-1)) * (16*2*(2^2-1)+8*(3^2-1)+3*
  (4^2-1)))
T_ <- abs((W-E_0_W)/sqrt(var_0_W))
Z_crit <- qnorm(0.025, 0, 1, lower.tail = F)
abs(T_) > Z_crit

```

### For BC and KA with AADT, just applie the codes above.

#

#####

### We now conduct set of 2\*2 tables test for turns and lighting.

## For lighting, we forming the tables:

#st3:

```

st3_noncrash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$KA==0) & (dat_1$BC
  ==0) & (dat_1$LIGHTING==0)],),
  nrow(dat_1[(dat_1$PDO==0) & (dat_1$KA==0) & (dat_1$BC==0)
    & (dat_1$LIGHTING==1)],),
  nrow(dat_1[(dat_1$PDO==0) & (dat_1$KA==0) & (dat_1$BC==0)
    & (dat_1$LIGHTING==0)],) + nrow(dat_1[(dat_1$PDO==0)
    & (dat_1$KA==0) & (dat_1$BC==0) & (dat_1$LIGHTING==1)
    ],))

```

```

total = c(nrow(dat_1[ (dat_1$LIGHTING==0)],),
  nrow(dat_1[ (dat_1$LIGHTING==1)],),
  nrow(dat_1))

```

```

st3_crash = total - st3_noncrash

```

```

tbl_st3 <- data.frame(st3_noncrash, st3_crash, total)
rownames(tbl_st3) <- c('Light', "No_light", "total")
tbl_st3

```

#st4:

```

st4_noncrash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$KA==0) & (dat_2$BC

```

```

==0) & (dat_2$LIGHTING==0),)) ,
      nrow(dat_2[(dat_2$PDO==0) & (dat_2$KA==0) & (dat_2$BC
==0) & (dat_2$LIGHTING==1),)) ,
      nrow(dat_2[(dat_2$PDO==0) & (dat_2$KA==0) & (dat_2$BC
==0) & (dat_2$LIGHTING==0),)) + nrow(dat_2[(dat_2$PDO
==0) & (dat_2$KA==0) & (dat_2$BC==0) & (dat_2$
LIGHTING==1),))

total = c(nrow(dat_2[ (dat_2$LIGHTING==0),]) ,
          nrow(dat_2[ (dat_2$LIGHTING==1),]) ,
          nrow(dat_2))

st4_crash = total - st4_noncrash

tbl_st4 <- data.frame(st4_noncrash, st4_crash, total)
rownames(tbl_st4) <- c('Light', "No_light", "total")
tbl_st4

#sg4:
sg4_noncrash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$KA==0) & (dat_3$BC
==0) & (dat_3$LIGHTING==0),)) ,
      nrow(dat_3[(dat_3$PDO==0) & (dat_3$KA==0) & (dat_3$BC
==0) & (dat_3$LIGHTING==1),)) ,
      nrow(dat_3[(dat_3$PDO==0) & (dat_3$KA==0) & (dat_3$BC
==0) & (dat_3$LIGHTING==0),)) + nrow(dat_3[(dat_3$PDO
==0) & (dat_3$KA==0) & (dat_3$BC==0) & (dat_3$
LIGHTING==1),))

total = c(nrow(dat_3[ (dat_3$LIGHTING==0),]) ,
          nrow(dat_3[ (dat_3$LIGHTING==1),]) ,
          nrow(dat_3))

sg4_crash = total - sg4_noncrash

tbl_sg4 <- data.frame(sg4_noncrash, sg4_crash, total)
rownames(tbl_sg4) <- c('Light', "No_light", "total")
tbl_sg4

#QMH
nume = 0
deno = 0
df.list <- list(tbl_st3, tbl_st4, tbl_sg4)
for (tbl in df.list) {
  nume = nume + (tbl[1,1] - (tbl[1,3]*tbl[3,1])/tbl[3,3])
  deno = deno + (tbl[1,3]*tbl[2,3]*tbl[3,1]*tbl[3,2])/(tbl[3,3]*tbl[3,3]*
    (tbl[3,3]-1))
}
QMH = (nume^2)/deno
QMH >= qchisq(1-0.05, df=1)

```



*## It shows that we have no reason to refuse null hypothesis of MH test.*

#

#####

*### Now the MH test for turns.*

```
st3_noncrash = c(nrow(dat_1[(dat_1$PDO==0) & (dat_1$KA==0) & (dat_1$BC
==0) & (dat_1$APPROACH_LEFTTURN==0) & (dat_1$APPROACH_RIGHTTURN==0)
,]),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$KA==0) & (dat_1$BC
==0) & ((dat_1$APPROACH_LEFTTURN==1) | (dat_1$
APPROACH_RIGHTTURN==1)) ,]),
               nrow(dat_1[(dat_1$PDO==0) & (dat_1$KA==0) & (dat_1$BC
==0) & (dat_1$APPROACH_LEFTTURN==0) & (dat_1$APPROACH
_RIGHTTURN==0) ,]) + nrow(dat_1[(dat_1$PDO==0) & (dat_
1$KA==0) & (dat_1$BC==0) & ((dat_1$APPROACH_LEFTTURN
==1) | (dat_1$APPROACH_RIGHTTURN==1)) ,]))
```

```
total = c(nrow(dat_1[(dat_1$APPROACH_LEFTTURN==0) & (dat_1$APPROACH_
RIGHTTURN==0) ,]),
          nrow(dat_1[((dat_1$APPROACH_LEFTTURN==1) | (dat_1$APPROACH_
RIGHTTURN==1)) ,]),
          nrow(dat_1))
```

```
st3_crash = total - st3_noncrash
```

```
tbl_st3 <- data.frame(st3_noncrash, st3_crash, total)
rownames(tbl_st3) <- c('No_turns', 'Turns', 'total')
tbl_st3
```

*#st4:*

```
st4_noncrash = c(nrow(dat_2[(dat_2$PDO==0) & (dat_2$KA==0) & (dat_2$BC
==0) & (dat_2$APPROACH_LEFTTURN==0) & (dat_2$APPROACH_RIGHTTURN==0) ,])
,
               nrow(dat_2[(dat_2$PDO==0) & (dat_2$KA==0) & (dat_2$BC
==0) & ((dat_2$APPROACH_LEFTTURN==1) | (dat_2$
APPROACH_RIGHTTURN==1)) ,]),
               nrow(dat_2[(dat_2$PDO==0) & (dat_2$KA==0) & (dat_2$BC
==0) & (dat_2$APPROACH_LEFTTURN==0) & (dat_2$APPROACH
_RIGHTTURN==0) ,]) + nrow(dat_2[(dat_2$PDO==0) & (dat_
2$KA==0) & (dat_2$BC==0) & ((dat_2$APPROACH_LEFTTURN
==1) | (dat_2$APPROACH_RIGHTTURN==1)) ,]))
```

```
total = c(nrow(dat_2[ (dat_2$APPROACH_LEFTTURN==0) & (dat_2$APPROACH_
RIGHTTURN==0) ,]),
          nrow(dat_2[ ((dat_2$APPROACH_LEFTTURN==1) | (dat_2$APPROACH_
RIGHTTURN==1)) ,]),
```

```

      nrow(dat_2))

st4_crash = total - st4_noncrash

tbl_st4 <- data.frame(st4_noncrash, st4_crash, total)
rownames(tbl_st4) <- c('No_turns', "Turns", "total")
tbl_st4

#sg4
sg4_noncrash = c(nrow(dat_3[(dat_3$PDO==0) & (dat_3$KA==0) & (dat_3$BC
==0) & (dat_3$APPROACH_LEFTTURN==0) & (dat_3$APPROACH_RIGHTTURN==0)],)
,
      nrow(dat_3[(dat_3$PDO==0) & (dat_3$KA==0) & (dat_3$BC
==0) & ((dat_3$APPROACH_LEFTTURN==1) | (dat_3$
APPROACH_RIGHTTURN==1))]),) ,
      nrow(dat_3[(dat_3$PDO==0) & (dat_3$KA==0) & (dat_3$BC
==0) & (dat_3$APPROACH_LEFTTURN==0) & (dat_3$APPROACH
_RIGHTTURN==0)],) + nrow(dat_3[(dat_3$PDO==0) & (dat_
3$KA==0) & (dat_3$BC==0) & ((dat_3$APPROACH_LEFTTURN
==1) | (dat_3$APPROACH_RIGHTTURN==1))]),))

total = c(nrow(dat_3[ (dat_3$APPROACH_LEFTTURN==0) & (dat_3$APPROACH_
RIGHTTURN==0)],) ,
      nrow(dat_3[ ((dat_3$APPROACH_LEFTTURN==1) | (dat_3$APPROACH_
RIGHTTURN==1))]),) ,
      nrow(dat_3))

sg4_crash = total - sg4_noncrash

tbl_sg4 <- data.frame(sg4_noncrash, sg4_crash, total)
rownames(tbl_sg4) <- c('No_turns', "Turns", "total")
tbl_sg4

#QMH
nume = 0
deno = 0
df.list <- list(tbl_st3, tbl_st4, tbl_sg4)
for (tbl in df.list) {
  nume = nume + (tbl[1,1] - (tbl[1,3]*tbl[3,1])/tbl[3,3])
  deno = deno + (tbl[1,3]*tbl[2,3]*tbl[3,1]*tbl[3,2])/(tbl[3,3]*tbl[3,3]*
    (tbl[3,3]-1))
}
QMH = (nume^2)/deno
QMH >= qchisq(1-0.05, df=1)

## It also shows that we have no reason to refuse null hypothesis of MH
test.

### Conduct chi-square test on individual table
chisq.test(tbl_sg4[1:2,1:2], correct = FALSE)

```

*### Conduct Fisher-exact test on individual table*

```
fisher.test(tbl_st4[1:2,1:2], alternative = "two.sided", conf.int=FALSE)
```

### 6.2.3 Python - PSVM code (Implementing file)

```
import numpy as np
import pandas as pd
from inp import PSVM
from scipy.linalg import eigh

st3_tbl = pd.read_csv('../data/3ST.csv')
st4_tbl = pd.read_csv('../data/4ST.csv')
sg4_tbl = pd.read_csv('../data/4SG.csv')

tbl_total = pd.concat([pd.concat([st3_tbl, st4_tbl]), sg4_tbl])
X = tbl_total[['AADT_MAJOR', 'AADT_MINOR', 'LIGHTING', 'APPROACH_LEFTTURN',
               'APPROACH_RIGHTTURN', 'SKEW_ANGLE']].values
Y_PDO = tbl_total['PDO'].values
Y_BC = tbl_total['BC'].values
Y_KA = tbl_total['KA'].values

M_pdo = PSVM(X, Y_PDO, h=4, Lambda =1, printf=False)
lam, ve = eigh(M_pdo)

def select_d(lam):
    jump = []
    for i in range(len(lam)-1):
        jump.append(lam[-(i+1)]/lam[-(i+2)])
    return jump.index(max(jump))+1

d = select_d(lam)
Ve_pdo = ve[:, -d:]

M_bc = PSVM(X, Y_BC, h=4, Lambda =1, printf=False)
lam, ve = eigh(M_bc)

def select_d(lam):
    jump = []
    for i in range(len(lam)-1):
        jump.append(lam[-(i+1)]/lam[-(i+2)])
    return jump.index(max(jump))+1

d = select_d(lam)
Ve_bc = ve[:, -d:]

M_ka = PSVM(X, Y_KA, h=4, Lambda =1, printf=False)
lam, ve = eigh(M_ka)

def select_d(lam):
    jump = []
```

```

    for i in range(len(lam)-1):
        jump.append(lam[-(i+1)]/lam[-(i+2)])
    return jump.index(max(jump))+1

d = select_d(lam)
Ve_ka = ve[:, -d:]

6.2.4 Python - PSVM code (Support file)

import numpy as np

def sign2indict(x):
    x[np.where(x == 0)] = -1
    return x

def PSVM(X,Y,h,Lambda,printf):
    p = X.shape[1]
    n = X.shape[0]
    tile = np.percentile(Y, np.linspace(0, 100, h + 2)[1:-1].tolist())
    for i in range(h):
        locals()['Y' + str(i + 1)] = sign2indict(np.sign(Y - tile[i]))
    Sigma = np.cov(X.T)
    Lambda = Lambda
    def minussqrtm(M):
        (U,S,VT) = np.linalg.svd(M)
        D = np.diag(1/np.sqrt(S))
        return np.dot(np.dot(U,D),VT)
    Z = np.dot(X-X.mean(axis=0),minussqrtm(Sigma))
    from sklearn.svm import SVC
    for l in range(h):
        svc = SVC(C=Lambda/(2*n), kernel='linear')
        svc.fit(Z,locals()['Y'+ str(l+1)].ravel())
        locals()['beta' + str(l+1)] = np.dot(minussqrtm(Sigma),svc.coef_.
            reshape(p,1))
        if printf == True:
            print(np.concatenate([svc.intercept_.ravel(),locals()['beta'
                + str(l+1)].ravel()]))
    M = 0
    for i in range(h):
        M += np.dot(locals()['beta' + str(i + 1)].reshape(p, 1), locals()
            ['beta' + str(i + 1)].reshape(1, p))
    return M

```

## REFERENCES

- [1] LI, BING & ARTEMIOU, ANDREAS & LI, LEXIN(2011). Principal support vector machines for linear and nonlinear sufficient dimension reduction. *The Annals of Statistics*. **3182–3210**.
- [2] XIA, QIANG & XU, WANGLI & ZHU, LIXING(2015). Consistently determining the number of factors in multivariate volatility modeling. *Statistica Sinica*. **1025–1044**.