Motor Vehicle Crashes in Iowa (OpenData, Iowa Gov)

Group 2 - Jerry Jacob, Purna Chandra Kuntla, Sailaja Vuyyuru

Introduction to Dataset-test

The data is compiled from the Iowa Traffic Safety Data and Analysis website (www.iowadot.gov/tsda) we have 3 tables with 10 years of crash data. The tables are
crash_location:
crash_person:
crash_vehicle:
The crash_location table is the master table with the unique attribute, casenumber. The person table have multiple rows with every person involved in the crash. The vehicle table have multiple rows with each vehicle involved. All the tables are linked using the casenumber field. The data is for 2008 until 2018 (partial year).

Table Definitions

crash_location_raw table:

```
CREATE TABLE IF NOT EXISTS crash_location_raw
(
X double,
Y double,
OBJECTID int,
CRASH_KEY bigint,
CASENUMBER bigint,
LECASENUM String,
CRASH_DATE String,
CRASH_MONTH String,
CRASH_DAY String,
TIMESTR String,
DISTRICT int,
COUNTY_NUMBER int,
CITY_NUMBER int,
SYSTEMSTR String,
LITERAL String,
FRSTHARM String,
LOCFSTHRM String,
CRCOMNNR String,
MAJCSE String,
DRUGALC String,
ECNTCRC String,
LIGHT String,
CSRFCND String,
WEATHER String,
RCNTCRC String,
RDTYP String,
PAVED String,
WZRELATED String,
CSEV String,
FATALITIES int,
INJURIES int,
MAJINJURY int,
MININJURY int,
POSSINJURY int,
UNKINJURY int,
PROPDMG double,
VEHICLES int,
TOCCUPANTS int,
REPORT String,
XCOORD double,
YCOORD double,
REST_UPDATED String,
REST_UPDATE_UTC_OFFSET String,
CRASH_DATETIME String,
CRASH_DATETIME_UTC String,
CRASH_DATETIME_UTC_OFFSET String,
CITY_NAME String,
COUNTY_NAME  String
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
STORED AS TEXTFILE
tblproperties ("skip.header.line.count"="1");
```

```
Load data local inpath "project/Motor_Vehicle_Crashes/crash_location.csv" into table
crash_location_raw;
```

crash_location table:

```
Create Table crash_location
ROW FORMAT Delimited
STORED AS textfile
AS
select X,
Y,
OBJECTID,
CRASH_KEY,
CASENUMBER,
LECASENUM,
cast(from_unixtime(unix_timestamp(crash_date , "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'")) as timestamp) as
crash_date,
CRASH_MONTH,
CRASH_DAY,
TIMESTR,
DISTRICT,
COUNTY_NUMBER,
CITY_NUMBER,
SYSTEMSTR,
LITERAL,
FRSTHARM,
LOCFSTHRM,
CRCOMNNR,
MAJCSE,
DRUGALC,
ECNTCRC,
LIGHT,
CSRFCND,
WEATHER,
RCNTCRC,
RDTYP,
PAVED,
WZRELATED,
CSEV,
FATALITIES,
INJURIES,
MAJINJURY,
MININJURY,
POSSINJURY,
UNKINJURY,
PROPDMG,
VEHICLES,
TOCCUPANTS,
REPORT,
XCOORD,
YCOORD,
cast(from_unixtime(unix_timestamp(REST_UPDATED , "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'")) as
timestamp) as REST_UPDATED,
REST_UPDATE_UTC_OFFSET,
cast(from_unixtime(unix_timestamp(CRASH_DATETIME , "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'")) as
timestamp) as CRASH_DATETIME,
cast(from_unixtime(unix_timestamp(CRASH_DATETIME_UTC , "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"))
as timestamp) as CRASH_DATETIME_UTC,
CRASH_DATETIME_UTC_OFFSET,
CITY_NAME,
COUNTY_NAME
FROM crash_location_raw;
```

crash_location_p partition table:

```
CREATE TABLE IF NOT EXISTS crash_location_p
(
x             double,
y             double,
objectid         int,
crash_key          bigint,
casenumber          bigint,
lecasenum          string,
crash_date         timestamp,
crash_month          string,
crash_day          string,
timestr          string,
district        int,
county_number         int,
city_number         int,
systemstr         string,
literal        string,
frstharm          string,
locfsthrm         string,
crcomnnr           string,
majcse           string,
drugalc          string,
ecntcrc          string,
light          string,
csrfcnd               string,
weather               string,
rcntcrc               string,
rdtyp                 string,
paved                 string,
wzrelated             string,
csev                  string,
fatalities            int,
injuries              int,
majinjury             int,
mininjury             int,
possinjury            int,
unkinjury             int,
propdmg               double,
vehicles              int,
toccupants            int,
report                string,
xcoord                double,
ycoord                double,
rest_updated          timestamp,
rest_update_utc_offset  string,
crash_datetime        timestamp,
crash_datetime_utc      timestamp,
crash_datetime_utc_offset      string,
city_name             string,
county_name           string
)
PARTITIONED BY (crashmonth string)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
STORED AS TEXTFILE;


INSERT INTO TABLE crash_location_p Partition (crashmonth)

SELECT *, month(crash_date) as crashmonth FROM crash_location where crash_date is not NULL;
```

crash_vehicle table:

```
CREATE TABLE IF NOT EXISTS crash_vehicle
(
X double,
Y double,
OBJECTID int,
VEH_CRASH_KEY bigint,
VEH_UNITKEY bigint,
CASENUMBER bigint,
DRIVERAGE int,
DRIVERGEN varchar(10),
DL_STATE varchar(10),
CHARGED varchar(50),
ALCRESULT double,
DRUGTEST varchar(10),
DRUGRESULT varchar(50),
DRIVERCOND varchar(50),
VISIONOBS varchar(50),
DCONTCIRC1 varchar(50),
DCONTCIRC2 varchar(50),
VCONFIG varchar(50),
CARGOBODY varchar(50),
VYEAR int,
MAKE varchar(20),
MODEL varchar(20),
STYLE varchar(10),
VLP_STATE char(2),
OCCUPANTS int,
VACTION varchar(50),
SEQEVENTS1 varchar(50),
SEQEVENTS2 varchar(50),
SEQEVENTS3 varchar(50),
SEQEVENTS4 varchar(50),
MOSTHARM varchar(50),
SPEEDLIMIT varchar(10),
TRAFCONT varchar(50),
FIXOBJSTR varchar(50),
MOSTDAMAGE varchar(50),
DAMAGE varchar(50),
CSEVERITY varchar(50),
MAJORCAUSE varchar(50),
CSURFCOND varchar(20),
DRUGALCREL varchar(50),
ROADTYPE varchar(50),
WZ_RELATED varchar(50),
FATALITIES int,
CRASH_YEAR int,
XCOORD double,
YCOORD double,
FROM_MEASURE varchar(50),
TO_MEASURE varchar(50),
ROUTEID varchar(50),
CRASH_DATETIME varchar(25),
CRASH_DATETIME_UTC varchar(25),
CRASH_DATETIME_UTC_OFFSET varchar(50),
REST_UPDATED  varchar(25),
REST_UPDATED_UTC_OFFSET varchar(50)
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
STORED AS TEXTFILE
tblproperties ("skip.header.line.count"="1");

Load data local inpath "project/Moter_Vehicle_Crashes/crash_vehicle.csv" into table crash_vehicle;
```

crash_person Raw table

```
CREATE TABLE IF NOT EXISTS crash_person
(X double,
 Y double,
 OBJECTID int,
 CRASH_KEY bigint,
 CASENUMBER bigint ,
 PERSONKEY bigint,
 ZINJ_UNITKEY double,
 ZINJ_INJUREDAGE int,
 ZINJ_INJUREDGEN string,
 ZUNI_UNITKEY double,
 ZUNI_INJUREDAGE int,
 ZUNI_INJUREDGEN double,
 XCOORD double,
 YCOORD double,
 FATALITIES int,
 VEHICLES int,
 CRASH_YEAR int,
 ZINJ_INJSTATUS string,
 ZINJ_SEATING string,
 ZINJ_OCCPROTECT string,
 ZINJ_EJECTION string,
 ZINJ_EJECTIONPATH string,
 ZINJ_AIRBAGDEP string,
 ZINJ_TRAPPED string,
 ZUNI_INJSTATUS string,
 ZUNI_SEATING string,
 ZUNI_OCCPROTECT string,
 ZUNI_EJECTION string,
 ZUNI_EJECTIONPATH string,
 ZUNI_AIRBAGDEP string,
 ZUNI_TRAPPED string,
 CSEVERITY string,
 MAJORCAUSE string,
 CSURFCOND string,
 DRUGALCREL string,
 ROADTYPE string,
 WZ_RELATED string,
 NM_TYPE string,
 NM_LOC string,
 NM_ACTION string,
 NM_SAFETY string,
 NMCONTCIRC string,
 FROM_MEASURE double,
 TO_MEASURE double,
 ROUTEID varchar(50),
 CRASH_DATETIME varchar(25),
 CRASH_DATETIME_UTC varchar(25),
 CRASH_DATETIME_UTC_OFFSET varchar(50),
 REST_UPDATED varchar(25),
 REST_UPDATED_UTC_OFFSET varchar(50)
)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
STORED AS TEXTFILE
tblproperties ("skip.header.line.count"="1");
```

```
Load data local inpath "project/crash_person.csv" into table crash_person;
```

crash_person table

```
Create Table crash_person1
ROW FORMAT Delimited
STORED AS textfile
AS

Select X ,
 Y ,
 OBJECTID ,
 CRASH_KEY ,
 CASENUMBER  ,
 PERSONKEY ,
 ZINJ_UNITKEY ,
 ZINJ_INJUREDAGE ,
 ZINJ_INJUREDGEN ,
 ZUNI_UNITKEY ,
 ZUNI_INJUREDAGE ,
 ZUNI_INJUREDGEN ,
 XCOORD ,
 YCOORD ,
 FATALITIES ,
 VEHICLES ,
 CRASH_YEAR ,
 ZINJ_INJSTATUS ,
 ZINJ_SEATING ,
 ZINJ_OCCPROTECT ,
 ZINJ_EJECTION ,
 ZINJ_EJECTIONPATH ,
 ZINJ_AIRBAGDEP ,
 ZINJ_TRAPPED ,
 ZUNI_INJSTATUS ,
 ZUNI_SEATING ,
 ZUNI_OCCPROTECT ,
 ZUNI_EJECTION ,
 ZUNI_EJECTIONPATH ,
 ZUNI_AIRBAGDEP ,
 ZUNI_TRAPPED ,
 CSEVERITY ,
 MAJORCAUSE ,
 CSURFCOND ,
 DRUGALCREL ,
 ROADTYPE ,
 WZ_RELATED ,
 NM_TYPE ,
 NM_LOC ,
 NM_ACTION ,
 NM_SAFETY ,
 NMCONTCIRC ,
 FROM_MEASURE ,
 TO_MEASURE ,
 ROUTEID ,
 cast(from_unixtime(unix_timestamp(CRASH_DATETIME , "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'")) as
timestamp) as CRASH_DATETIME,
 cast(from_unixtime(unix_timestamp(CRASH_DATETIME_UTC , "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"))
as timestamp) as CRASH_DATETIME_UTC,
 CRASH_DATETIME_UTC_OFFSET ,
 cast(from_unixtime(unix_timestamp(REST_UPDATED , "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'")) as
timestamp) as REST_UPDATED,
REST_UPDATED_UTC_OFFSET
FROM crash_person;
```

Data observations

casenumber is the unique Id
crash_person has 897413 records with multiple rows for each case number
crash_location has 557186 records with unique case number
crash_vehicle has 960406 records with multiple rows for each case number

Frequency of injured and fatal population as it could be used for target variable
4 % of the people involved in crash had fatal injured status
12.4% Possible (complaint of pain/injury)
7.3% Suspected minor/non-incapacitating
1.8% Suspected serious/incapacitating
1.2% has unknown injure status
76.8% are blank/uninjured
In some of the data columns, data is shifting to left after loading the csv files to hive found some outliers

Queries

Show the number of crashes in each county, during each month using partition table:

```
SELECT COUNTY_NAME, crashmonth, count(*) as count FROM crash_location_p GROUP BY
COUNTY_NAME, crashmonth ORDER BY COUNTY_NAME, crashmonth;
```

Crashes based on weather:

```
SELECT WEATHER, count(*) as count FROM crash_location GROUP BY WEATHER ORDER BY
WEATHER;
```

weather	count
dirt"	7353
Blowing sand, 3182 Fog	3108
Mud, 445 Sleet	4120
7000	1
Alcohol (< Statutory)	16
Alcohol (Statutory)	254
Animal in roadway	11
Blowing Snow	1037
Clear	211518
Cloudy	102601
Dark - roadway lighted	14
Dark - roadway not lighted	7
Dark - unknown roadway lighting	1
Dawn	1
Daylight	26
Drug	14
Drug/Alcohol (Statutory)	5
Dry	68860

Dusk	1
Freezing rain/drizzle	2221
Glare	28
Gravel	180
Ice/frost	4560
Non-motorist action	16
None Indicated	1630
None apparent	8472
Not Reported	52069
Other (explain in narrative)	447
Rain	27944
Refused	102
Sand	12
Severe Winds	1137
Severe crosswind	1
Slush	1471
Snow	36631
Under Influence of Alcohol/Drugs/Medications	70
Unknown	5098
Visual obstruction	28
Water (standing or moving)	27
Weather conditions	159
Wet	12307

Crashes based on Drug or Alcohal:

```
SELECT DRUGALC, count(*) as count FROM crash_location GROUP BY DRUGALC ORDER BY
DRUGALC;
```

drugalc	count
NULL 1	
climate)"	558
erratic	8954
oncoming left turn"	1
opposite direction"	1
same direction"	5
Driver Distraction: Adjusting devices (radio, 44 Operating vehicle in an reckless	2091
0	1
Aggressive driving/road rage	117
Alcohol (< Statutory)	1382
Alcohol (Statutory)	12706
Animal	74
Cargo/equipment loss or shift	50
Crossed centerline (undivided)	8315
Crossed median (divided)	162
Disregarded RR Signal	8
Downhill runaway	29
Driver Distraction: Exterior distraction	243
Driver Distraction: Inattentive/lost in thought	274
Driver Distraction: Manual operation of an electronic communication device	297
Driver Distraction: Other electronic device activity	37

Driver Distraction: Other interior distraction	723
Driver Distraction: Passenger	191
Driver Distraction: Reaching for object(s)/fallen object(s)	202
Driver Distraction: Talking on a hand-held device	34
Driver Distraction: Talking on a hands free device	3
Driver Distraction: Unrestrained animal	18
Driving less than the posted speed limit	1
Driving too fast for conditions	3133
Drove around RR grade crossing gates	3
Drug	614
Drug/Alcohol (< Statutory)	42
Drug/Alcohol (Statutory)	168
Equipment failure	151
Exceeded authorized speed	367
FTYROW: At uncontrolled intersection	292
FTYROW: From driveway	1183
FTYROW: From parked position	1434
FTYROW: From stop sign	2823
FTYROW: From yield sign	448
FTYROW: Making left turn	18220
FTYROW: Making right turn on red signal	431
FTYROW: Other (explain in narrative)	5564
FTYROW: To pedestrian	10
Failed to keep in proper lane	702
Failed to yield to emergency vehicle	41
Failure to dim lights/have lights on	11
Failure to signal intentions	37
Followed too close	911
Illegally Parked/Unattended	431
Improper Backing	697
Improper or erratic lane changing	5412
Lost Control	1156
Made improper turn	8425
None Indicated	428137
Operator inexperience	68
Other (explain in narrative): Disregarded Warning Sign	9
Other (explain in narrative): Disregarded signs/road markings	24
Other (explain in narrative): Getting off/out of vehicle	11
Other (explain in narrative): Improper operation	7
Other (explain in narrative): No improper action	1230
Other (explain in narrative): Other	6998
Other (explain in narrative): Vision obstructed	745
Over correcting/over steering	438
Oversized Load/Vehicle	29
Passing: On wrong side	138
Passing: Other passing (explain in narrative)	574
Passing: Through/around barrier	64
Passing: Where prohibited by signs/markings	100
Passing: With insufficient distance/inadequate visibility	155

Ran Stop Sign	703
Ran Traffic Signal	2273
Ran off road - left	5157
Ran off road - right	50
Ran off road - straight	635
Refused	3676
Separation of units	157
Swerving/Evasive Action	2779
Towing Improperly	5
Traveling on prohibited traffic way	10
Traveling wrong way or on wrong side of road	566
Under Influence of Alcohol/Drugs/Medications	2975
Unknown	10244
Vehicle stopped on railroad tracks	1

Crashes based on DISTRICT:

```
SELECT DISTRICT, count(*) as count FROM crash_location GROUP BY DISTRICT ORDER BY DISTRICT;
```

district	count
NULL	61
1	157165
2	68026
3	62412
4	55264
5	63506
6	150752

Crashes based on Roadway conditions:

```
SELECT RCNTCRC, count(*) as count FROM crash_location GROUP BY RCNTCRC ORDER BY RCNTCRC;
```

rcntcrc	count
NULL	1
dirt"	445
hail"	4120
smoke	3108
soil	3182
Blowing sand, 587 Fog	623
Shoulders (none, 584 Sleet	732
Slippery, 1127 Surface condition (e.g.wet	51564
Traffic backup	971
Animal in roadway	1
Blowing Snow	131
Clear	57935
Cloudy	28788
Dark - roadway lighted	2730

Dark - roadway not lighted	1353
Dark - unknown roadway lighting	112
Dawn	111
Daylight	4365
Debris	1109
Disabled vehicle	78
Dry	43
Dusk	203
Freezing rain/drizzle	346
Glare	5
Ice/frost	1
Non-highway work	456
Non-motorist action	1
None apparent	305854
Not Reported	59955
Obstruction in roadway	748
Other (explain in narrative)	1766
Rain	6654
Ruts/holes/bumps	525
Severe Winds	184
Snow	6224
Traffic control obscured	213
Unknown	5172
Visual obstruction	2
Weather conditions	37
Wet	5
Work Zone (roadway-related)	5034

```
Select ZINJ_INJSTATUS, count(PERSONKEY) as count
From crash_person
Group by ZINJ_INJSTATUS;
```

Injured_status	count
Fatal	3703
Not reported	2
Possible (complaint of pain/injury)	111625
Suspected minor/non-incapacitating	65505
Suspected serious/incapacitating	16247
Unknown	11150
(blank)	689181

Injured status without protection

```
Select ZINJ_INJSTATUS, count(PERSONKEY) as count
From crash_person
where ZINJ_OCCPROTECT='None used'
Group by ZINJ_INJSTATUS;
```

Injured status	No protection used
Fatal	1659

Not reported	
Possible (complaint of pain/injury)	5617
Suspected minor/non-incapacitating	8223
Suspected serious/incapacitating	4431
Unknown	152

# of fatalities by gender

```
select zinj_injuredgen,sum(FATALITIES) as Fatalitiescount from crash_person group by
ZINJ_INJUREDGEN;
```

Gender	Fatalities count
Fe	836
Ma	1533
No	46
Un	1
(blank)	417

Number of injured vs uninjured

```
select ZUNI_INJSTATUS, count(personkey) as  count From crash_person  Group by ZUNI_INJSTATUS;
```

Injured Status	count
Uninjured	689181
blank/injured	208232

Injured status by year

```
Select crash_year, ZINJ_INJSTATUS, count(PERSONKEY) as count
From crash_person
Group by crash_year,  ZINJ_INJSTATUS;
```

Vehicles crashed based on DL State

```
select DL_STATE, count(*) as total from crash_vehicle group by DL_STATE order by total;
```

DL State (Only 8 of the highest displayed)	Total Vehicles
Texas	2411
South Dako	3511
Missouri	4714
Wisconsin	5673
Minnesota	7536
Nebraska	12557
Illinois	21987
Iowa	776543

Weekly analysis of Vehicles crashes

```
select from_unixtime(unix_timestamp(CRASH_DATETIME_UTC, "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"),
'E') as dow, count(*) as total from crash_vehicle group by
from_unixtime(unix_timestamp(CRASH_DATETIME_UTC, "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"), 'E')
order by dow;
```

Day	Total Vehicles
Sun	90323
Mon	129142

Tue	131203
Wed	129116
Thu	129685
Fri	146458
Sat	113134

Drinking and driving

```
select from_unixtime(unix_timestamp(CRASH_DATETIME_UTC, "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"),
'E') as dow, count(ALCRESULT) as total from crash_vehicle where ALCRESULT>0 group by
from_unixtime(unix_timestamp(CRASH_DATETIME_UTC, "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"), 'E')
order by total;
```

Day	Alcohol Influenced
Sun	32973
Mon	44860
Tue	45491
Wed	45337
Thu	45549
Fri	51221
Sat	40538

Time of Day Analysis

```
select hour(from_unixtime(unix_timestamp(CRASH_DATETIME_UTC, "yyyy-MM-
dd'T'HH:mm:ss.SSS'Z'"))) as hour, count(*) as total from crash_vehicle group by
hour(from_unixtime(unix_timestamp(CRASH_DATETIME_UTC, "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'")))
order by total;
```

Time (Highest 7 hours displayed)	Total vehicles
16	46554
19	53840
18	54363
17	54641
23	56977
20	66384
22	69151

Summary Statistics

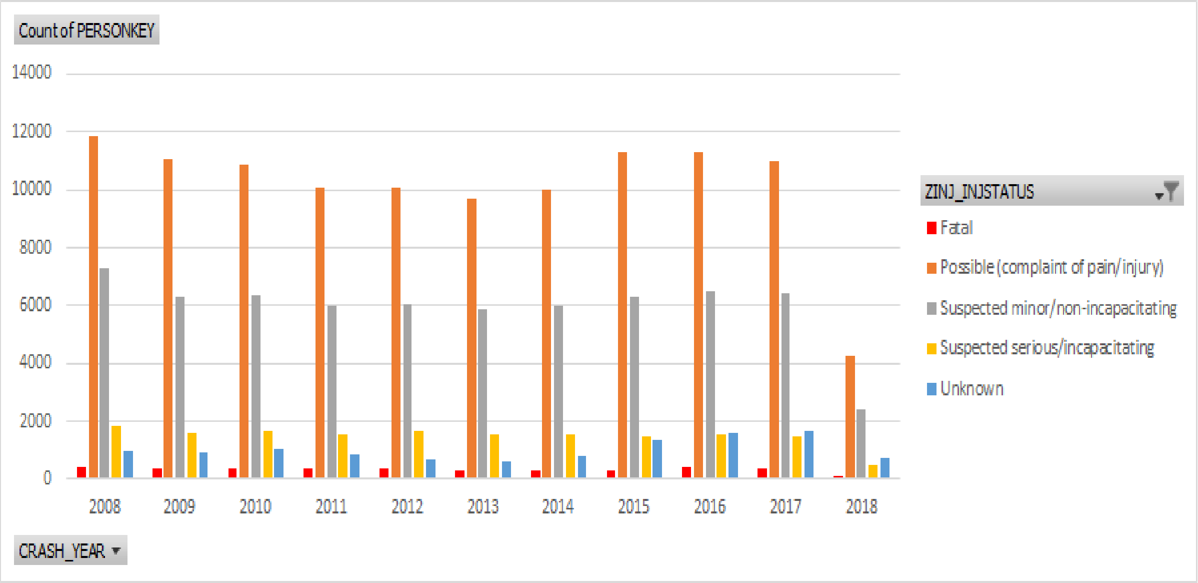
Analyze statistics crash_location:

```
hive -e "use pkuntla;analyze table crash_location compute statistics for columns FATALITIES,
INJURIES, MAJINJURY, MININJURY, POSSINJURY, UNKINJURY, PROPDMG, VEHICLES,
toccupants;describe formatted crash_location FATALITIES;describe formatted crash_location
INJURIES;describe formatted crash_location MAJINJURY;describe formatted crash_location
MININJURY;describe formatted crash_location POSSINJURY;describe formatted crash_location
UNKINJURY;describe formatted crash_location PROPDMG;describe formatted crash_location
VEHICLES;describe formatted crash_location toccupants;" > loc_stats.csv;
```

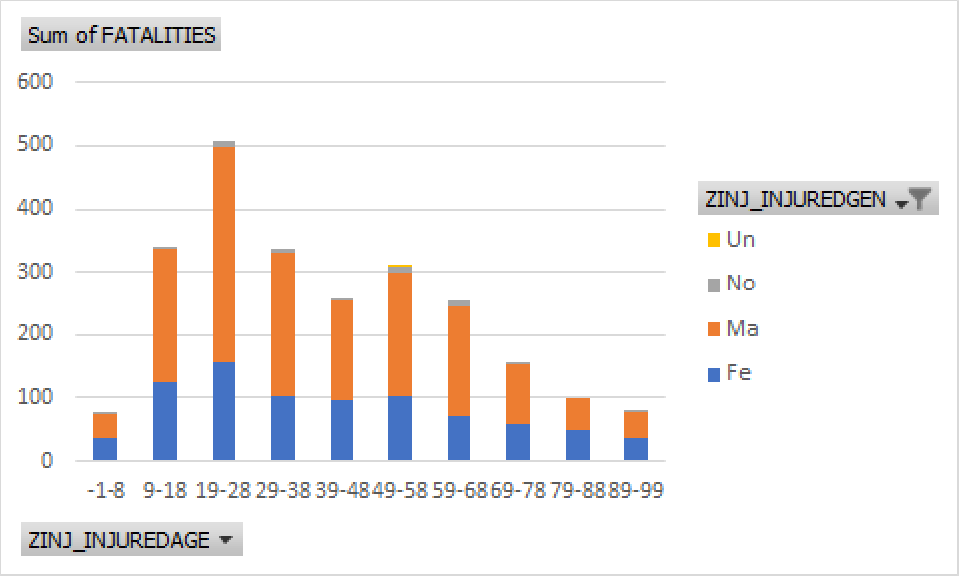
# col_name	data_type	min	max	num_nulls	distinct_count	avg_col_len	max_col_len	num_trues	num_falses	comment
FATALITIES	int	0	9	184474	7					from deserializer
INJURIES	int	0	38	35140	18					from deserializer
MAJINJURY	int	0	38	16450	13					from deserializer
MININJURY	int	0	38	3566	16					from deserializer
POSSINJURY	int	0	26	211	15					from deserializer
UNKINJURY	int	0	25	44	11					from deserializer
PROPDMG	double	0.0	4851387.0	6	6867					from deserializer
VEHICLES	int	0	4000000	3	3910					from deserializer
toccupants	int	0	1100000	2	1162					from deserializer

Plots

Bar Plot of Number of people injured by injured status for each year



AGE GROUP: FATALITIES by GENDER

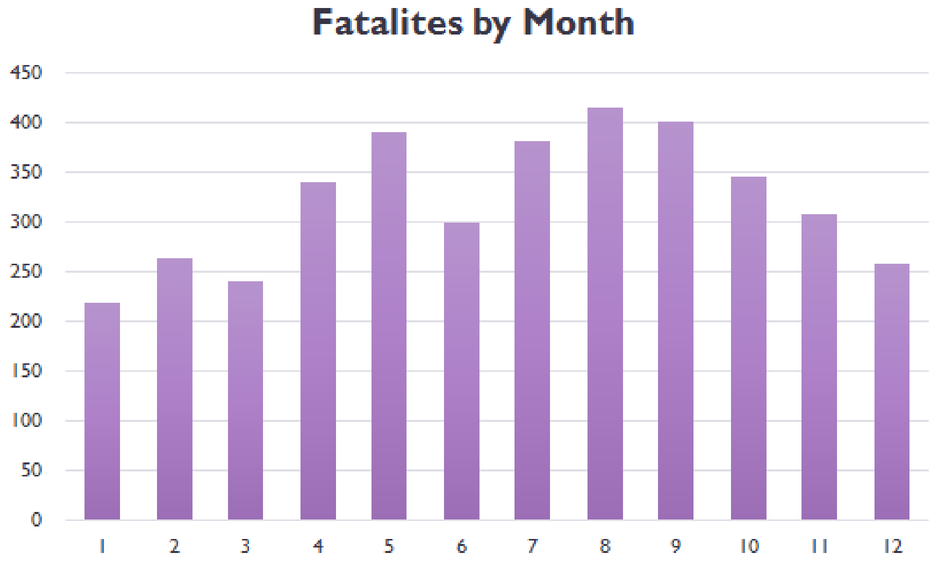


No Protection VS Injured Status

Injured status	People count	No protection used	Percentage
Fatal	3703	1659	44.80%
Not reported	2		0.00%
Possible (complaint of pain/injury)	111625	5617	5.03%
Suspected minor/non-incapacitating	65505	8223	12.55%
Suspected serious/incapacitating	16247	4431	27.27%
Unknown	11150	152	1.36%

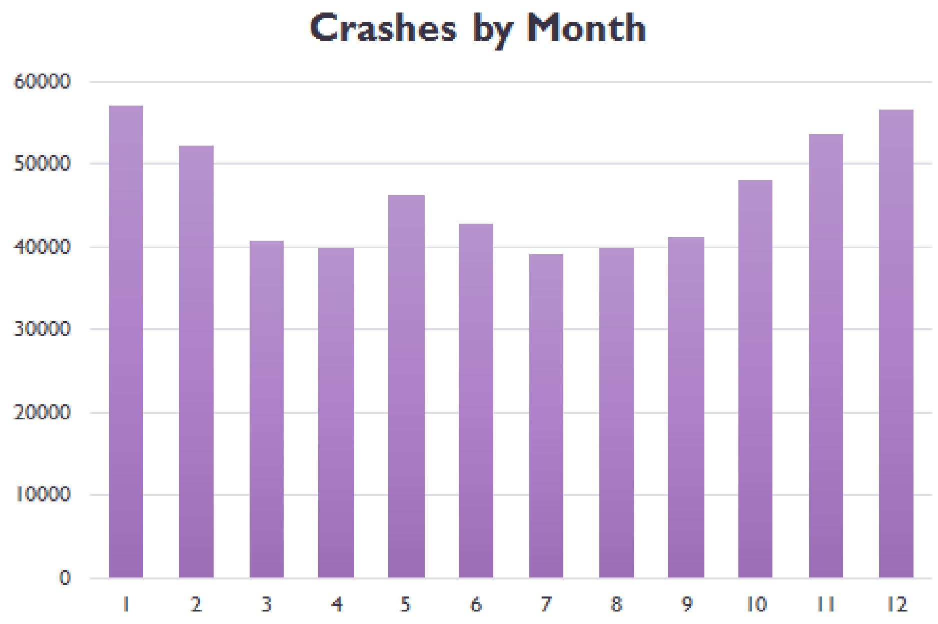
Show the number of FATALITIES by month:

```
SELECT month(crash_date) month, sum(FATALITIES) as count FROM crash_location GROUP BY
month(crash_date) ORDER BY month;;
```



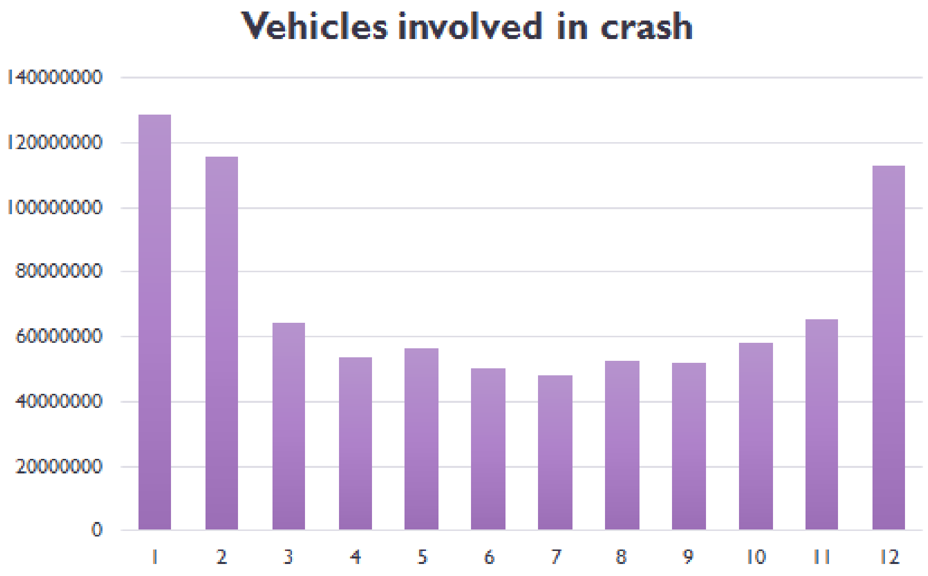
Show the number of crashes by month:

```
SELECT month(crash_date) month, count(*) as count FROM crash_location GROUP BY
month(crash_date) ORDER BY month;
```

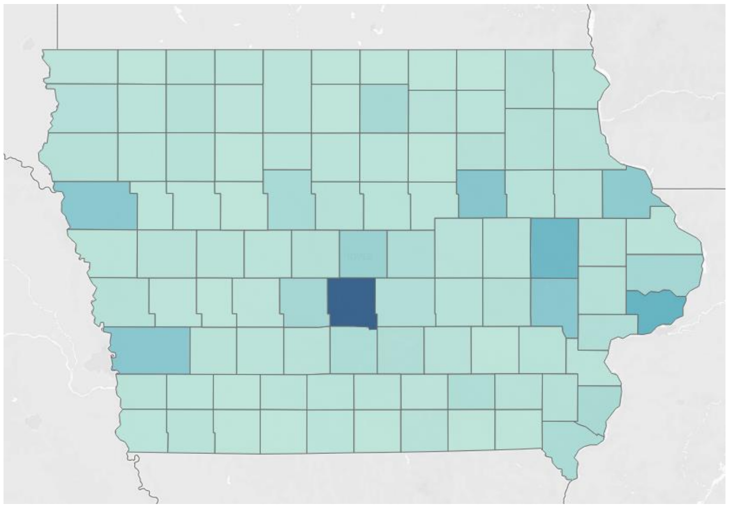


Show the number of VEHICLES involved in crash by month:

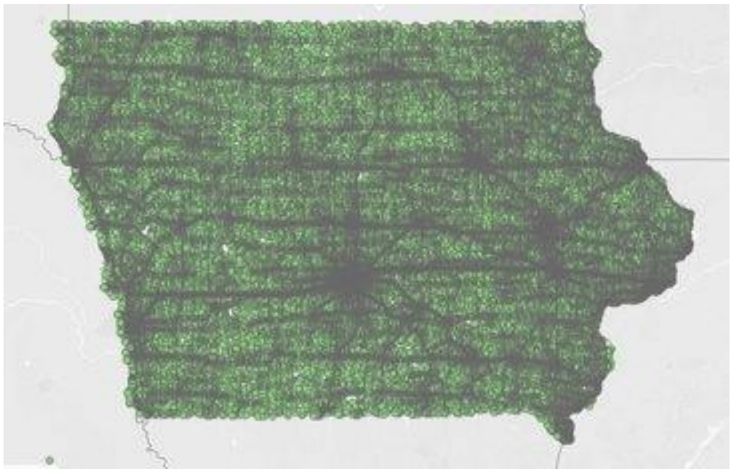
SELECT month(crash_date) month, sum(VEHICLES) as count FROM crash_location GROUP BY month(crash_date) ORDER BY month;



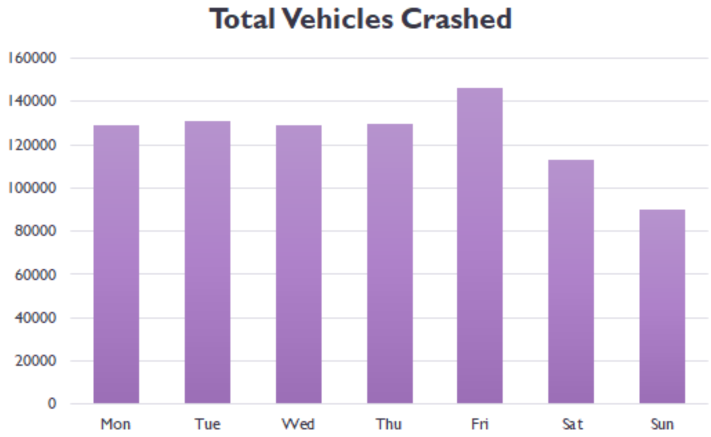
Map Visualization - County based density of the crashes



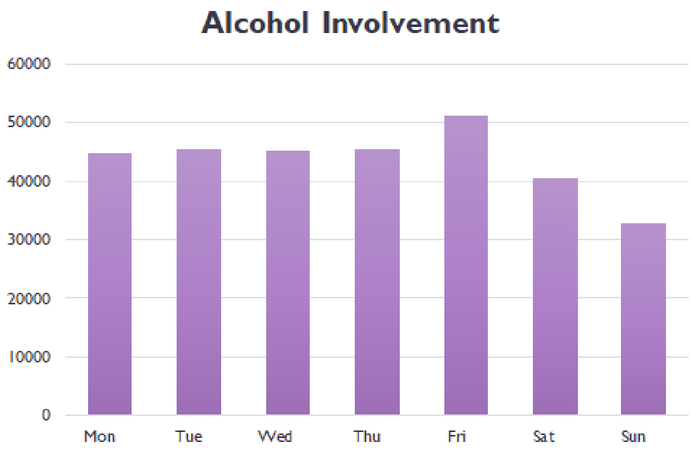
Map Visualization - Heatmap of the crashes



Weekly Analysis - Total vehicles crashed



Weekly Analysis - Alcohol Influence



Car models mostly involved

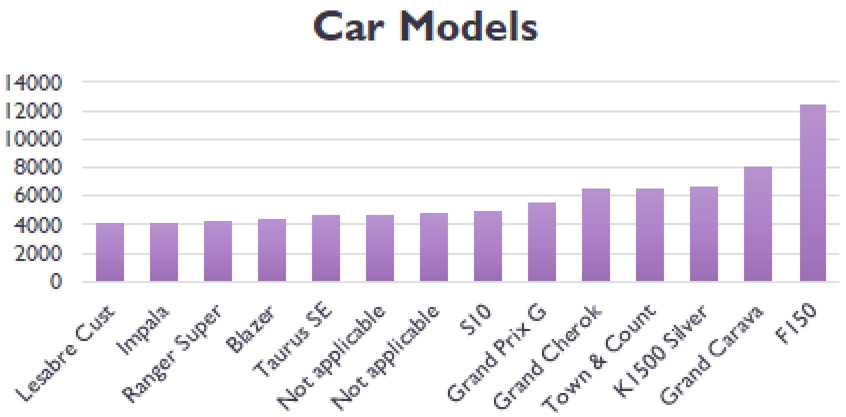


Table Partitioning/ Buckets

Since most of the analysis is based on months, partitioning the hive tables based on months will be good.

We tried partitioning with month to improve the performance.

Buckets would be good in our dataset; for example, age

Comparison of hive logs for partition and non-partition tables:

```
hive> select count(*), crashmonth from crash_location_p group by crashmonth;
Query ID = purnack_20191002224639_8e52d033-925c-4cf4-b8ad-eb09eb485b02
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 2
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
```

```
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1555710740360_3979, Tracking URL =
http://r383.opa.bridges.psc.edu:8088/proxy/application_1555710740360_3979/
Kill Command = /opt/packages/hadoop-testing/hadoop/hadoop/bin/hadoop job  -kill
job_1555710740360_3979
Hadoop job information for Stage-1: number of mappers: 3; number of reducers: 2
2019-10-02 22:46:44,942 Stage-1 map = 0%,  reduce = 0%
2019-10-02 22:46:50,259 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 9.54 sec
2019-10-02 22:46:54,419 Stage-1 map = 100%,  reduce = 50%, Cumulative CPU 11.53 sec
2019-10-02 22:46:55,463 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 13.68 sec
MapReduce Total cumulative CPU time: 13 seconds 680 msec
Ended Job = job_1555710740360_3979
MapReduce Jobs Launched:
Stage-Stage-1: Map: 3  Reduce: 2   Cumulative CPU: 13.68 sec   HDFS Read: 286677670 HDFS Write: 99
SUCCESS
Total MapReduce CPU Time Spent: 13 seconds 680 msec
OK
53533   11
52191   2
39872   4
42831   6
39801   8
57068   1
48028   10
56637   12
40612   3
46289   5
39065   7
41198   9
Time taken: 16.601 seconds, Fetched: 12 row(s)

hive> select count(*), month(crash_date) as crashmonth from crash_location group by
month(crash_date);
Query ID = purnack_20191002225052_ed611f33-dd8c-471b-af1f-0d87be6a4131
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 2
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1555710740360_3985, Tracking URL =
http://r383.opa.bridges.psc.edu:8088/proxy/application_1555710740360_3985/
Kill Command = /opt/packages/hadoop-testing/hadoop/hadoop/bin/hadoop job  -kill
job_1555710740360_3985
Hadoop job information for Stage-1: number of mappers: 3; number of reducers: 2
2019-10-02 22:50:57,014 Stage-1 map = 0%,  reduce = 0%
2019-10-02 22:51:02,231 Stage-1 map = 33%,  reduce = 0%, Cumulative CPU 4.05 sec
2019-10-02 22:51:03,267 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 17.05 sec
2019-10-02 22:51:08,431 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 21.21 sec
MapReduce Total cumulative CPU time: 21 seconds 210 msec
Ended Job = job_1555710740360_3985
```

```
MapReduce Jobs Launched:
Stage-Stage-1: Map: 3  Reduce: 2   Cumulative CPU: 21.21 sec   HDFS Read: 286691660 HDFS Write: 105
SUCCESS
Total MapReduce CPU Time Spent: 21 seconds 210 msec
OK
61      NULL
52191   2
39872   4
42831   6
39801   8
48028   10
56637   12
57068   1
40612   3
46289   5
39065   7
41198   9
53533   11
Time taken: 17.208 seconds, Fetched: 13 row(s)
```

Next Steps

- Data cleaning
- Impute missing values
- Correlation
- Feature selection
- Predictive Analysis based on the different features
  - Eg. Vehicle Type, Speed, Weather, Alcohol/ Drug, Road Condition…etc.
  - This would be beneficial for Insurance sector.
- Performance measure

Challenges

If we are predicting Fatality then we will have class imbalance challenge with the data as only 4% of the data has fatal injuries.