

Satya Ikyath Varma Dantuluri

NUID - 001051175

Project Proposal

I will use the dataset from US government's City of Chicago website. The dataset is called Crimes-2020. This dataset reflects reported incidents of crime (with the exception of murders where data exists for each victim) that occurred in the City of Chicago from 2001 to present, minus the most recent seven days. Data is extracted from the Chicago Police Department's CLEAR (Citizen Law Enforcement Analysis and Reporting) system. In order to protect the privacy of crime victims, addresses are shown at the block level only and specific locations are not identified and their field of study. Each row has 22 Columns such as location, block, arrest, district, reason etc.

The kinds of analysis I intend to perform are

1. Find number of cases per zip code
2. Find districts with highest number of cases.
3. Find different cases based on gender like homicide etc.,
4. Analysis of crime based on location.
5. Crime Trends over the years.
6. Analysis on different crime types which will help to take better actions on future.

The below analysis covers all kinds of Map Reduce Analysis covered in classes till date.

Analysis of Crime in Chicago:

OBJECTIVE:

The Objective is to highlight potential for improving Chicago policing through Big Data Analytics using Data Analysis and visualization related to crime in Chicago. Crime is an important and popular subject. But interpreting crime data is tricky business and developing coherent narratives and useful metrics is even harder. Our analysis can be used for daily operations in police department and can make significant progress.

“Big Data” does not just refer to the amount of data that is generated recently. It is a new way of approaching the way we analyze the world and needs to sit on a platform that allows for near-constant exponential growth of data.

The reason for considering Chicago Crime is high man slaughter rate which is 15.2 per 100,000 residents in Chicago while it is 4.0 for New York and 6.5 for Los Angeles according to the FBI crime statistics. Also, there is no decline in the past decade as compared to the other two large cities, which have been on a slow declining slope.

About the Dataset:

<https://data.cityofchicago.org/Public-Safety/Crimes-2020/qzdf-xmn8>

I will use the dataset from US government’s City of Chicago website. The dataset is called Crimes-2020. This dataset reflects reported incidents of crime (except for murders where data exists for each victim) that occurred in the City of Chicago from 2001 to present, minus the most recent seven days. Data is extracted from the Chicago Police Department's CLEAR (Citizen Law Enforcement Analysis and Reporting) system. In order to protect the privacy of crime victims, addresses are shown at the block level only and specific locations are not identified and their field of study. Each row has 22 Columns such as location, block, arrest, district, reason etc.

The kinds of analysis I intend to perform are

Find number of cases per zip code

Find districts with highest number of cases.

Find different cases based on gender like homicide etc.,

Analysis of crime based on location.

Crime Trends over the years.

Analysis on different crime types which will help to take better actions on future.

RESEARCH/FINDINGS

- Hadoop MapReduce, Hive, Mahout can be efficiently used on such Crime Dataset to find out the statistics of the crime based on different fields.
- Also, Machine Learning Algorithms like Logistic Regression can be used for making predictions like the Area where Crime can take place, Type of crime or Victim of a Particular race in a particular district.

Hadoop Analysis

Map Reduce Analysis:

Crime Count Year wise:

```
bytes written:220
Ikyaths-MacBook-Pro:sbin ikyathvarmadantuluri$ hadoop fs -head /chicagoCrimeOutput/CrimeCountPerYear/part-r-00000
2020-12-13 19:43:29,117 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2012      336149
2015      264462
2018      268254
Ikyaths-MacBook-Pro:sbin ikyathvarmadantuluri$ hadoop fs -head /chicagoCrimeOutput/CrimeCountPerYear/part-r-00001
2020-12-13 19:44:58,546 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2013      307310
2016      269477
2019      268444
Ikyaths-MacBook-Pro:sbin ikyathvarmadantuluri$ hadoop fs -head /chicagoCrimeOutput/CrimeCountPerYear/part-r-00002
2020-12-13 19:45:02,764 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2011      351881
2014      275567
2017      268694
2020      191744
Ikyaths-MacBook-Pro:sbin ikyathvarmadantuluri$ █
```

Conclusion: This analysis for the crimes based on crime category and subcategory on a yearly basis. This will be useful to Police Department to analyze what type of crimes are more occurring in each year and they can compare the occurrences of a particular crime in different years and they can take appropriate measures to reduce.

Crime Count by Time of the Day:

```
2020-12-13 21:19:47,851 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
 253451
Early Afternoon 429646
Early Evening 615782
Early Morning 251299
Late Afternoon 146871
Late Morning 293303
Night 803630
Ikyaths-MacBook-Pro:sbin ikyathvarmadantuluri$
```

Conclusion: This analysis for the crimes based on crime category and subcategory on a yearly daily basis. This will be useful to Police Department to analyze what type of crimes are more occurring in each time of the day and they can compare the occurrences of a particular crime in different parts of the day and they can take appropriate actions.

Crime Count by District wise:

```
2020-12-13 21:31:04,091 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
001 127639
002 118906
003 138875
004 161023
005 125387
006 170019
007 158111
008 181914
009 131279
010 127942
011 194012
012 134892
014 100435
015 118472
016 93956
017 79484
018 132455
019 126016
020 47826
022 90532
024 80581
025 154944
031 81
Ikyaths-MacBook-Pro:sbin ikyathvarmadantuluri$
```

Conclusion: This analysis for the crimes based on district. This will be useful to Police Department to analyze which districts need to be more protected accordingly.

Crime Type Count by District wise:

```
Ikyaths-MacBook-Pro:sbin ikyathvarmandanturi$ hadoop fs -head /chicagoCrimeOutput/CrimeTypeCountPerDistrict/part-r-00000
2020-12-13 21:38:56,650 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
ARSON-002      83
ARSON-005      267
ARSON-008      394
ARSON-011      331
ARSON-014      158
ARSON-017      137
ARSON-020      34
ASSAULT-003    11076
ASSAULT-006    13559
ASSAULT-009    9454
ASSAULT-012    8289
ASSAULT-015    8494
ASSAULT-018    4824
ASSAULT-024    5180
BATTERY-002    23145
BATTERY-005    27128
BATTERY-008    38760
BATTERY-011    37620
BATTERY-014    14279
BATTERY-017    12886
BATTERY-020    8123
BURGLARY-001    2100
BURGLARY-004    11088
BURGLARY-007    8647
BURGLARY-010    5331
BURGLARY-016    6188
BURGLARY-019    7844
BURGLARY-022    5611
BURGLARY-025    9540
BURGLARY-031    1
CONCEALED CARRY LICENSE VIOLATION-002    7
CONCEALED CARRY LICENSE VIOLATION-005    33
CONCEALED CARRY LICENSE VIOLATION-008    98
CONCEALED CARRY LICENSE VIOLATION-011    39
CONCEALED CARRY LICENSE VIOLATION-014    5
CONCEALED CARRY LICENSE VIOLATION-017    4
CONCEALED CARRY LICENSE VIOLATION-020    2
CRIM SEXUAL ASSAULT-002 559
CRIM SEXUAL ASSAULT-005 580
CRIM SEXUAL ASSAULT-008 801
CRIM SEXUAL ASSAULT-011 769
CRIM SEXUAL ASSAULT-014 399
CRIM SEXUAL ASSAULT-017 355
CRIM SEXUAL ASSAULT-020 297
CRIMINAL DAMAGE-003 16023
```

Conclusion: This analysis for the crime types based on district. This will be useful to Police Department to analyze which districts need to be more protected accordingly and act based on crime types like have more patrolling or other ways.

Top n Days of Crime and Moving Average of Crime count:

In this I tried to implement chain of map reduce jobs so to find Top n Days of Crime and Moving Average of Crime count

Initially I have written one map reduce job to find Crime count day wise which is in turn used to calculate Moving Average and Top n Days of Crime.

About Moving average – In Time Series Forecasting

The moving average model is probably the naivest approach to time series modelling. This model simply states that the next observation is the mean of all past observations.

Although simple, this model might be surprisingly good, and it represents a good starting point.

Otherwise, the moving average can be used to identify interesting trends in the data. We can define a window to apply the moving average model to smooth the time series and highlight different trends.

Below Screenshots for all the MapReduce jobs output of the scenario:

Crime Count Day wise:

```

2020-12-13 22:21:04,764 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
11/01/01      1548
11/01/02      712
11/01/03      885
11/01/04      845
11/01/05      896
11/01/06      889
11/01/07      937
11/01/08      837
11/01/09      749
11/01/10      918
11/01/11      846
11/01/12      910
11/01/13      937
11/01/14      987
11/01/15      895
11/01/16      769
11/01/17      889
11/01/18      884
11/01/19      933
11/01/20      818
11/01/21      840
11/01/22      756
11/01/23      811
11/01/24      844
11/01/25      914
11/01/26      919
11/01/27      842
11/01/28      982
11/01/29      826
11/01/30      756
11/01/31      794
11/02/01      700
11/02/02      320
11/02/03      524
11/02/04      669
11/02/05      705
11/02/06      653
11/02/07      763
11/02/08      718
11/02/09      762
11/02/10      780

```

Moving Average of Crime Count:

```

lizlyyathis-macbook-Pro:~/Desktop/lkydcrimedataanalysis$ hadoop fs -head /chicago/crimeoutput/crimeCountPerDay/part-r-00000
2020-12-13 22:21:04,764 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
11/01/01      1548
11/01/02      712
11/01/03      885
11/01/04      845
11/01/05      896
11/01/06      889
11/01/07      937
11/01/08      837
11/01/09      749
11/01/10      918
11/01/11      846
11/01/12      910
11/01/13      937
11/01/14      987
11/01/15      895
11/01/16      769
11/01/17      889
11/01/18      884
11/01/19      933
11/01/20      818
11/01/21      840
11/01/22      756
11/01/23      811
11/01/24      844
11/01/25      914
11/01/26      919
11/01/27      842
11/01/28      982
11/01/29      826
11/01/30      756
11/01/31      794
11/02/01      700
11/02/02      320
11/02/03      524
11/02/04      669
11/02/05      705
11/02/06      653
11/02/07      763
11/02/08      718
11/02/09      762
11/02/10      780

```

Top 100 Crime Days in Chicago:

```
2020-12-13 22:23:07,945 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
16/09/01    1031
12/08/24    1032
11/06/30    1033
11/08/29    1034
14/06/01    1035
11/10/31    1036
13/05/15    1037
11/08/13    1038
11/07/26    1039
12/07/27    1040
12/03/01    1042
15/08/01    1043
12/07/28    1044
12/07/04    1045
11/10/28    1046
12/06/27    1047
12/06/09    1049
11/08/27    1050
11/07/18    1051
11/12/01    1053
12/10/01    1054
11/10/07    1055
12/07/02    1056
11/05/09    1057
12/12/01    1058
12/07/03    1060
12/08/03    1061
12/07/07    1062
12/08/15    1063
12/05/01    1064
12/07/21    1065
12/06/11    1066
11/06/21    1067
12/06/18    1068
12/03/17    1069
11/06/06    1071
12/07/20    1072
11/10/01    1073
12/11/01    1074
11/07/30    1075
12/06/13    1076
12/06/24    1077
13/07/01    1078
14/08/01    1079
12/08/25    1080
13/06/01    1081
12/05/25    1083
12/07/13    1084
11/07/06    1087
12/06/12    1089
11/09/02    1090
```

Crime CaseID Sorting District wise:

In this I tried to implement Secondary Sorting in map reduce jobs so to sort crime Case ID in descending order by District wise.

Below Screenshots for all the MapReduce jobs output of the scenario:

```
Ikyaths-MacBook-Pro:sbin ikyathvarmadanturi$ hadoop fs -head /chicagoCrimeOutput/CrimeCountByDistrictByType/part-r-00000
2020-12-14 12:50:03,955 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
,JA549530
001,XX397722
001,XX313036
001,XX104853
001,TT654554
001,JS215740
001,JD566318
001,JD451059
001,JD450642
001,JD450061
001,JD449771
001,JD449750
001,JD449710
001,JD449577
001,JD449241
001,JD448667
001,JD448534
001,JD447989
001,JD447787
001,JD447627
001,JD447421
001,JD447269
001,JD447196
001,JD446940
001,JD446824
001,JD446752
001,JD446556
001,JD446514
001,JD446510
001,JD446452
001,JD446437
001,JD446295
001,JD446200
001,JD446115
001,JD446049
001,JD445704
001,JD445565
001,JD445467
001,JD445355
001,JD445219
001,JD445142
001,JD445022
001,JD444819
001,JD444779
001,JD444760
001,JD444716
001,JD444708
001,JD444680
001,JD444669
001,JD444649
001,JD444590
001,JD444463
001,JD444334
001,JD444332
001,JD444323
```

Hive Analysis:

Table - Crime

Crime Table Schema Initialization:

```
create external table crimetsv(
    ID_crime string,
    Case_Number string,
    Date_crime string,
    Block string,
    IUCR string,
    Primary_Type string,
    Description string,
    Location_Description string,
    Arrest string,
```

```

Domestic string,
Beat string,
District string,
Ward string,
Community_Area string,
FBI_Code string,
X_Coordinate string,
Y_Coordinate string,
Year_crime string,
Updated_On string,
Latitude string,
Longitude string
) row format delimited fields terminated by '\t' location '/chicagoCrimeHive';

```

Table – Arrest

Arrest Table Schema Initialization:

```

create external table arrest(
CB_NO string,
Case_Number string,
Arrest_Date string,
Race string,
Charge1Statute string,
Charge1StatuteDescription string,
Charge1Type string,
Charge1Class string,
Charge2Statute string,
Charge2StatuteDescription string,
Charge2Type string,
Charge2Class string,
Charge3Statute string,
Charge3StatuteDescription string,
Charge3Type string,
Charge3Class string,
Charge4Statute string,
Charge4StatuteDescription string,
Charge4Type string,

```

```
Charge4Class string,  
ChargesStatute string,  
ChargesDescription string,  
ChargesType string,  
ChargesClass string  
) row format delimited fields terminated by '\t' location '/chicagoCrimeArrests';
```

Query 1:

Crime Count by Primary Type:

```
create table query1 as  
select Primary_Type,count(*) as CrimeCount  
from crimetsv  
group by Primary_Type order by Primary_Type;
```

```
hive> select * from query1;
OK
ARSON      4438
ASSAULT    188190
BATTERY    509382
BURGLARY   151913
CONCEALED CARRY LICENSE VIOLATION      659
CRIM SEXUAL ASSAULT      12505
CRIMINAL DAMAGE 298151
CRIMINAL SEXUAL ASSAULT 2104
CRIMINAL TRESPASS       69685
DECEPTIVE PRACTICE     162999
GAMBLING     3507
HOMICIDE     5605
HUMAN TRAFFICKING     68
INTERFERENCE WITH PUBLIC OFFICER      11750
INTIMIDATION    1473
KIDNAPPING     2004
LIQUOR LAW VIOLATION     3400
MOTOR VEHICLE THEFT      119025
NARCOTICS     221517
NON - CRIMINAL    38
NON-CRIMINAL    170
NON-CRIMINAL (SUBJECT SPECIFIED)      9
OBSCENITY     505
OFFENSE INVOLVING CHILDREN      22841
OTHER NARCOTIC VIOLATION      61
OTHER OFFENSE    170164
PROSTITUTION    12433
PUBLIC INDECENCY     118
PUBLIC PEACE VIOLATION     21779
Primary Type    1
RITUALISM      1
ROBBERY      107511
SEX OFFENSE    10565
STALKING      1812
THEFT        632527
WEAPONS VIOLATION     45073
Time taken: 0.128 seconds, Fetched: 36 row(s)
```

Query 2:

Crime Count by Primary Type and Description:

```
create table query2 as
select Primary_Type,description,count(*) as CrimeCount
from crimetsv
group by Primary_Type,description order by Primary_Type,description;
```

ARSON AGGRAVATED 597
ARSON ATTEMPT ARSON 617
ARSON BY EXPLOSIVE 32
ARSON BY FIRE 3155
ARSON POS: CHEMICAL/DRY-ICE DEVICE 3
ARSON POS: EXPLOSIVE/INCENDIARY DEV 30
ARSON POSSESSION – EXPLOSIVE / INCENDIARY DEVICE 4
ASSAULT AGG PO HANDS NO/MIN INJURY 3443
ASSAULT AGG PRO.EMP: HANDGUN 230
ASSAULT AGG PRO.EMP: OTHER DANG WEAPON 432
ASSAULT AGG PRO.EMP: OTHER FIREARM 28
ASSAULT AGG PRO.EMP:KNIFE/CUTTING INST 323
ASSAULT AGGRAVATED – HANDGUN 3255
ASSAULT AGGRAVATED – KNIFE / CUTTING INSTRUMENT 1033
ASSAULT AGGRAVATED – OTHER DANGEROUS WEAPON 725
ASSAULT AGGRAVATED – OTHER FIREARM 93
ASSAULT AGGRAVATED PO: HANDGUN 458
ASSAULT AGGRAVATED PO: OTHER DANG WEAP 571
ASSAULT AGGRAVATED PO: OTHER FIREARM 25
ASSAULT AGGRAVATED PO:KNIFE/CUT INSTR 302
ASSAULT AGGRAVATED POLICE OFFICER – HANDGUN 61
ASSAULT AGGRAVATED POLICE OFFICER – HANDS, FISTS, FEET, NO INJURY 110
ASSAULT AGGRAVATED POLICE OFFICER – KNIFE / CUTTING INSTRUMENT 20
ASSAULT AGGRAVATED POLICE OFFICER – OTHER DANGEROUS WEAPON 51
ASSAULT AGGRAVATED POLICE OFFICER – OTHER FIREARM 1
ASSAULT AGGRAVATED PROTECTED EMPLOYEE – HANDGUN 26
ASSAULT AGGRAVATED PROTECTED EMPLOYEE – KNIFE / CUTTING INSTRUMENT 29
ASSAULT AGGRAVATED PROTECTED EMPLOYEE – OTHER DANGEROUS WEAPON 39
ASSAULT AGGRAVATED PROTECTED EMPLOYEE – OTHER FIREARM 1
ASSAULT AGGRAVATED: HANDGUN 21280
ASSAULT AGGRAVATED: OTHER DANG WEAPON 9541
ASSAULT AGGRAVATED: OTHER FIREARM 859

ASSAULT AGGRAVATED:KNIFE/CUTTING INSTR 12862
ASSAULT PRO EMP HANDS NO/MIN INJURY 6102
ASSAULT PROTECTED EMPLOYEE - HANDS, FISTS, FEET, NO / MINOR INJURY 166
ASSAULT SIMPLE 126124
BATTERY AGG PO HANDS ETC SERIOUS INJ 175
BATTERY AGG PO HANDS NO/MIN INJURY 6513
BATTERY AGG PRO EMP HANDS SERIOUS INJ 223
BATTERY AGG PRO.EMP: HANDGUN 1
BATTERY AGG PRO.EMP: OTHER DANG WEAPON 529
BATTERY AGG PRO.EMP: OTHER FIREARM 3
BATTERY AGG PRO.EMP:KNIFE/CUTTING INST 81
BATTERY AGG. DOMESTIC BATTERY - HANDS, FISTS, FEET, SERIOUS INJURY 511
BATTERY AGG. PROTECTED EMPLOYEE - HANDS, FISTS, FEET, SERIOUS INJURY 27
BATTERY AGG: HANDS/FIST/FEET NO/MINOR INJURY 1532
BATTERY AGG: HANDS/FIST/FEET SERIOUS INJURY 1472
BATTERY AGGRAVATED - HANDGUN 3250
BATTERY AGGRAVATED - HANDS, FISTS, FEET, NO / MINOR INJURY 114
BATTERY AGGRAVATED - HANDS, FISTS, FEET, SERIOUS INJURY 102
BATTERY AGGRAVATED - KNIFE / CUTTING INSTRUMENT 698
BATTERY AGGRAVATED - OTHER DANGEROUS WEAPON 1577
BATTERY AGGRAVATED - OTHER FIREARM 13
BATTERY AGGRAVATED DOMESTIC BATTERY - HANDGUN 28
BATTERY AGGRAVATED DOMESTIC BATTERY - KNIFE / CUTTING INSTRUMENT 546
BATTERY AGGRAVATED DOMESTIC BATTERY - OTHER DANGEROUS WEAPON 900
BATTERY AGGRAVATED DOMESTIC BATTERY - OTHER FIREARM 1
BATTERY AGGRAVATED DOMESTIC BATTERY: HANDGUN 92
BATTERY AGGRAVATED DOMESTIC BATTERY: HANDS/FIST/FEET SERIOUS INJURY 3013
BATTERY AGGRAVATED DOMESTIC BATTERY: KNIFE / CUTTING INSTRUMENT 2
BATTERY AGGRAVATED DOMESTIC BATTERY: KNIFE/CUTTING INST 6843
BATTERY AGGRAVATED DOMESTIC BATTERY: OTHER DANG WEAPON 9785
BATTERY AGGRAVATED DOMESTIC BATTERY: OTHER DANGEROUS WEAPON 3
BATTERY AGGRAVATED DOMESTIC BATTERY: OTHER FIREARM 3
BATTERY AGGRAVATED OF A CHILD 518
BATTERY AGGRAVATED OF A SENIOR CITIZEN 1832
BATTERY AGGRAVATED OF A UNBORN CHILD 25
BATTERY AGGRAVATED OF AN UNBORN CHILD 9
BATTERY AGGRAVATED P.O. - HANDS, FISTS, FEET, NO / MINOR INJURY 484
BATTERY AGGRAVATED P.O. - HANDS, FISTS, FEET, SERIOUS INJURY 23
BATTERY AGGRAVATED PO: HANDGUN 24

BATTERY AGGRAVATED PO: KNIFE/CUT INSTR 84
BATTERY AGGRAVATED PO: OTHER DANG WEAP 532
BATTERY AGGRAVATED PO: OTHER DANGEROUS WEAPON 1
BATTERY AGGRAVATED PO: OTHER FIREARM 1
BATTERY AGGRAVATED POLICE OFFICER - HANDGUN 9
BATTERY AGGRAVATED POLICE OFFICER - KNIFE / CUTTING INSTRUMENT 7
BATTERY AGGRAVATED POLICE OFFICER - OTHER DANGEROUS WEAPON 107
BATTERY AGGRAVATED PROTECTED EMPLOYEE - HANDGUN 2
BATTERY AGGRAVATED PROTECTED EMPLOYEE - KNIFE / CUTTING INSTRUMENT 6
BATTERY AGGRAVATED PROTECTED EMPLOYEE - OTHER DANGEROUS WEAPON 52
BATTERY AGGRAVATED: HANDGUN 15517
BATTERY AGGRAVATED: HANDS / FIST / FEET NO / MINOR INJURY 1
BATTERY AGGRAVATED: HANDS / FIST / FEET SERIOUS INJURY 1
BATTERY AGGRAVATED: KNIFE / CUTTING INSTRUMENT 1
BATTERY AGGRAVATED: OTHER DANG WEAPON 18508
BATTERY AGGRAVATED: OTHER DANGEROUS WEAPON 3
BATTERY AGGRAVATED: OTHER FIREARM 129
BATTERY AGGRAVATED:KNIFE/CUTTING INSTR 8661
BATTERY DOMESTIC BATTERY SIMPLE 250163
BATTERY OF AN UNBORN CHILD 3
BATTERY OF UNBORN CHILD 58
BATTERY PRO EMP HANDS NO/MIN INJURY 6965
BATTERY PROTECTED EMPLOYEE - HANDS, FISTS, FEET, NO / MINOR INJURY 237
BATTERY PROTECTED EMPLOYEE: HANDS NO / MIN INJURY 1
BATTERY SIMPLE 167381
BURGLARY ATTEMPT FORCIBLE ENTRY 7147
BURGLARY FORCIBLE ENTRY 95715
BURGLARY HOME INVASION 2588
BURGLARY UNLAWFUL ENTRY 46463
CONCEALED CARRY LICENSE VIOLATION ARMED WHILE UNDER THE INFLUENCE 205
CONCEALED CARRY LICENSE VIOLATION OTHER 228
CONCEALED CARRY LICENSE VIOLATION PROHIBITED PLACES 226
CRIM SEXUAL ASSAULT AGGRAVATED: HANDGUN 580
CRIM SEXUAL ASSAULT AGGRAVATED: KNIFE/CUT INSTR 339
CRIM SEXUAL ASSAULT AGGRAVATED: OTHER 1709
CRIM SEXUAL ASSAULT AGGRAVATED: OTHER DANG WEAPON 105
CRIM SEXUAL ASSAULT AGGRAVATED: OTHER FIREARM 8
CRIM SEXUAL ASSAULT ATTEMPT AGG: HANDGUN 49
CRIM SEXUAL ASSAULT ATTEMPT AGG: KNIFE/CUT INSTR 48

CRIM SEXUAL ASSAULT ATTEMPT AGG: OTHER 106
CRIM SEXUAL ASSAULT ATTEMPT AGG: OTHER DANG WEAPON 10
CRIM SEXUAL ASSAULT ATTEMPT AGG: OTHER FIREARM 1
CRIM SEXUAL ASSAULT ATTEMPT NON-AGGRAVATED 634
CRIM SEXUAL ASSAULT NON-AGGRAVATED 7743
CRIM SEXUAL ASSAULT PREDATORY 1173
CRIMINAL DAMAGE CRIMINAL DEFACEMENT 12847
CRIMINAL DAMAGE INSTITUTIONAL VANDALISM 181
CRIMINAL DAMAGE LIBRARY VANDALISM 20
CRIMINAL DAMAGE TO CITY OF CHICAGO PROPERTY 6097
CRIMINAL DAMAGE TO FIRE FIGHT.APP.EQUIP 88
CRIMINAL DAMAGE TO PROPERTY 139599
CRIMINAL DAMAGE TO STATE SUP PROP 1446
CRIMINAL DAMAGE TO STATE SUPPORTED PROPERTY 74
CRIMINAL DAMAGE TO VEHICLE 137799
CRIMINAL SEXUAL ASSAULT AGGRAVATED - HANDGUN 87
CRIMINAL SEXUAL ASSAULT AGGRAVATED - KNIFE / CUTTING INSTRUMENT 44
CRIMINAL SEXUAL ASSAULT AGGRAVATED - OTHER 304
CRIMINAL SEXUAL ASSAULT AGGRAVATED - OTHER DANGEROUS WEAPON 14
CRIMINAL SEXUAL ASSAULT AGGRAVATED - OTHER FIREARM 1
CRIMINAL SEXUAL ASSAULT AGGRAVATED: KNIFE / CUTTING INSTRUMENT 1
CRIMINAL SEXUAL ASSAULT ATTEMPT AGGRAVATED - HANDGUN 5
CRIMINAL SEXUAL ASSAULT ATTEMPT AGGRAVATED - KNIFE / CUTTING INSTRUMENT 4
CRIMINAL SEXUAL ASSAULT ATTEMPT AGGRAVATED - OTHER 7
CRIMINAL SEXUAL ASSAULT ATTEMPT AGGRAVATED - OTHER DANGEROUS WEAPON 1
CRIMINAL SEXUAL ASSAULT ATTEMPT NON-AGGRAVATED 55
CRIMINAL SEXUAL ASSAULT NON-AGGRAVATED 1383
CRIMINAL SEXUAL ASSAULT PREDATORY 198
CRIMINAL TRESPASS TO AIRPORT 140
CRIMINAL TRESPASS TO LAND 42710
CRIMINAL TRESPASS TO RESIDENCE 12837
CRIMINAL TRESPASS TO STATE SUP LAND 6983
CRIMINAL TRESPASS TO VEHICLE 7015
DECEPTIVE PRACTICE AGGRAVATED COMPUTER TAMPERING 19
DECEPTIVE PRACTICE AGGRAVATED FINANCIAL IDENTITY THEFT 821
DECEPTIVE PRACTICE ALTER COINS 4
DECEPTIVE PRACTICE ATTEMPT - FINANCIAL IDENTITY THEFT 4003
DECEPTIVE PRACTICE BOGUS CHECK 4559
DECEPTIVE PRACTICE COMPUTER FRAUD 1125

DECEPTIVE PRACTICE COUNTERFEIT CHECK 4498
DECEPTIVE PRACTICE COUNTERFEITING DOCUMENT 5916
DECEPTIVE PRACTICE CREDIT CARD FRAUD 33025
DECEPTIVE PRACTICE DECEPTIVE COLLECTION PRACTICES 589
DECEPTIVE PRACTICE EMBEZZLEMENT 508
DECEPTIVE PRACTICE FINAN EXPLOIT-ELDERLY/DISABLED 716
DECEPTIVE PRACTICE FINANCIAL EXPLOITATION OF AN ELDERLY OR DISABLED PERSON 81
DECEPTIVE PRACTICE FINANCIAL IDENTITY THEFT \$300 AND UNDER 11947
DECEPTIVE PRACTICE FINANCIAL IDENTITY THEFT OVER \$ 300 34651
DECEPTIVE PRACTICE FORGERY 5400
DECEPTIVE PRACTICE FRAUD OR CONFIDENCE GAME 17328
DECEPTIVE PRACTICE ILLEGAL POSSESSION CASH CARD 413
DECEPTIVE PRACTICE ILLEGAL USE CASH CARD 16693
DECEPTIVE PRACTICE IMPERSONATION 313
DECEPTIVE PRACTICE INSURANCE FRAUD 48
DECEPTIVE PRACTICE LIBRARY THEFT 27
DECEPTIVE PRACTICE PAY TV SERVICE OFFENSES 105
DECEPTIVE PRACTICE POSS. KEYS OR DEV.TO COIN MACH 1
DECEPTIVE PRACTICE STOLEN PROP: BUY/RECEIVE/POS. 685
DECEPTIVE PRACTICE STOLEN PROPERTY BUY / RECEIVE / POSSESS 56
DECEPTIVE PRACTICE THEFT BY LESSEE, MOTOR VEHICLE 183
DECEPTIVE PRACTICE THEFT BY LESSEE, NON-MOTOR VEHICLE 3
DECEPTIVE PRACTICE THEFT BY LESSEE,MOTOR VEH 2358
DECEPTIVE PRACTICE THEFT BY LESSEE,NON-VEH 106
DECEPTIVE PRACTICE THEFT OF LABOR / SERVICES 222
DECEPTIVE PRACTICE THEFT OF LABOR/SERVICES 12493
DECEPTIVE PRACTICE THEFT OF LOST / MISLAID PROPERTY 133
DECEPTIVE PRACTICE THEFT OF LOST/MISLAID PROP 2923
DECEPTIVE PRACTICE UNAUTHORIZED VIDEOTAPING 327
DECEPTIVE PRACTICE UNIDENTIFIABLE RECORDING SOUND 118
DECEPTIVE PRACTICE UNLAWFUL USE OF A COMPUTER 194
DECEPTIVE PRACTICE UNLAWFUL USE OF RECORDED SOUND 408
GAMBLING BOOKMAKING/SPORTS 4
GAMBLING GAME / AMUSEMENT DEVICE 1
GAMBLING GAME/AMUSEMENT DEVICE 65
GAMBLING GAME/CARDS 33
GAMBLING GAME/DICE 3319
GAMBLING LOTTERY/OTHER 1
GAMBLING OTHER 83

GAMBLING WIREROOM/SPORTS 1
HOMICIDE FIRST DEGREE MURDER 5562
HOMICIDE INVOLUNTARY MANSLAUGHTER 3
HOMICIDE RECKLESS HOMICIDE 40
HUMAN TRAFFICKING COMMERCIAL SEX ACTS 41
HUMAN TRAFFICKING INVOLUNTARY SERVITUDE 27
INTERFERENCE WITH PUBLIC OFFICER AIDING ARRESTEE ESCAPE 4
INTERFERENCE WITH PUBLIC OFFICER BRIBERY 36
INTERFERENCE WITH PUBLIC OFFICER CONCEALING/AIDING A FUGITIVE 7
INTERFERENCE WITH PUBLIC OFFICER CONTRABAND IN PRISON 2
INTERFERENCE WITH PUBLIC OFFICER ESCAPE 119
INTERFERENCE WITH PUBLIC OFFICER INTERFERENCE JUDICIAL PROCESS 150
INTERFERENCE WITH PUBLIC OFFICER OBSTRUCTING IDENTIFICATION 6679
INTERFERENCE WITH PUBLIC OFFICER OBSTRUCTING JUSTICE 1061
INTERFERENCE WITH PUBLIC OFFICER OBSTRUCTING SERVICE 345
INTERFERENCE WITH PUBLIC OFFICER OFFICIAL MISCONDUCT 13
INTERFERENCE WITH PUBLIC OFFICER REFUSING TO AID AN OFFICER 1
INTERFERENCE WITH PUBLIC OFFICER RESIST / OBSTRUCT / DISARM OFFICER 153
INTERFERENCE WITH PUBLIC OFFICER RESIST/OBSTRUCT/DISARM OFFICER 3180
INTIMIDATION COMPELLING CONFESSION 5
INTIMIDATION COMPELLING ORG MEMBERSHIP 13
INTIMIDATION EDUCATIONAL INTIMIDATION 8
INTIMIDATION EXTORTION 435
INTIMIDATION INTIMIDATION 1005
INTIMIDATION INTIMIDATION OF LAW ENFORCEMENT OFFICIAL 7
KIDNAPPING AGGRAVATED 102
KIDNAPPING CHILD ABDUCTION / STRANGER 18
KIDNAPPING CHILD ABDUCTION/STRANGER 766
KIDNAPPING FORCIBLE DETENTION 26
KIDNAPPING KIDNAPPING 181
KIDNAPPING UNLAWFUL INTERFERE/VISITATION 515
KIDNAPPING UNLAWFUL RESTRAINT 370
KIDNAPPING UNLAWFUL VISITATION INTERFERENCE 26
LIQUOR LAW VIOLATION EMPLOY MINOR 2
LIQUOR LAW VIOLATION ILLEGAL CONSUMPTION BY MINOR 408
LIQUOR LAW VIOLATION ILLEGAL POSSESSION BY MINOR 861
LIQUOR LAW VIOLATION LIQUOR LICENSE VIOLATION 1319
LIQUOR LAW VIOLATION MINOR MISREPRESENT AGE 5
LIQUOR LAW VIOLATION SELL / GIVE / DELIVER LIQUOR TO MINOR 21

LIQUOR LAW VIOLATION SELL/GIVE/DEL LIQUOR TO MINOR 784
MOTOR VEHICLE THEFT ATT: AUTOMOBILE 3879
MOTOR VEHICLE THEFT ATT: TRUCK, BUS, MOTOR HOME 165
MOTOR VEHICLE THEFT ATTEMPT - AUTOMOBILE 124
MOTOR VEHICLE THEFT ATTEMPT - CYCLE, SCOOTER, BIKE WITH VIN 3
MOTOR VEHICLE THEFT ATTEMPT - TRUCK, BUS, MOTOR HOME 1
MOTOR VEHICLE THEFT ATTEMPT: CYCLE, SCOOTER, BIKE NO VIN 5
MOTOR VEHICLE THEFT ATTEMPT: CYCLE, SCOOTER, BIKE W-VIN 41
MOTOR VEHICLE THEFT AUTOMOBILE 101041
MOTOR VEHICLE THEFT CYCLE, SCOOTER, BIKE NO VIN 104
MOTOR VEHICLE THEFT CYCLE, SCOOTER, BIKE W-VIN 2091
MOTOR VEHICLE THEFT CYCLE, SCOOTER, BIKE WITH VIN 250
MOTOR VEHICLE THEFT THEFT / RECOVERY - AUTOMOBILE 452
MOTOR VEHICLE THEFT THEFT / RECOVERY - CYCLE, SCOOTER, BIKE WITH VIN 15
MOTOR VEHICLE THEFT THEFT / RECOVERY - TRUCK, BUS, MOBILE HOME 22
MOTOR VEHICLE THEFT THEFT/RECOVERY: AUTOMOBILE 6245
MOTOR VEHICLE THEFT THEFT/RECOVERY: CYCLE, SCOOTER, BIKE NO VIN 4
MOTOR VEHICLE THEFT THEFT/RECOVERY: CYCLE, SCOOTER, BIKE W-VIN 70
MOTOR VEHICLE THEFT THEFT/RECOVERY: TRUCK,BUS,MHOME 352
MOTOR VEHICLE THEFT TRUCK, BUS, MOTOR HOME 4161
NARCOTICS ALTER / FORGE PRESCRIPTION 10
NARCOTICS ALTER/FORGE PRESCRIPTION 539
NARCOTICS ATTEMPT POSSESSION CANNABIS 42
NARCOTICS ATTEMPT POSSESSION NARCOTICS 544
NARCOTICS CALCULATED CANNABIS CONSPIRACY 4
NARCOTICS CANNABIS PLANT 142
NARCOTICS CONT SUBS:FAIL TO MAINT RECORD 21
NARCOTICS CRIMINAL DRUG CONSPIRACY 369
NARCOTICS DEL CONT SUBS TO PERSON <18 16
NARCOTICS DELIVER CANNABIS TO PERSON <18 22
NARCOTICS FAIL REGISTER LIC:CONT SUBS 1
NARCOTICS FORFEIT PROPERTY 1429
NARCOTICS FOUND SUSPECT NARCOTICS 8505
NARCOTICS MANU/DEL:CANNABIS 10GM OR LESS 6059
NARCOTICS MANU/DEL:CANNABIS OVER 10 GMS 5512
NARCOTICS MANU/DELIVER: HALLUCINOGEN 264
NARCOTICS MANU/DELIVER: HEROIN (WHITE) 12651
NARCOTICS MANU/DELIVER: HEROIN(BRN/TAN) 323
NARCOTICS MANU/DELIVER: METHAMPHETAMINES 75

NARCOTICS MANU/DELIVER:AMPHETAMINES 43
NARCOTICS MANU/DELIVER:BARBITUATES 161
NARCOTICS MANU/DELIVER:COCAINE 734
NARCOTICS MANU/DELIVER:CRACK 4959
NARCOTICS MANU/DELIVER:HEROIN(BLACK TAR) 10
NARCOTICS MANU/DELIVER:LOOK-ALIKE DRUG 27
NARCOTICS MANU/DELIVER:PCP 329
NARCOTICS MANU/DELIVER:SYNTHETIC DRUGS 372
NARCOTICS MANU/POSS. W/INTENT TO DELIVER: SYNTHETIC MARIJUANA 33
NARCOTICS MANUFACTURE / DELIVER - HEROIN (WHITE) 665
NARCOTICS MANUFACTURE / DELIVER - AMPHETAMINES 3
NARCOTICS MANUFACTURE / DELIVER - BARBITURATES 9
NARCOTICS MANUFACTURE / DELIVER - CANNABIS 10 GRAMS OR LESS 62
NARCOTICS MANUFACTURE / DELIVER - CANNABIS OVER 10 GRAMS 355
NARCOTICS MANUFACTURE / DELIVER - COCAINE 42
NARCOTICS MANUFACTURE / DELIVER - CRACK 305
NARCOTICS MANUFACTURE / DELIVER - HALLUCINOGEN 30
NARCOTICS MANUFACTURE / DELIVER - HEROIN (BLACK TAR) 2
NARCOTICS MANUFACTURE / DELIVER - HEROIN (TAN / BROWN TAR) 16
NARCOTICS MANUFACTURE / DELIVER - METHAMPHETAMINE 4
NARCOTICS MANUFACTURE / DELIVER - PCP 7
NARCOTICS MANUFACTURE / DELIVER - SYNTHETIC DRUGS 31
NARCOTICS MANUFACTURE / DELIVER - SYNTHETIC MARIJUANA 1
NARCOTICS POS: HYPODERMIC NEEDLE 169
NARCOTICS POSS: AMPHETAMINES 364
NARCOTICS POSS: BARBITUATES 1441
NARCOTICS POSS: CANNABIS 30GMS OR LESS 84671
NARCOTICS POSS: CANNABIS MORE THAN 30GMS 8436
NARCOTICS POSS: COCAINE 7358
NARCOTICS POSS: CRACK 19794
NARCOTICS POSS: HALLUCINOGENS 1680
NARCOTICS POSS: HEROIN(BLACK TAR) 105
NARCOTICS POSS: HEROIN(BRN/TAN) 1513
NARCOTICS POSS: HEROIN(WHITE) 34313
NARCOTICS POSS: LOOK-ALIKE DRUGS 43
NARCOTICS POSS: METHAMPHETAMINES 602
NARCOTICS POSS: PCP 2134
NARCOTICS POSS: SYNTHETIC DRUGS 3008
NARCOTICS POSSESS - AMPHETAMINES 36

NARCOTICS POSSESS – BARBITURATES 83
NARCOTICS POSSESS – CANNABIS 30 GRAMS OR LESS 125
NARCOTICS POSSESS – CANNABIS MORE THAN 30 GRAMS 414
NARCOTICS POSSESS – COCAINE 251
NARCOTICS POSSESS – CRACK 518
NARCOTICS POSSESS – HALLUCINOGENS 108
NARCOTICS POSSESS – HEROIN (BLACK TAR) 4
NARCOTICS POSSESS – HEROIN (TAN / BROWN TAR) 66
NARCOTICS POSSESS – HEROIN (WHITE) 956
NARCOTICS POSSESS – METHAMPHETAMINE 62
NARCOTICS POSSESS – PCP 55
NARCOTICS POSSESS – SYNTHETIC DRUGS 198
NARCOTICS POSSESSION OF DRUG EQUIPMENT 2928
NARCOTICS POSSESSION: SYNTHETIC MARIJUANA 175
NARCOTICS SALE / DELIVER – DRUG PARAPHERNALIA 2
NARCOTICS SALE/DEL DRUG PARAPHERNALIA 48
NARCOTICS SALE/DEL HYPODERMIC NEEDLE 1
NARCOTICS SOLICIT NARCOTICS ON PUBLIC WAY 110
NARCOTICS SOLICIT NARCOTICS ON PUBLICWAY 5018
NON – CRIMINAL FOID – REVOCATION 38
NON-CRIMINAL CONCEALED CARRY LICENSE REVOCATION 9
NON-CRIMINAL FOID – REVOCATION 38
NON-CRIMINAL FOUND PASSPORT 4
NON-CRIMINAL GUN OFFENDER NOTIFICATION-NO CONTACT 3
NON-CRIMINAL LOST PASSPORT 108
NON-CRIMINAL NOTIFICATION OF STALKING – NO CONTACT ORDER 8
NON-CRIMINAL (SUBJECT SPECIFIED) NOTIFICATION OF CIVIL NO CONTACT ORDER 9
OBSCENITY OBSCENE MATTER 300
OBSCENITY OBSCENITY 85
OBSCENITY SALE / DISTRIBUTE OBSCENE MATERIAL TO MINOR 13
OBSCENITY SALE/DIST OBSCENE MAT TO MINOR 107
OFFENSE INVOLVING CHILDREN AGG CRIM SEX ABUSE FAM MEMBER 1709
OFFENSE INVOLVING CHILDREN AGG SEX ASSLT OF CHILD FAM MBR 939
OFFENSE INVOLVING CHILDREN AGGRAVATED CRIMINAL SEXUAL ABUSE BY FAMILY MEMBER 400
OFFENSE INVOLVING CHILDREN AGGRAVATED SEXUAL ASSAULT OF CHILD BY FAMILY MEMBER 283
OFFENSE INVOLVING CHILDREN CHILD ABANDONMENT 401
OFFENSE INVOLVING CHILDREN CHILD ABDUCTION 1049
OFFENSE INVOLVING CHILDREN CHILD ABUSE 6669
OFFENSE INVOLVING CHILDREN CHILD PORNOGRAPHY 1057

OFFENSE INVOLVING CHILDREN CONTRIBUTE CRIM DELINQUENCY JUVENILE 83
OFFENSE INVOLVING CHILDREN CONTRIBUTE DELINQUENCY OF A CHILD 138
OFFENSE INVOLVING CHILDREN CONTRIBUTE TO THE CRIMINAL DELINQUENCY OF CHILD 4
OFFENSE INVOLVING CHILDREN CONTRIBUTE TO THE DELINQUENCY OF CHILD 9
OFFENSE INVOLVING CHILDREN CRIM SEX ABUSE BY FAM MEMBER 937
OFFENSE INVOLVING CHILDREN CRIMINAL SEXUAL ABUSE BY FAMILY MEMBER 132
OFFENSE INVOLVING CHILDREN ENDANGERING LIFE / HEALTH OF CHILD 243
OFFENSE INVOLVING CHILDREN ENDANGERING LIFE/HEALTH CHILD 2899
OFFENSE INVOLVING CHILDREN HARBOR RUNAWAY 74
OFFENSE INVOLVING CHILDREN OTHER OFFENSE 5141
OFFENSE INVOLVING CHILDREN POS: PORNOGRAPHIC PRINT 53
OFFENSE INVOLVING CHILDREN POSSESSION OF PORNOGRAPHIC PRINT 7
OFFENSE INVOLVING CHILDREN SALE OF TOBACCO PRODUCTS TO MINOR 1
OFFENSE INVOLVING CHILDREN SALE TOBACCO PRODUCTS TO MINOR 16
OFFENSE INVOLVING CHILDREN SEX ASSLT OF CHILD BY FAM MBR 502
OFFENSE INVOLVING CHILDREN SEXUAL ASSAULT OF CHILD BY FAMILY MEMBER 95
OTHER NARCOTIC VIOLATION INTOXICATING COMPOUNDS 61
OTHER OFFENSE ABUSE / NEGLECT - CARE FACILITY 9
OTHER OFFENSE ABUSE/NEGLECT: CARE FACILITY 74
OTHER OFFENSE ANIMAL ABUSE / NEGLECT 53
OTHER OFFENSE ANIMAL ABUSE/NEGLECT 1383
OTHER OFFENSE ANIMAL FIGHTING 12
OTHER OFFENSE ARSONIST - ANNUAL REGISTRATION 2
OTHER OFFENSE ARSONIST - FAIL TO REGISTER NEW ADDRESS 1
OTHER OFFENSE ARSONIST: ANNUAL REGISTRATION 19
OTHER OFFENSE ARSONIST: DUTY TO REGISTER 4
OTHER OFFENSE ARSONIST: FAIL TO REGISTER NEW ADDRESS 4
OTHER OFFENSE BOARD PLANE WITH WEAPON 98
OTHER OFFENSE COMPOUNDING A CRIME 4
OTHER OFFENSE DISCLOSE DV VICTIM LOCATION 1
OTHER OFFENSE EAVESDROPPING 59
OTHER OFFENSE FALSE / STOLEN / ALTERED TRP 177
OTHER OFFENSE FALSE/STOLEN/ALTERED TRP 6935
OTHER OFFENSE FIREARM REGISTRATION VIOLATION 51
OTHER OFFENSE GUN OFFENDER - ANNUAL REGISTRATION 113
OTHER OFFENSE GUN OFFENDER - DUTY TO REGISTER 27
OTHER OFFENSE GUN OFFENDER - DUTY TO REPORT CHANGE OF INFORMATION 20
OTHER OFFENSE GUN OFFENDER: ANNUAL REGISTRATION 1694
OTHER OFFENSE GUN OFFENDER: DUTY TO REGISTER 969

OTHER OFFENSE GUN OFFENDER: DUTY TO REPORT CHANGE OF INFORMATION 416
OTHER OFFENSE HARASSMENT BY ELECTRONIC MEANS 22909
OTHER OFFENSE HARASSMENT BY TELEPHONE 30867
OTHER OFFENSE HAZARDOUS MATERIALS VIOLATION 35
OTHER OFFENSE INTERFERE W/ HIGHER EDUCATION 12
OTHER OFFENSE LICENSE VIOLATION 4866
OTHER OFFENSE MONEY LAUNDERING 33
OTHER OFFENSE OBSCENE TELEPHONE CALLS 451
OTHER OFFENSE OTHER ARSON / EXPLOSIVE INCIDENT 7
OTHER OFFENSE OTHER ARSON/EXPLOSIVE INCIDENT 63
OTHER OFFENSE OTHER CRIME AGAINST PERSON 3671
OTHER OFFENSE OTHER CRIME INVOLVING PROPERTY 2773
OTHER OFFENSE OTHER VEHICLE OFFENSE 12392
OTHER OFFENSE OTHER WEAPONS VIOLATION 3415
OTHER OFFENSE PAROLE VIOLATION 5242
OTHER OFFENSE POSSESSION OF BURGLARY TOOLS 227
OTHER OFFENSE PROBATION VIOLATION 987
OTHER OFFENSE SEX OFFENDER - FAIL TO REGISTER 88
OTHER OFFENSE SEX OFFENDER - FAIL TO REGISTER NEW ADDRESS 26
OTHER OFFENSE SEX OFFENDER: FAIL REG NEW ADD 1761
OTHER OFFENSE SEX OFFENDER: FAIL TO REGISTER 2037
OTHER OFFENSE SEX OFFENDER: PROHIBITED ZONE 69
OTHER OFFENSE TAMPER WITH MOTOR VEHICLE 214
OTHER OFFENSE TELEPHONE THREAT 44299
OTHER OFFENSE UNLAWFUL USE OF BODY ARMOR 7
OTHER OFFENSE VEHICLE TITLE / REGISTRATION OFFENSE 164
OTHER OFFENSE VEHICLE TITLE/REG OFFENSE 4504
OTHER OFFENSE VIO BAIL BOND: DOM VIOLENCE 203
OTHER OFFENSE VIOLATE ORDER OF PROTECTION 15610
OTHER OFFENSE VIOLATION GPS MONITORING DEVICE 108
OTHER OFFENSE VIOLATION OF BAIL BOND - DOMESTIC VIOLENCE 23
OTHER OFFENSE VIOLATION OF CIVIL NO CONTACT ORDER 489
OTHER OFFENSE VIOLATION OF SMOKING BAN 25
OTHER OFFENSE VIOLATION OF SUMMARY CLOSURE 11
OTHER OFFENSE VIOLENT OFFENDER - ANNUAL REGISTRATION 20
OTHER OFFENSE VIOLENT OFFENDER - DUTY TO REGISTER 4
OTHER OFFENSE VIOLENT OFFENDER - FAIL TO REGISTER NEW ADDRESS 2
OTHER OFFENSE VIOLENT OFFENDER: ANNUAL REGISTRATION 248
OTHER OFFENSE VIOLENT OFFENDER: DUTY TO REGISTER 118

OTHER OFFENSE VIOLENT OFFENDER: FAIL TO REGISTER NEW ADDRESS 59

PROSTITUTION CAB OPERATION 1

PROSTITUTION CALL OPERATION 739

PROSTITUTION IN TAVERN 3

PROSTITUTION JUVENILE PIMPING 4

PROSTITUTION KEEP PLACE OF PROSTITUTION 26

PROSTITUTION OTHER PROSTITUTION OFFENSE 120

PROSTITUTION PANDERING 11

PROSTITUTION PATRONIZE JUVENILE PROSTITUTE 6

PROSTITUTION PATRONIZING A PROSTITUTE 67

PROSTITUTION PIMPING 18

PROSTITUTION SOLICIT FOR BUSINESS 1510

PROSTITUTION SOLICIT FOR PROSTITUTE 1325

PROSTITUTION SOLICIT OFF PUBLIC WAY 60

PROSTITUTION SOLICIT ON PUBLIC WAY 8539

PROSTITUTION SOLICITING FOR A PROSTITUTE 2

PROSTITUTION SOLICITING FOR BUSINESS 2

PUBLIC INDECENCY LICENSED PREMISE 118

PUBLIC PEACE VIOLATION ARMED VIOLENCE 3

PUBLIC PEACE VIOLATION ARSON THREAT 1120

PUBLIC PEACE VIOLATION BOMB THREAT 1419

PUBLIC PEACE VIOLATION FALSE FIRE ALARM 238

PUBLIC PEACE VIOLATION FALSE POLICE REPORT 928

PUBLIC PEACE VIOLATION INTERFERE W/ EMERGENCY EQUIP 12

PUBLIC PEACE VIOLATION INTERFERE WITH EMERGENCY EQUIPMENT 4

PUBLIC PEACE VIOLATION LOOTING 395

PUBLIC PEACE VIOLATION MOB ACTION 456

PUBLIC PEACE VIOLATION OTHER VIOLATION 1108

PUBLIC PEACE VIOLATION PEEPING TOM 411

PUBLIC PEACE VIOLATION PUBLIC DEMONSTRATION 92

PUBLIC PEACE VIOLATION RECKLESS CONDUCT 15574

PUBLIC PEACE VIOLATION SELL/ADVERTISE FIREWORKS 19

Primary Type Description 1

RITUALISM AGG. RITUAL MUTILATION - HANDS, FISTS, FEET, SERIOUS INJURY 1

ROBBERY AGGRAVATED 6032

ROBBERY AGGRAVATED VEHICULAR HIJACKING 3870

ROBBERY ARMED - HANDGUN 2374

ROBBERY ARMED - KNIFE / CUTTING INSTRUMENT 276

ROBBERY ARMED - OTHER DANGEROUS WEAPON 251

ROBBERY ARMED – OTHER FIREARM 41
ROBBERY ARMED: HANDGUN 36490
ROBBERY ARMED: OTHER DANGEROUS WEAPON 4400
ROBBERY ARMED: OTHER FIREARM 539
ROBBERY ARMED:KNIFE/CUTTING INSTRUMENT 4239
ROBBERY ATTEMPT AGGRAVATED 39
ROBBERY ATTEMPT ARMED – HANDGUN 254
ROBBERY ATTEMPT ARMED – KNIFE / CUTTING INSTRUMENT 62
ROBBERY ATTEMPT ARMED – OTHER DANGEROUS WEAPON 32
ROBBERY ATTEMPT ARMED – OTHER FIREARM 10
ROBBERY ATTEMPT ARMED: HANDGUN 1
ROBBERY ATTEMPT STRONG ARM – NO WEAPON 196
ROBBERY ATTEMPT STRONG ARM: NO WEAPON 1
ROBBERY ATTEMPT: AGGRAVATED 577
ROBBERY ATTEMPT: ARMED-HANDGUN 3133
ROBBERY ATTEMPT: ARMED-KNIFE/CUT INSTR 846
ROBBERY ATTEMPT: ARMED-OTHER DANG WEAP 602
ROBBERY ATTEMPT: ARMED-OTHER FIREARM 106
ROBBERY ATTEMPT: STRONGARM-NO WEAPON 3698
ROBBERY STRONG ARM – NO WEAPON 1806
ROBBERY STRONGARM – NO WEAPON 35247
ROBBERY STRONGARM: NO WEAPON 4
ROBBERY VEHICULAR HIJACKING 2385
SEX OFFENSE AGG CRIMINAL SEXUAL ABUSE 1870
SEX OFFENSE AGGRAVATED CRIMINAL SEXUAL ABUSE 247
SEX OFFENSE ATT AGG CRIMINAL SEXUAL ABUSE 69
SEX OFFENSE ATT CRIM SEXUAL ABUSE 271
SEX OFFENSE ATTEMPT AGGRAVATED CRIMINAL SEXUAL ABUSE 3
SEX OFFENSE ATTEMPT CRIMINAL SEXUAL ABUSE 25
SEX OFFENSE BIGAMY 10
SEX OFFENSE CRIMINAL SEXUAL ABUSE 4000
SEX OFFENSE CRIMINAL TRANSMISSION OF HIV 64
SEX OFFENSE FORNICATION 1
SEX OFFENSE INDECENT SOLICITATION OF A CHILD 27
SEX OFFENSE INDECENT SOLICITATION OF AN ADULT 2
SEX OFFENSE INDECENT SOLICITATION/ADULT 43
SEX OFFENSE INDECENT SOLICITATION/CHILD 260
SEX OFFENSE NON-CONSENSUAL DISSEMINATION OF PRIVATE SEXUAL IMAGES 177
SEX OFFENSE NON-CONSENSUAL DISSEMINATION PRIVATE SEXUAL IMAGES 246

SEX OFFENSE OTHER 569
SEX OFFENSE PUBLIC INDECENCY 2188
SEX OFFENSE SEX RELATION IN FAMILY 27
SEX OFFENSE SEXUAL EXPLOITATION OF A CHILD 460
SEX OFFENSE SEXUAL RELATIONS IN FAMILY 6
STALKING AGGRAVATED 47
STALKING CYBERSTALKING 222
STALKING SIMPLE 1333
STALKING VIOLATION OF STALKING NO CONTACT ORDER 210
THEFT \$500 AND UNDER 252894
THEFT AGG: FINANCIAL ID THEFT 461
THEFT ATTEMPT FINANCIAL IDENTITY THEFT 1018
THEFT ATTEMPT THEFT 4594
THEFT DELIVERY CONTAINER THEFT 376
THEFT FINANCIAL ID THEFT: OVER \$300 9506
THEFT FINANCIAL ID THEFT:\$300 &UNDER 3074
THEFT FINANCIAL IDENTITY THEFT: OVER \$300 1
THEFT FROM BUILDING 98866
THEFT FROM COIN-OP MACHINE/DEVICE 241
THEFT FROM COIN-OPERATED MACHINE OR DEVICE 13
THEFT OVER \$500 145142
THEFT POCKET-PICKING 19672
THEFT PURSE-SNATCHING 6782
THEFT RETAIL THEFT 89886
THEFT THEFT RETAIL 1
WEAPONS VIOLATION DEFACE IDENT MARKS OF FIREARM 22
WEAPONS VIOLATION DEFACE IDENTIFICATION MARKS OF FIREARM 7
WEAPONS VIOLATION POSS FIREARM/AMMO:NO FOID CARD 784
WEAPONS VIOLATION POSSESS FIREARM / AMMUNITION – NO FOID CARD 56
WEAPONS VIOLATION RECKLESS FIREARM DISCHARGE 4365
WEAPONS VIOLATION UNLAWFUL POSS AMMUNITION 979
WEAPONS VIOLATION UNLAWFUL POSS OF HANDGUN 26642
WEAPONS VIOLATION UNLAWFUL POSS OTHER FIREARM 1048
WEAPONS VIOLATION UNLAWFUL POSSESSION – AMMUNITION 52
WEAPONS VIOLATION UNLAWFUL POSSESSION – HANDGUN 4121
WEAPONS VIOLATION UNLAWFUL POSSESSION – OTHER FIREARM 72
WEAPONS VIOLATION UNLAWFUL SALE – DELIVERY OF FIREARM AT SCHOOL 2
WEAPONS VIOLATION UNLAWFUL SALE – HANDGUN 4
WEAPONS VIOLATION UNLAWFUL SALE – OTHER FIREARM 1

WEAPONS VIOLATION UNLAWFUL SALE HANDGUN 70
WEAPONS VIOLATION UNLAWFUL SALE OTHER FIREARM 14
WEAPONS VIOLATION UNLAWFUL SALE/DELIVERY OF FIREARM AT SCHOOL 20
WEAPONS VIOLATION UNLAWFUL USE – HANDGUN 531
WEAPONS VIOLATION UNLAWFUL USE – OTHER DANGEROUS WEAPON 28
WEAPONS VIOLATION UNLAWFUL USE – OTHER FIREARM 218
WEAPONS VIOLATION UNLAWFUL USE / SALE OF AIR RIFLE 9
WEAPONS VIOLATION UNLAWFUL USE HANDGUN 2838
WEAPONS VIOLATION UNLAWFUL USE OTHER DANG WEAPON 2146
WEAPONS VIOLATION UNLAWFUL USE OTHER FIREARM 468
WEAPONS VIOLATION UNLAWFUL USE/SALE AIR RIFLE 575
WEAPONS VIOLATION USE OF METAL PIERCING BULLETS 1

Query 3:

Crime Count by Primary Type and year:

```
create table query4 as
select year_crime,primary_type,count(*) as CrimeCount
from crimetsv
group by year_crime,primary_type order by year_crime,primary_type;
```

2011	ARSON	504
2011	ASSAULT	20411
2011	BATTERY	60458
2011	BURGLARY	26620
2011	CRIM SEXUAL ASSAULT	1478
2011	CRIMINAL DAMAGE	37332
2011	CRIMINAL SEXUAL ASSAULT	16
2011	CRIMINAL TRESPASS	8659
2011	DECEPTIVE PRACTICE	12703
2011	GAMBLING	736
2011	HOMICIDE	438
2011	INTERFERENCE WITH PUBLIC OFFICER	1048
2011	INTIMIDATION	171
2011	KIDNAPPING	266
2011	LIQUOR LAW VIOLATION	619
2011	MOTOR VEHICLE THEFT	19388
2011	NARCOTICS	38605
2011	OBSCENITY	40
2011	OFFENSE INVOLVING CHILDREN	2384
2011	OTHER NARCOTIC VIOLATION	5
2011	OTHER OFFENSE	20192
2011	PROSTITUTION	2424
2011	PUBLIC INDECENCY	13
2011	PUBLIC PEACE VIOLATION	3095
2011	ROBBERY	13983
2011	SEX OFFENSE	1082
2011	STALKING	181
2011	THEFT	75150
2011	WEAPONS VIOLATION	3880
2012	ARSON	469
2012	ASSAULT	19899
2012	BATTERY	59137
2012	BURGLARY	22845
2012	CRIM SEXUAL ASSAULT	1413
2012	CRIMINAL DAMAGE	35854
2012	CRIMINAL SEXUAL ASSAULT	22
2012	CRIMINAL TRESPASS	8216
2012	DECEPTIVE PRACTICE	13735
2012	GAMBLING	724
2012	HOMICIDE	515
2012	INTERFERENCE WITH PUBLIC OFFICER	1228
2012	INTIMIDATION	156
2012	KIDNAPPING	235
2012	LIQUOR LAW VIOLATION	573
2012	MOTOR VEHICLE THEFT	16492

Query 5:

```
create table query5 as
select a.year_crime,CrimeCount,ArrestCount from
```

```

(select year_crime, count(*) as CrimeCount
from crimetsv
group by year_crime)a

join

(select year_crime, count(*) as ArrestCount
from crimetsv
where arrest='True'
group by year_crime)b

on a.year_crime=b.year_crime
order by year_crime;

```

Query 6:

Crime Count that involved GUNS and Crime Count that involved violence with GUNS - Year Wise:

```

create table query6 as

select Year_crime, Count (case_number) as TotalCrimes,
SUM ( if ( (DESCRIPTION LIKE '%FIREARM%' OR
DESCRIPTION LIKE '%GUN%'), 1 , 0 ) ) AS TotalWithGuns,
SUM ( if (
(primary_type = 'ASSAULT' AND (DESCRIPTION LIKE '%FIREARM%' OR DESCRIPTION LIKE '%GUN%') AND (NOT DESCRIPTION LIKE '%SALE%' OR NOT
DESCRIPTION LIKE '%POS%'))
OR
(primary_type = 'CRIM SEXUAL ASSAULT' AND (DESCRIPTION LIKE '%FIREARM%' OR DESCRIPTION LIKE '%GUN%') AND (NOT DESCRIPTION LIKE
'%SALE%' OR NOT DESCRIPTION LIKE '%POS%'))
OR
(primary_type = 'OFFENSE INVOLVING CHILDREN' AND (DESCRIPTION LIKE '%FIREARM%' OR DESCRIPTION LIKE '%GUN%') AND (NOT DESCRIPTION
LIKE "%SALE%" OR NOT DESCRIPTION LIKE "%POS%"))
OR
(primary_type = 'OTHER OFFENSE' AND (DESCRIPTION LIKE '%FIREARM%' OR DESCRIPTION LIKE '%GUN%') AND (NOT DESCRIPTION LIKE '%SALE%' OR NOT
DESCRIPTION LIKE '%POS%'))
OR
(primary_type = 'RITUALISM' AND (DESCRIPTION LIKE '%FIREARM%' OR DESCRIPTION LIKE '%GUN%') AND (NOT DESCRIPTION LIKE '%SALE%' OR NOT
DESCRIPTION LIKE '%POS%'))
OR
(primary_type = 'ROBBERY' AND (DESCRIPTION LIKE '%FIREARM%' OR DESCRIPTION LIKE '%GUN%') AND (NOT DESCRIPTION LIKE '%SALE%' OR NOT
DESCRIPTION LIKE '%POS%'))

```

```
(primary_type = 'WEAPONS VIOLATION' AND (DESCRIPTION LIKE '%FIREARM%' OR DESCRIPTION LIKE '%GUN%') AND (NOT DESCRIPTION LIKE '%SALE%' OR NOT DESCRIPTION LIKE '%POS%'))  
, 1 , 0)) AS TotalViolentWlthGuns  
from crimetsv  
group by Year_crime  
order by Year_crime ;
```

```
hive> select * from query6;  
OK  
2011      351881    12871     7881  
2012      336149    13017     7817  
2013      307310    10534     6459  
2014      275567    10364     6091  
2015      264462    11419     6560  
2016      269477    14654     8737  
2017      268694    15171     8648  
2018      268254    14696     7662  
2019      260444    14526     6783  
2020      191744    16393     6555
```

Query 8 :

```

create table query8 as
select location_description,count(location_description) as Domestic_Count
from crimetsv
group by location_description;

```

Crime Count by Location wise like Hotel, Parking Lot or Driveway:

CONVENIENCE STORE	15728
HOTEL	12
LAGOON	1
OFFICE	4
OTHER RAILROAD PROP / TRAIN DEPOT	2051
POLICE FACILITY/VEH PARKING LOT	8129
PORCH	180
RESIDENCE - PORCH / HALLWAY	2984
SAVINGS AND LOAN	124
SCHOOL - PUBLIC BUILDING	459
VACANT LOT / LAND	456
AIRPORT TERMINAL LOWER LEVEL - NON-SECURE AREA	1660
CHA PLAY LOT	2
CTA PROPERTY	4
DRIVEWAY	10
Location Description	1
RESIDENCE PORCH/HALLWAY	46134
VEHICLE - OTHER RIDE SHARE SERVICE (E.G., UBER, LYFT)	470
VESTIBULE	11
WOODED AREA	2
ATHLETIC CLUB	4884
COIN OPERATED MACHINE	289
COLLEGE/UNIVERSITY GROUNDS	1846
CTA BUS STOP	6314
ELEVATOR	1
MOVIE HOUSE/THEATER	962
OTHER RAILROAD PROPERTY / TRAIN DEPOT	114
PARKING LOT/GARAGE(NON.RESID.)	76299
VEHICLE - COMMERCIAL: TROLLEY BUS	2
AIRPORT TRANSPORTATION SYSTEM (ATS)	86
AUTO	549
CTA SUBWAY STATION	1
FEDERAL BUILDING	363
GARAGE	24
NEWSSTAND	62
PUBLIC HIGH SCHOOL	1

Query 9:

Crime Count by Exact Location on Map –

This will help to know the hotspots of Crime in Chicago which will be helping to have more patrolling in such areas

```
create table query9 as
select district,latitude,longitude,count(*) Crimecount
from crimetsv
group by district,latitude,longitude having Crimecount >= 100 ;
```

```
hive> select * from query9;
OK
001      41.841164277      -87.6171782      162
001      41.858594819      -87.627214427      370
001      41.872910562      -87.63062942      132
001      41.882389156      -87.624374606      484
001      41.882470918      -87.627847625      572
001      41.884276844      -87.622098929      257
001      41.885369466      -87.63243193      255
001      41.885888079      -87.627942238      572
001      41.886169843      -87.620686081      151
001      41.888165132      -87.622937212      542
002      41.80062644       -87.590467932      121
002      41.833139425      -87.615319213      271
003      41.765651252      -87.624909847      128
003      41.765843381      -87.576291395      145
003      41.77999793       -87.629294594      151
004      41.722133485      -87.59934358      121
004      41.722362598      -87.58382047      372
004      41.73365047       -87.557845321      402
004      41.752174882      -87.551824707      135
005      41.705267858      -87.629323117      107
005      41.707095328      -87.628834415      388
006      41.721627204      -87.624485177      758
006      41.735752519      -87.625429032      233
006      41.735871734      -87.657534129      122
006      41.73614018       -87.62911796      217
006      41.74719631       -87.602638387      127
006      41.750396912      -87.653735437      100
007      41.760065748      -87.663661519      125
007      41.772972076      -87.644610871      678
007      41.779012821      -87.664208692      475
007      41.786499587      -87.649815884      336
008      41.764245207      -87.683232026      133
008      41.778661058      -87.68227253      187
008      41.79798374       -87.723294914      150
009      41.794172544      -87.70357593      235
009      41.830954079      -87.630703226      568
010      41.851393305      -87.723903515      265
010      41.854864309      -87.724846958      206
```

Query 11:

The below Analysis on Arrest Data
Count on Charge-1 statute or clause

```
create table query11 as
select charge1statute,count(*) CaseCount
from arrest
group by charge1statute having CaseCount>1000;
```

```
hive> select * from query11;
OK
4-64-350          1094
720  ILCS 5.0/16-2      2141
720  ILCS 5.0/19-3-A    1358
625  ILCS 5.0/6-101     3245
720  ILCS 5.0/12-3-A-1   9744
720  ILCS 550.0/4-A     6180
725  ILCS 225.0/13      3904
720  ILCS 5.0/12-2-C-1   5333
720  ILCS 5.0/12-3-A-2   13713
720  ILCS 5.0/21-2-A     8449
720  ILCS 550.0/4-B     10737
8-26-030-B          1238
720  ILCS 550.0/4-C     6140
720  ILCS 550.0/5-B     2270
720  ILCS 570.0/402-C    41917
720  ILCS 5.0/12-2-B-4   1063
720  ILCS 550.0/4-D     2001
720  ILCS 550.0/5-C     2032
720  ILCS 5.0/21-5-A     3170
720  ILCS 5.0/24-1.7-A   1570
720  ILCS 550.0/5-D     2076
10-8-526            1584
720  ILCS 5.0/25-5-A-4   6706
625  ILCS 5.0/6-303-A    45137
720  ILCS 5.0/16-25-A-1   23127
625  ILCS 5.0/11-501-A    4678
625  ILCS 5.0/11-501-A-1  2796
720  ILCS 5.0/12-1-A     9689
720  ILCS 5.0/18-2-A-2    1322
720  ILCS 5.0/24-1-A-4    1040
720  ILCS 5.0/9-1-A-1    1642
```

Query 12:

```
create table query12 as  
select a.case_number,a.race,a.charge1type,c.district,c.primary_type,c.description,c.year_crime from  
arrest a join crimetsv c on a.case_number = c.case_number;
```

Here I joined Crime Data with Arrest Data on Case Number

```
hive> select * from query12 limit 10;  
OK  
HN329056      BLACK   F    002      HOMICIDE      FIRST DEGREE MURDER      2016  
HV211634      WHITE  HISPANIC F    025      ROBBERY      ARMED:KNIFE/CUTTING INSTRUMENT  2012  
HV247084      BLACK   F    006      BURGLARY      FORCIBLE ENTRY      2012  
HV379031      BLACK   F    002      BURGLARY      UNLAWFUL ENTRY      2012  
HV444785      WHITE   F    009      CRIM SEXUAL ASSAULT  AGGRAVATED: OTHER      2012  
HV487456      BLACK   F    018      BURGLARY      FORCIBLE ENTRY      2012  
HV506122      BLACK   F    011      CRIM SEXUAL ASSAULT  PREDATORY      2012  
HV542636      WHITE  HISPANIC F    002      OTHER OFFENSE  SEX OFFENDER: FAIL REG NEW ADD  2012  
HW146845      BLACK   M    015      BURGLARY      UNLAWFUL ENTRY      2013  
HW146845      BLACK   M    015      BURGLARY      UNLAWFUL ENTRY      2013  
Time taken: 0.12 seconds, Fetched: 10 row(s)  
hive> █
```

Query13:

```
create table query13 as  
select race, primary_type, Count(*) from query12 group by race, primary_type;
```

```
hive> select * from query13;
OK
ASIAN / PACIFIC ISLANDER      HOMICIDE      1
ASIAN / PACIFIC ISLANDER      OBSCENITY     1
BLACK   LIQUOR LAW VIOLATION 80
BLACK   NON-CRIMINAL      3
UNKNOWN / REFUSED      GAMBLING      1
WHITE HISPANIC CRIM SEXUAL ASSAULT 213
AMER INDIAN / ALASKAN NATIVE MOTOR VEHICLE THEFT 1
AMER INDIAN / ALASKAN NATIVE PUBLIC PEACE VIOLATION 2
AMER INDIAN / ALASKAN NATIVE WEAPONS VIOLATION 5
ASIAN / PACIFIC ISLANDER      CONCEALED CARRY LICENSE VIOLATION 2
BLACK   CRIMINAL DAMAGE 5804
BLACK   CRIMINAL SEXUAL ASSAULT 84
BLACK   PROSTITUTION      1054
BLACK   SEX OFFENSE      498
BLACK HISPANIC INTERFERENCE WITH PUBLIC OFFICER 27
BLACK HISPANIC OFFENSE INVOLVING CHILDREN 14
UNKNOWN / REFUSED      CRIMINAL TRESPASS 42
WHITE HISPANIC BATTERY 11352
ASIAN / PACIFIC ISLANDER      SEX OFFENSE      17
BLACK   GAMBLING      2493
BLACK   NON - CRIMINAL 2
BLACK HISPANIC MOTOR VEHICLE THEFT 27
UNKNOWN / REFUSED      OFFENSE INVOLVING CHILDREN 7
WHITE   INTERFERENCE WITH PUBLIC OFFICER 254
WHITE   OFFENSE INVOLVING CHILDREN 141
AMER INDIAN / ALASKAN NATIVE HOMICIDE      1
BLACK   CRIMINAL TRESPASS 18877
BLACK HISPANIC OTHER OFFENSE 113
UNKNOWN / REFUSED      ARSON 1
UNKNOWN / REFUSED      ASSAULT 39
UNKNOWN / REFUSED      BATTERY 114
WHITE   ARSON 36
WHITE   ASSAULT 1652
WHITE   PUBLIC INDECENCY 2
WHITE HISPANIC MOTOR VEHICLE THEFT 746
WHITE HISPANIC WEAPONS VIOLATION 3492
AMER INDIAN / ALASKAN NATIVE SEX OFFENSE 2
ASIAN / PACIFIC ISLANDER      INTERFERENCE WITH PUBLIC OFFICER 15
ASIAN / PACIFIC ISLANDER      OFFENSE INVOLVING CHILDREN 15
```

Query 14:

```
create table query14 as  
select charge1type, primary_type, Count(*) from query12 group by charge1type, primary_type;
```

	CRIMINAL DAMAGE	259
	CRIMINAL SEXUAL ASSAULT	11
	PROSTITUTION	612
	SEX OFFENSE	69
F	HOMICIDE	1022
F	OBSCENITY	6
R	OTHER OFFENSE	3
	GAMBLING	287
	KIDNAPPING	5
F	CONCEALED CARRY LICENSE VIOLATION	55
F	ROBBERY	4083
F	STALKING	24
	CRIMINAL TRESPASS	295
M	INTERFERENCE WITH PUBLIC OFFICER	5210
M	OFFENSE INVOLVING CHILDREN	1260
R	SEX OFFENSE	1
A	BURGLARY	2
A	NARCOTICS	3
F	HUMAN TRAFFICKING	5
M	MOTOR VEHICLE THEFT	3158
	CRIM SEXUAL ASSAULT	21
	OTHER NARCOTIC VIOLATION	2
M	ARSON	46
M	ASSAULT	17297
M	PUBLIC INDECENCY	26
	BATTERY	729
F	INTERFERENCE WITH PUBLIC OFFICER	345
F	OFFENSE INVOLVING CHILDREN	517
M	OTHER OFFENSE	12453
	DECEPTIVE PRACTICE	186
F	MOTOR VEHICLE THEFT	1234
M	LIQUOR LAW VIOLATION	185
M	NON-CRIMINAL	3
M	THEFT	25341
	INTIMIDATION	2
F	ARSON	158
F	ASSAULT	1896
M	CRIMINAL DAMAGE	7964
M	CRIMINAL SEXUAL ASSAULT	30
M	PROSTITUTION	1026
M	SEX OFFENSE	592
	BURGLARY	96505
	NARCOTICS	4850
	PUBLIC PEACE VIOLATION	1212
	WEAPONS VIOLATION	736
F	OTHER OFFENSE	2222

Pig Analysis:

Analysis – 1 :

CrimeData =

```
crimeData = LOAD '/chicagoCrimeHive/chicagoCrime' As  
(Id,CaseNumber,Date,Block,IUCR,PrimaryType,Description,LocationDescription,Arrest,Domestic,Beat,District,Ward,CommunityArea,FBICode,XCoordinate,YCoordinate,Year,UpdatedOn,Latitude,Longitude,Location);
```

ArrestData =

```
arrestData = LOAD '/chicagoCrimeArrests/Arrests' As (CB_N0,CaseNumber, ArrestDate, Race,  
Charge1Statute,Charge1Description,Charge1Type,Charge1Class,Charge2Statute,Charge2Description,Char  
ge2Type,Charge2Class,Charge3Statute,Charge3Description,Charge3Type,Charge3Class,Charge4Statute,Ch  
arge4Description,Charge4Type,Charge4Class,ChargesStatute,ChargesDescription,ChargesType,ChargesCl  
ass);
```

The Below analysis is Used to Count Total Number of Arrests by Counting Domestic Violence kind of analysis in Charge_1_Description.

```
8 crimeData = LOAD '/chicagoCrimeHive/chicagoCrime' As [Id,CaseNumber,Date,Block,IUCR,PrimaryType,Description,LocationDescription,Arrest,Domestic,Beat,District,Ward,CommunityArea,FBICode,XCoordinate,YCoordinate,Year,UpdatedOn,Latitude,L  
ongitude,Location];  
9 group_District_Dates = GROUP crimeData BY District;  
10 group_District_Dates = LOAD group_District_Dates AS (CB_N0,CASENUMBER, ArrestDate, Race, Charge1Statute,Charge1Description,Charge1Type,Charge1Class,Charge2Statute,Charge2Description,Charge2Type,Charge2Class,Charge3Statute,Charge3Description,Charge3Type,Charge3Class,Charge4Statute,Charge4Description,Charge4Type,Charge4Class,ChargesStatute,ChargesDescription,ChargesType,ChargesClass);  
11 ArrestCrimeData = JOIN crimeData BY CaseNumber , arrestData BY CaseNumber  
12 filter_ArrestCrimeData.License = FILTER ArrestCrimeData BY ChargeDescription matches '+License.';  
13 filter_ArrestCrimeData.License = FILTER ArrestCrimeData BY ChargeDescription matches 'License';  
14 filter_ArrestCrimeData.License = FILTER ArrestCrimeData BY ChargeDescription matches '+License.';  
15 filter_ArrestCrimeData.License = FILTER ArrestCrimeData BY ChargeDescription matches '+DOMESTIC BATTERY.';  
16 filter_ArrestCrimeData.License_Count = Group filter_ArrestCrimeData.License ALL;  
17 filter_ArrestCrimeData.License_Count_DomesticBattery = FOREACH filter_ArrestCrimeData_License_Count GENERATE COUNT (filter_ArrestCrimeData_License);  
grunts; ||
```

```
crimeData = LOAD '/chicagoCrimeHive/chicagoCrime' As  
(Id,CaseNumber,Date,Block,IUCR,PrimaryType,Description,LocationDescription,Arrest,Domestic,Beat,District,Ward,CommunityArea,FBICode,XCoordinate,YCoordinate,Year,UpdatedOn,Latitude,Longitude,Location);
```

```
arrestData = LOAD '/chicagoCrimeArrests/Arrests' As (CB_N0,CaseNumber, ArrestDate, Race,  
Charge1Statute,Charge1Description,Charge1Type,Charge1Class,Charge2Statute,Charge2Description,Char  
ge2Type,Charge2Class,Charge3Statute,Charge3Description,Charge3Type,Charge3Class,Charge4Statute,Ch  
arge4Description,Charge4Type,Charge4Class,ChargesStatute,ChargesDescription,ChargesType,ChargesCl  
ass);
```

```

ArrestCrimeData = JOIN crimeData by CaseNumber , arrestData by CaseNumber
;

Filter_ArrestCrimeData_License = FILTER ArrestCrimeData BY Charge1Description matches '.*DOMESTIC
BATTERY.*';

Filter_ArrestCrimeData_License_Count = Group Filter_ArrestCrimeData_License ALL;

Filter_ArrestCrimeData_License_Count_DomesticBattery = FOREACH
Filter_ArrestCrimeData_License_Count GENERATE COUNT (Filter_ArrestCrimeData_License);

```

```
2020-12-15 19:25:36,639 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(32011)
```

Analysis – 2:

This analysis is used to Count total charges for trespassing in Chicago.

```

2020-12-15 20:26:46,779 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!
2020-12-15 20:26:46,780 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!
2020-12-15 20:26:46,781 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!
2020-12-15 20:26:46,783 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!
2020-12-15 20:26:46,784 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!
2020-12-15 20:26:46,785 [main] WARN org.apache.hadoop.metrics2.impl.MetricsSystemImpl - JobTracker metrics system already initialized!
2020-12-15 20:26:46,785 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
2020-12-15 20:26:46,786 [main] WARN org.apache.pig.data.SchemaTupleBackend - SchemaTupleBackend has already been initialized
2020-12-15 20:26:46,789 [main] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input files to process : 1
2020-12-15 20:26:46,789 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
(35547)
grunt> ■

```

```

19 Filter_ArrestCrimeData.TresPass_Count = Group Filter_ArrestCrimeData.TresPass ALL;
20 Filter_ArrestCrimeData.TresPass_Count_FE = FOREACH Filter_ArrestCrimeData.TresPass_Count GENERATE COUNT (Filter_ArrestCrimeData.TresPass);
21 Filter_ArrestCrimeData.TresPass = FILTER ArrestCrimeData BY Charge1Description matches '.*TRESPASS.*' or Charge2Description matches '.*TRESPASS.*' or Charge3Description matches '.*TRESPASS.*' or Charge4Description matches '.*TRESPASS.*';
22 Filter_ArrestCrimeData.TresPass_Count = Group Filter_ArrestCrimeData.TresPass ALL;
23 Filter_ArrestCrimeData.TresPass_Count_FE = FOREACH Filter_ArrestCrimeData.TresPass_Count GENERATE COUNT (filter_ArrestCrimeData.TresPass);
grunts> ■

```

Mahout Analysis -

```
Arrest ~
0.409*Primary Type=ARSON + 1.538*Primary Type=ASSAULT + 1.538*Primary Type=BATTERY + 1.538*Primary Type=BURGLARY + 1.538*Primary Type=CONCEALED CARRY LICENSE VIOLATION + 1.538*Primary Type=CRIM SEXUAL ASSAULT + 1.538*Primary Type=CRIMINAL DAMAGE + 0.409*Primary Type=CRIMINAL SEXUAL ASSAULT + 0.409*Primary Type=CRIMINAL TRESPASS + 1.538*Primary Type=DECEPTIVE PRACTICE + 1.538*Primary Type=GAMBLING + 0.409*Primary Type=HOMICIDE + 1.129*Primary Type=HUMAN TRAFFICKING + 1.129*Primary Type=INTERFERENCE WITH PUBLIC OFFICER + 1.129*Primary Type=INTIMIDATION + 1.538*Primary Type=KIDNAPPING + 1.129*Primary Type=LIQUOR LAW VIOLATION + 0.409*Primary Type=MOTOR VEHICLE THEFT + 0.409*Primary Type=NARCOTICS + 1.538*Primary Type=NON - CRIMINAL + 0.409*Primary Type=NON-CRIMINAL + 1.129*Primary Type=NON-CRIMINAL (SUBJECT SPECIFIED) + 1.129*Primary Type=OBSCENITY + 0.409*Primary Type=OFFENSE INVOLVING CHILDREN + 1.538*Primary Type=OTHER NARCOTIC VIOLATION + 1.129*Primary Type=OTHER OFFENSE + 1.538*Primary Type=PROSTITUTION + 1.538*Primary Type=PUBLIC INDECENCY + 1.538*Primary Type=PUBLIC PEACE VIOLATION + 1.129*Primary Type=RITUALISM + 1.129*Primary Type=ROBBERY + 1.129*Primary Type=SEX OFFENSE + 1.538*Primary Type=STALKING + 1.129*Primary Type=THEFT + 0.409*Primary Type=UNLAWFUL ENTRY + 1.538*Primary Type=WEAPONS VIOLATION
Primary Type=ARSON 0.40948
Primary Type=ASSAULT 1.53842
Primary Type=BATTERY 1.53842
Primary Type=BURGLARY 1.53842
Primary Type=CONCEALED CARRY LICENSE VIOLATION 1.53842
Primary Type=CRIM SEXUAL ASSAULT 1.53842
Primary Type=CRIMINAL DAMAGE 1.53842
Primary Type=CRIMINAL SEXUAL ASSAULT 0.40948
Primary Type=CRIMINAL TRESPASS 0.40948
Primary Type=DECEPTIVE PRACTICE 1.53842
Primary Type=GAMBLING 1.53842
Primary Type=HOMICIDE 0.40948
Primary Type=HUMAN TRAFFICKING 1.12893
Primary Type=INTERFERENCE WITH PUBLIC OFFICER 1.12893
Primary Type=KIDNAPPING 1.53842
Primary Type=LIQUOR LAW VIOLATION 1.12893
Primary Type=MOTOR VEHICLE THEFT 0.40948
Primary Type=NARCOTICS 0.40948
Primary Type=NON - CRIMINAL 1.53842
Primary Type=NON-CRIMINAL 0.40948
Primary Type=NON-CRIMINAL (SUBJECT SPECIFIED) 1.12893
Primary Type=OBSCENITY 1.12893
Primary Type=OFFENSE INVOLVING CHILDREN 0.40948
Primary Type=OTHER NARCOTIC VIOLATION 1.53842
Primary Type=OTHER OFFENSE 1.12893
Primary Type=PROSTITUTION 1.53842
```

Conclusion and Future Work:

Big Data Analytics refers to the tools and practices that can be used for transforming this raw data into meaningful and crucial information which helps in forming a decision support system for the judiciary and legislature to take steps towards keeping crimes in check. With the ever-increasing population and crime rates, certain trends must be discovered, studied and discussed to take well informed decisions so that law and order can be maintained properly. If the number of complaints from a particular state is found to be very high, extra security must be provided to the residents there by increasing police presence, quick redressal of complaints and strict vigilance. Crimes against women are becoming an increasingly worrying and disturbing problem for the government. The number of such crimes must be found. Extra security must be provided so that law and order can be maintained properly and there is a sense of safety and well-being among the citizens of the country.

Code Documentation :

Crime Count Year wise:

Mapper :

```
import java.io.IOException;

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;
```

```

public class CrimeCountPerYearMapper extends Mapper<LongWritable, Text, Text, IntWritable> {
    public void map(LongWritable key, Text value, Context context) throws IOException, InterruptedException{
        String input = value.toString();
        String[] words = input.split("\t");
        IntWritable one = new IntWritable(1);

        if (!words[0].equals("ID")) {
            String year = words[17];
            if (year.matches("^\\d{4}$")) {
                context.write(new Text(year), one);
            }
        }
    }
}

```

Reducer:

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

import java.io.IOException;

public class CrimeCountPerYearReducer extends Reducer<Text, IntWritable, Text, IntWritable> {
    @Override
    protected void reduce(Text key, Iterable<IntWritable> values, Context context) throws IOException,
    InterruptedException {
        int count = 0;
        for (IntWritable val : values) {
            count += val.get();
        }
        context.write(key, new IntWritable(count));
    }
}

```

Driver:

```

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.FileSystem;
import org.apache.hadoop.fs.Path;

```

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

import java.io.IOException;

public class CrimeCountPerYearDriver {
    public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException {
        Configuration conf = new Configuration();

        FileSystem fs = FileSystem.get(conf);

        if (fs.exists(new Path(args[1]))) {
            fs.delete(new Path(args[1]), true);
        }

        Job job = Job.getInstance(conf);
        job.setMapperClass(CrimeCountPerYearMapper.class);
        job.setReducerClass(CrimeCountPerYearReducer.class);

        job.setJarByClass(CrimeCountPerYearDriver.class);

        job.setNumReduceTasks(3);

        TextInputFormat.addInputPath(job, new Path(args[0]));
        TextOutputFormat.setOutputPath(job, new Path(args[1]));

        job.setMapOutputKeyClass(Text.class);
        job.setMapOutputValueClass(IntWritable.class);

        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);

        System.exit(job.waitForCompletion(true) ? 0 : 1);
    }
}

```

Crime Count by Time of the Day:

Mapper

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;
import java.text.DateFormat;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;

public class CrimeCountByTimeOfDayMapper extends Mapper<LongWritable, Text, Text, IntWritable> {
    public void map(LongWritable key, Text value, org.apache.hadoop.mapreduce.Mapper.Context context) throws
IOException, InterruptedException {
        String input = value.toString();
        Text CrimeCountByTime = new Text();

        String[] words = input.split("\t");
        IntWritable one = new IntWritable(1);

        DateFormat df = new SimpleDateFormat("MM/dd/yyyy hh:mm:ss aa");
        SimpleDateFormat formatter = new SimpleDateFormat("dd/MM/yyyy HH:mm:ss");
        String inputDate = words[2];
        Date idate = null;
        String odate = null;
        Date outdate = null;
        if (inputDate.matches("^((\\d{1,2})/(\\d{2})\\d{4})\\s+(\\d+):(\\d{2})\\d+(AM|PM)$")) {
            try {
                idate = df.parse(inputDate);
            } catch (ParseException e) {
                e.printStackTrace();
            }
            try {
                odate = formatter.format(idate); // parsed the date in String type to Date Type
            } catch (Exception e) {
                System.out.println(e.getMessage());
            }
            try {
                outdate = formatter.parse(odate);
            } catch (ParseException e) {
                e.printStackTrace();
            }
        }

        int hours = outdate.getHours(); //Extract hours from date to categorize them into different ranges like (Early
Morning,Late Night,Late Morning)
    }
}

```

```

if (hours >= 5 && hours <= 8) {
    CrimeCountByTime.set("Early Morning");
}
if (hours >= 11 && hours <= 12) {
    CrimeCountByTime.set("Late Morning");
}
if (hours >= 13 && hours <= 15) {
    CrimeCountByTime.set("Early Afternoon");
}
if (hours >= 16 && hours <= 17) {
    CrimeCountByTime.set("Late Afternoon");
}
if (hours >= 17 && hours <= 21) {
    CrimeCountByTime.set("Early Evening");
}
if (hours >= 21 && hours <= 24) {
    CrimeCountByTime.set("Night");
}
if (hours >= 0 && hours <= 4) {
    CrimeCountByTime.set("Night");
}
context.write(CrimeCountByTime, one);
}
}
}

```

Reducer:

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

import java.io.IOException;

public class CrimeCountByTimeOfDayReducer extends Reducer<Text, IntWritable, Text, IntWritable> {
    protected void reduce(Text key, Iterable<IntWritable> values, Context context) throws
IOException, InterruptedException
    {
        int count=0;
        for(IntWritable val: values){
            count+=val.get();
        }
        context.write(key, new IntWritable(count));
    }
}

```

```
}
```

Driver :

```
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.FileSystem;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

import java.io.IOException;

public class CrimeCountByTimeOfDayDriver {
    public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException {
        Configuration conf = new Configuration();

        FileSystem fs = FileSystem.get(conf);

        if(fs.exists(new Path(args[1])))
        {
            fs.delete(new Path(args[1]),true);
        }

        Job job = Job.getInstance(conf);
        job.setMapperClass(CrimeCountByTimeOfDayMapper.class);
        job.setReducerClass(CrimeCountByTimeOfDayReducer.class);

        job.setJarByClass(CrimeCountByTimeOfDayDriver.class);

        job.setNumReduceTasks(1);

        TextInputFormat.addInputPath(job,new Path(args[0]));
        TextOutputFormat.setOutputPath(job,new Path(args[1]));

        job.setMapOutputKeyClass(Text.class);
        job.setMapOutputValueClass(IntWritable.class);

        job.setOutputKeyClass(Text.class);
        job.setOutputValueClass(IntWritable.class);

        System.exit(job.waitForCompletion(true) ? 0 : 1);
    }
}
```

```
    }
}
```

Crime Count by District:

Mapper:

```
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;

public class CrimeCountPerDistrictMapper extends Mapper<LongWritable, Text, Text, LongWritable> {
    public void map(LongWritable key, Text value, org.apache.hadoop.mapreduce.Mapper.Context context) throws
IOException, InterruptedException {
        String input = value.toString();
        String[] words = input.split("\t");
        LongWritable one = new LongWritable(1);

        if (!words[0].equals("ID")) {
            String district = words[1];
            if (district.matches("^\\d{11}$")) {
                try {
                    if (Integer.parseInt(district) <= 33 && district.length() > 0) {
                        String formatted = String.format("%03d", Integer.parseInt(district));
                        context.write(new Text(formatted), one);
                    }
                } catch (Exception e) {
                }
            }
        }
    }
}
```

Reducer:

```
import org.apache.hadoop.io.IntWritable;
```

```

import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

import java.io.IOException;

public class CrimeCountPerDistrictReducer extends Reducer<Text, LongWritable, Text, LongWritable> {

    protected void reduce(Text key, Iterable<LongWritable> values, Context context) throws
IOException, InterruptedException
    {
        int count=0;
        for(LongWritable val: values){
            count+=val.get();
        }
        context.write(key, new LongWritable(count));
    }
}

```

Driver:

```

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.FileSystem;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;
import org.apache.hadoop.io.Text;

import java.io.IOException;

public class CrimeCountPerDistrictDriver {
    public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException {
        Configuration conf = new Configuration();

        FileSystem fs = FileSystem.get(conf);

        if(fs.exists(new Path(args[1])))
        {
            fs.delete(new Path(args[1]),true);
        }
    }
}

```

```

Job job = Job.getInstance(conf);
job.setMapperClass(CrimeCountPerDistrictMapper.class);
job.setReducerClass(CrimeCountPerDistrictReducer.class);

job.setJarByClass(CrimeCountPerDistrictDriver.class);

job.setNumReduceTasks(1);

TextInputFormat.addInputPath(job,new Path(args[0]));
TextOutputFormat.setOutputPath(job,new Path(args[1]));

job.setMapOutputKeyClass(Text.class);
job.setMapOutputValueClass(LongWritable.class);

job.setOutputKeyClass(Text.class);
job.setOutputValueClass(LongWritable.class);

System.exit(job.waitForCompletion(true) ? 0 : 1);

}

}

```

Crime Type Count by District wise:

Mapper:

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;

public class CrimeTypeCountPerDistrictMapper extends Mapper<LongWritable, Text,Text, IntWritable> {
    public void map(LongWritable key, Text value, Context context) throws IOException,InterruptedException{
        String input = value.toString();
        String[] words = input.split("\t");
        IntWritable one = new IntWritable(1);

        if (!words[0].equals("ID")) {
            String district = words[1];
            if (district.matches("[0-9]*$")) {
                context.write(new Text(words[5]+"-"+words[1]), one);
            }
        }
    }
}

```

```
        }
    }
}
```

Reducer:

```
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

import java.io.IOException;

public class CrimeTypeCountPerDistrictReducer extends Reducer<Text, IntWritable, Text, IntWritable> {
    protected void reduce(Text key, Iterable<IntWritable> values, Context context) throws
IOException, InterruptedException
    {
        int count=0;
        for(IntWritable val: values){
            count+=val.get();
        }
        context.write(key, new IntWritable(count));
    }
}
```

Driver:

```
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.FileSystem;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

import java.io.IOException;

public class CrimeTypeCountPerDistrictDriver {
    public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException {
        Configuration conf = new Configuration();
```

```

FileSystem fs = FileSystem.get(conf);

if(fs.exists(new Path(args[1])))
{
    fs.delete(new Path(args[1]),true);
}

Job job = Job.getInstance(conf);
job.setMapperClass(CrimeTypeCountPerDistrictMapper.class);
job.setReducerClass(CrimeTypeCountPerDistrictReducer.class);

job.setJarByClass(CrimeTypeCountPerDistrictDriver.class);

job.setNumReduceTasks(3);

TextInputFormat.addInputPath(job,new Path(args[0]));
TextOutputFormat.setOutputPath(job,new Path(args[1]));

job.setMapOutputKeyClass(Text.class);
job.setMapOutputValueClass(IntWritable.class);

job.setOutputKeyClass(Text.class);
job.setOutputValueClass(IntWritable.class);

System.exit(job.waitForCompletion(true) ? 0 : 1);

}
}

```

Crime Count Day wise:

Mapper:

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;

public class CrimeCountPerDayMapper extends Mapper<LongWritable, Text, Text, IntWritable> {

```

```

public void map(LongWritable key, Text value, org.apache.hadoop.mapreduce.Mapper.Context context) throws
IOException, InterruptedException {
    String input = value.toString();

    String[] words = input.split("\t");
    IntWritable one = new IntWritable(1);

    if(words[2] != null && words[2].matches("^\\d{1,2}\\\\d{1,2}\\\\d{4} \\d{1,2}:\\d{1,2}:\\d{1,2} [AP]M\\z")) {
        SimpleDateFormat formatter = new SimpleDateFormat("MM/dd/yyyy h:mm:ss a");
        SimpleDateFormat f2 = new SimpleDateFormat("yy/MM/dd");

        String date = words[2];
        // Date dt=new Date();

        Date da = null; // parsed the date in String type to Date Type
        try {
            da = formatter.parse(date);
        } catch (ParseException e) {
            e.printStackTrace();
        }

        String dat = f2.format(da);

        context.write(new Text(dat), one);
    }
}
}
}

```

Reducer:

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

import java.io.IOException;
public class CrimeCountPerDayReducer extends Reducer<Text, IntWritable, Text, IntWritable> {
    protected void reduce(Text key, Iterable<IntWritable> values, Context context) throws
IOException, InterruptedException
    {
        int count=0;

```

```

        for(IntWritable val: values){
            count+=val.get();
        }
        context.write(key, new IntWritable(count));
    }
}

```

Moving Average Mapper:

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;

public class MovingAverageCCPDMapper extends Mapper<LongWritable, Text, Text, LongWritable> {
    public void map(LongWritable key, Text value, org.apache.hadoop.mapreduce.Mapper.Context context) throws
IOException, InterruptedException {
        String input = value.toString();
        String[] words = input.split("\t");
        LongWritable one = new LongWritable(Long.parseLong(words[1]));

        context.write(new Text(words[0]),one);
    }
}

```

Moving Average Reducer:

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

import java.io.IOException;

public class MovingAverageCCPDReducer extends Reducer<Text, LongWritable, Text, LongWritable> {
    private int windowSize = 4;
    long sum = 0;
    long n = 0;

    protected void reduce(Text key, Iterable<LongWritable> values, Context context) throws IOException,

```

```

InterruptedException {
    for (LongWritable value : values) {
        if(n%30==0){
            sum=0;
            n=0;
        }
        sum += value.get();
        n++;
    }

    long movingAverage = sum / n;
    context.write(key, new LongWritable(movingAverage));
}
}

```

Top Crime Days Mapper:

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;
import java.util.Map;
import java.util.TreeMap;

public class TopCrimeDaysMapper extends Mapper<LongWritable, Text, Text, IntWritable> {
    private TreeMap<Integer, Text> top100;
    private int N=100;

    @Override
    protected void setup(Mapper.Context context) throws IOException, InterruptedException {
        top100 = new TreeMap<Integer, Text>();
    }

    @Override
    protected void map(LongWritable key, Text value, Mapper.Context context) throws IOException, InterruptedException {
        String input = value.toString();
        String[] tokens = input.split("\t");

        String DateField = tokens[0];
        top100.put(Integer.parseInt(tokens[1]),new Text(DateField));

        if (top100.size()>N){
            top100.remove(top100.firstKey());
        }
    }
}

```

```

        }

    }

    @Override
    protected void cleanup(Mapper.Context context) throws IOException, InterruptedException {
        for (Map.Entry<Integer, Text> entry : top100.entrySet()){
            Text d = entry.getValue();
            Integer count = entry.getKey();

            context.write(new Text(d), new IntWritable(count));

        }
    }
}

```

Top Crime Days Reducer:

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;

import java.io.IOException;
import java.util.Map;
import java.util.TreeMap;

public class TopCrimeDaysReducer extends Reducer<Text, IntWritable, Text, IntWritable>{
    private TreeMap<Integer, Text> tmap2;

    @Override
    protected void setup(Reducer.Context context) throws IOException, InterruptedException {
        tmap2 = new TreeMap<Integer,Text>();
    }

    @Override
    protected void reduce(Text key, Iterable<IntWritable> values, Context context) throws IOException,
    InterruptedException {
        String name = key.toString();
        int count = 0;

        for (IntWritable val : values)
        {
            count = val.get();
        }
    }
}

```

```

        tmap2.put(count,new Text(name));

        if (tmap2.size()>100)
        {
            tmap2.remove(tmap2.firstKey());
        }

    }

@Override
protected void cleanup(Reducer.Context context) throws IOException, InterruptedException {
    for (Map.Entry<Integer, Text> entry : tmap2.entrySet())
    {

        Text count = entry.getValue();
        Integer name = entry.getKey();
        context.write(new Text(count), new IntWritable(name));
    }
}
}

```

Driver for Three Jobs Together:

```

import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.FileSystem;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

import java.io.IOException;

public class CrimeCountPerDayDriver {
    public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException {
        Configuration conf = new Configuration();

        FileSystem fs = FileSystem.get(conf);

        if(fs.exists(new Path(args[1])))
        {
            fs.delete(new Path(args[1]),true);
        }
    }
}

```

```

Job job1 = Job.getInstance(conf,"CrimeCountPerDay");
job1.setJarByClass(CrimeCountPerDayDriver.class);
job1.setMapperClass(CrimeCountPerDayMapper.class);
job1.setReducerClass(CrimeCountPerDayReducer.class);
job1.setOutputKeyClass(Text.class);
job1.setOutputValueClass(IntWritable.class);
job1.setMapOutputKeyClass(Text.class);
job1.setMapOutputValueClass(IntWritable.class);

TextInputFormat.addInputPath(job1,new Path(args[0]));
TextOutputFormat.setOutputPath(job1,new Path(args[1]));
job1.setNumReduceTasks(1);
job1.waitForCompletion(true);

```

// Job2

```

Configuration conf2 = new Configuration();

FileSystem fs2 = FileSystem.get(conf2);
if(fs2.exists(new Path(args[2])))
{
    fs2.delete(new Path(args[2]),true);
}

Job job2 = Job.getInstance(conf2,"MovingAverage");

job2.setJarByClass(CrimeCountPerDayDriver.class);
job2.setMapperClass(MovingAverageCCPDMapper.class);
job2.setReducerClass(MovingAverageCCPDReducer.class);
job2.setOutputKeyClass(Text.class);
job2.setOutputValueClass(LongWritable.class);
job2.setMapOutputKeyClass(Text.class);
job2.setMapOutputValueClass(LongWritable.class);

TextInputFormat.addInputPath(job2,new Path(args[1]));
TextOutputFormat.setOutputPath(job2,new Path(args[2]));

```

```

job2.setNumReduceTasks(1);
job2.waitForCompletion(true);

```

// Job3

```

Configuration conf3 = new Configuration();

```

```

FileSystem fs3 = FileSystem.get(conf3);
if(fs2.exists(new Path(args[3])))
{
    fs2.delete(new Path(args[3]),true);
}

Job job3 = Job.getInstance(conf2,"Top100CrimeDays");

job3.setJarByClass(CrimeCountPerDayDriver.class);
job3.setMapperClass(TopCrimeDaysMapper.class);
job3.setReducerClass(TopCrimeDaysReducer.class);
job3.setOutputKeyClass(Text.class);
job3.setOutputValueClass(IntWritable.class);
job3.setMapOutputKeyClass(Text.class);
job3.setMapOutputValueClass(IntWritable.class);

TextInputFormat.addInputPath(job3,new Path(args[1]));
TextOutputFormat.setOutputPath(job3,new Path(args[3]));

job3.setNumReduceTasks(1);

System.exit(job3.waitForCompletion(true) ? 0 : 1);

}
}

```

Crime CaseID Sorting District wise:

Mapper :

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;

public class CrimeCountByDistrictByType extends Mapper<LongWritable, Text, CompositeKey, NullWritable> {
    public void map(LongWritable key, Text value, Context context) throws IOException, InterruptedException {
        String line = value.toString();
        String[] tokens = line.split("\t");
        String districtCode = null;
        String crimeType = null;
        // IntWritable one = new IntWritable(1);

        districtCode = tokens[11];
    }
}

```

```

        crimeType = tokens[1];

        if(districtCode!=null && crimeType!=null)
        {
            CompositeKey outKey = new CompositeKey(districtCode,crimeType);
            context.write(outKey,NullWritable.get());
        }

    }
}

```

Reducer

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;

public class CrimeCountByDistrictByType extends Mapper<LongWritable, Text, CompositeKey, NullWritable> {
    public void map(LongWritable key, Text value, Context context) throws IOException, InterruptedException {
        String line = value.toString();
        String[] tokens = line.split("\t");
        String districtCode = null;
        String crimeType = null;
        // IntWritable one = new IntWritable(1);

        districtCode = tokens[11];
        crimeType = tokens[1];

        if(districtCode!=null && crimeType!=null)
        {
            CompositeKey outKey = new CompositeKey(districtCode,crimeType);
            context.write(outKey,NullWritable.get());
        }

    }
}

```

Composite Key :

```
import org.apache.hadoop.io.WritableComparable;

import java.io.DataInput;
import java.io.DataOutput;
import java.io.IOException;

public class CompositeKey implements WritableComparable<CompositeKey> {

    private String crimeType;
    private String districtCode;

    public CompositeKey(){}
        super();

    }

    public CompositeKey(String districtCode, String crimeType){
        super();
        this.districtCode=districtCode;
        this.crimeType=crimeType;

    }

    public String getCrimeType() {
        return crimeType;
    }

    public void setCrimeType(String crimeType) {
        this.crimeType = crimeType;
    }

    public String getDistrictCode() {
        return districtCode;
    }

    public void setDistrictCode(String districtCode) {
        this.districtCode = districtCode;
    }

    public int compareTo(CompositeKey o){
        //comparing if district code equals check for crimeType // -1,0,1
        int result = this.districtCode.compareTo(o.districtCode);
        if (result==0){
            return this.crimeType.compareTo(o.crimeType);
        }
        return result;
    }
}
```

```

}

public void write(DataOutput out) throws IOException{
    out.writeUTF(districtCode);
    out.writeUTF(crimeType);

}

public void readFields(DataInput in) throws IOException{
    districtCode=in.readUTF();
    crimeType=in.readUTF();

}

@Override
public String toString(){
    return districtCode + "," +crimeType;
}
}
}

```

Key Partitioner :

```

import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.mapreduce.Partitioner;

public class KeyPartitioner extends Partitioner<CompositeKey, IntWritable> {
    @Override
    public int getPartition(CompositeKey key, IntWritable value, int numPartitions){
        return key.getDistrictCode().hashCode()%numPartitions;
    }
}

```

Sec Sort Comparator:

```

import org.apache.hadoop.io.WritableComparable;
import org.apache.hadoop.io.WritableComparator;

public class SecSortComparator extends WritableComparator {
    public SecSortComparator(){
        super(CompositeKey.class,true);
    }

    public int compare(WritableComparable a,WritableComparable b)
    {
        CompositeKey ck1 = (CompositeKey) a;
        CompositeKey ck2 = (CompositeKey) b;

```

```

int result = ck1.getDistrictCode().compareTo(ck2.getDistrictCode());

if(result==0){
    return -1*ck1.getCrimeType().compareTo(ck2.getCrimeType());
}

return result;
}
}

```

Sec Sort Group Comparator:

```

import org.apache.hadoop.io.WritableComparable;
import org.apache.hadoop.io.WritableComparator;

public class SecSortGroupComparator extends WritableComparator {
    public SecSortGroupComparator() {
        super(CompositeKey.class,true);
    }

    @Override
    public int compare(WritableComparable a, WritableComparable b) {
        CompositeKey ck1 = (CompositeKey)a;
        CompositeKey ck2 = (CompositeKey)b;

        return ck1.getDistrictCode().compareTo(ck2.getCrimeType());
    }
}

```

Driver

```

import org.apache.hadoop.fs.FileSystem;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.NullWritable;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.lib.input.TextInputFormat;
import org.apache.hadoop.mapreduce.lib.output.TextOutputFormat;

import java.io.IOException;

public class CrimeCountByDistrictByTypeDriver {
    public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException {
        Job job = Job.getInstance();

        job.setJarByClass(CrimeCountByDistrictByTypeDriver.class);
    }
}

```

```

job.setGroupingComparatorClass(SecSortGroupComparator.class);
job.setSortComparatorClass(SecSortComparator.class);
job.setPartitionerClass(KeyPartitioner.class);

TextInputFormat.addInputPath(job, new Path(args[0]));
Path outdir = new Path(args[1]);
TextOutputFormat.setOutputPath(job, outdir);

job.setMapperClass(CrimeCountByDistrictByType.class);
job.setReducerClass(CrimeCountByDistrictByTypeReducer.class);

job.setNumReduceTasks(1);

job.setOutputKeyClass(CompositeKey.class);
job.setOutputValueClass(NullWritable.class);

FileSystem fs = FileSystem.get(job.getConfiguration());
if(fs.exists(outdir))
{
    fs.delete(outdir,true);
}

System.exit(job.waitForCompletion(true) ? 0 : 1);

}
}

```

Mahout Analysis:

Code for converting TSV to CSV:

```

import java.io.BufferedReader;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.StringTokenizer;

/**
 * @author Kushal Paudyal
 * www.icodejava.com
 * Convert Tab Separate Value (TSV) file to Comma Separated Value (CSV) file
 */
public class Convertor {
    public static void main(String[] args) throws Exception {

```

```

//The input TSV File
String tsvFilePath = "/Users/ikyathvarmadantuluri/Downloads/Crimes_-_2001_to_Present.tsv";

//The output CSV File
String csvFilePath = "/Users/ikyathvarmadantuluri/Downloads/Crimes_-_2001_to_Present.csv";

convertTSVToCSVFile(csvFilePath, tsvFilePath);

}

/**
 * Converts a TSV file into CSV file.
 * - Reads one line at a time
 * - Replaces Double Quotes with Single Quotes
 * - Puts Double Quotes Around Every Field
 * - Fileds in the output are separated by comma.
 *
 * @param csvFilePath
 * @param tsvFilePath
 * @throws IOException
 */
private static void convertTSVToCSVFile(String csvFilePath, String tsvFilePath) throws IOException {

    //TODO: If outfile does not exist, create one.

    StringTokenizer tokenizer;
    try (BufferedReader br = new BufferedReader(new FileReader(tsvFilePath));
         PrintWriter writer = new PrintWriter(new FileWriter(csvFilePath))) {

        int i = 0;
        for (String line; (line = br.readLine()) != null; ) {
            i++;
            if (i % 10000 == 0) {
                System.out.println("Processed: " + i);
            }

            tokenizer = new StringTokenizer(line, "\t");

            String csvLine = "";
            String token;
            while (tokenizer.hasMoreTokens()) {
                token = tokenizer.nextToken().replaceAll(",", " ");
                csvLine += token + ",";
            }

            if (csvLine.endsWith(",")) {

```

```

        csvLine = csvLine.substring(0, csvLine.length() - 1);
    }

    writer.write(csvLine + System.getProperty("line.separator"));

}

}

private static String convertToCSV(String line) {
    String csv = "";
    line = line.replaceAll("\t", ",");
    return line;
}
}

```

Code to find regression equation for dependent variable based on independent variables:

```

import java.io.BufferedReader;
import java.io.FileOutputStream;
import java.io.FileReader;
import java.io.IOException;
import java.io.OutputStream;
import java.io.OutputStreamWriter;
import java.io.PrintWriter;
import java.util.Arrays;
import java.util.List;
import java.util.Locale;

//import
import org.apache.mahout.classifier.sgd.CsvRecordFactory;
import org.apache.mahout.classifier.sgd.LogisticModelParameters;
import org.apache.mahout.classifier.sgd.OnlineLogisticRegression;
import org.apache.mahout.classifier.sgd.RecordFactory;
import org.apache.mahout.math.RandomAccessSparseVector;
import org.apache.mahout.math.Vector;

import com.google.common.base.Charsets;

public class Regression {
// Extends the basic on-line logistic regression learner with a specific set of learning rate annealing schedules.
    private static double predictorWeight(OnlineLogisticRegression lr, int row, RecordFactory csv, String predictor) {

```

```

    double weight = 0;
    //Converts CSV data lines to vectors
    for (Integer column : csv.getTraceDictionary().get(predictor)) {
        weight += lr.getBeta().get(row, column);
    }
    return weight;
}

public static void main(String[] args) throws IOException
{
    String inputFile = "/Users/ikyathvarmadantuluri/Downloads/Crimes_-_2001_to_Present.csv";
    String outputFile = "/Users/ikyathvarmadantuluri/model";

    //      List<String> predictorList =Arrays.asList("ID","Case Number","Date","Block","IUCR","Primary
    //      Type","Description","Location Description","Arrest","Domestic","Beat","District","Ward","Community Area","FBI
    //      Code","X Coordinate","Y Coordinate","Year","Updated On","Latitude","Longitude","Location");
    //      List<String> typeList = Arrays.asList("n", "w", "w", "w", "w", "w", "w", "w", "w", "w", "n", "n", "n", "n",
    //      "n", "w", "w", "w", "n", "w", "n", "n", "w");
    List<String> predictorList =Arrays.asList("Primary Type");
    //      "Domestic","Beat","District","Ward","Community Area","FBI Code","Year");
    List<String> typeList = Arrays.asList("w");
    //      "w", "n", "n", "n",
    //      "w","w","n");

    //      List<String> predictorList =Arrays.asList("Year","Month","Day");
    //      List<String> typeList = Arrays.asList("n","n","n");

    LogisticModelParameters Imp = new LogisticModelParameters();
    Imp.setTargetVariable("Arrest");
    Imp.setMaxTargetCategories(2);
    Imp.setNumFeatures(2);
    Imp.setUseBias(false);
    Imp.setTypeMap(predictorList,typeList);
    Imp.setLearningRate(0.5);

    int passes = 5;
    OnlineLogisticRegression lr;

    CsvRecordFactory csv = Imp.getCsvRecordFactory();
    lr = Imp.createRegression();

    int k = 0;

    for (int pass = 0; pass < passes; pass++) {
        BufferedReader in = new BufferedReader(new FileReader(inputFile));

```

```

csv.firstLine(in.readLine());

String line = in.readLine();
int lineCount = 2;
while (line != null) {

    Vector input = new RandomAccessSparseVector(lImp.getNumFeatures());
//    if(input.)
    int targetValue = csv.processLine(line, input);

    // update model
    lr.train(targetValue, input);
    System.out.println(lineCount);
    k++;

    line = in.readLine();
    lineCount++;
}
in.close();
}

/*      best = model.getBest();
if (best != null) {
    learner = best.getPayload().getLearner();
}*/

```



```

OutputStream modelOutput = new FileOutputStream(outputFile);
try {
    lImp.saveTo(modelOutput);
} finally {
    modelOutput.close();
}
PrintWriter output=new PrintWriter(new OutputStreamWriter(System.out,Charsets.UTF_8),true);
output.println(lImp.getNumFeatures());
output.println(lImp.getTargetVariable() + " ~ ");
String sep ="";
for (String v : csv.getTraceDictionary().keySet()) {
    double weight = predictorWeight(lr, 0, csv, v);
    if (weight != 0) {
        output.printf(Locale.ENGLISH, "%s%.3f%s", sep, weight, v);
        sep = " + ";
    }
}
output.printf("%n");

```

```
for (int row = 0; row < lr.getBeta().numRows(); row++) {
    for (String key : csv.getTraceDictionary().keySet()) {
        double weight = predictorWeight(lr, row, csv, key);
        if (weight != 0) {
            output.printf(Locale.ENGLISH, "%20s %.5f%n", key, weight);
        }
    }
    for (int column = 0; column < lr.getBeta().numCols(); column++) {
        output.printf(Locale.ENGLISH, "%15.9f ", lr.getBeta().get(row, column));
    }
    output.println();
}
}
```