# Features

## Segment attributes

Data source: [street segment](http://opendata.dc.gov/datasets/street-segments)

Features:

1. Float: SHAPE\_Length
2. Binary:
   1. DIRECTIONALITY\_Bi-direction, DIRECTIONALITY\_one-way
   2. STREETTYPE\_AVE, STREETTYPE\_BLVD, STREETTYPE\_CIR, STREETTYPE\_CRES, STREETTYPE\_CT, STREETTYPE\_DR, STREETTYPE\_LN, STREETTYPE\_OTHER, STREETTYPE\_PKWY, STREETTYPE\_PL, STREETTYPE\_RD, STREETTYPE\_ST, STREETTYPE\_TER, STREETTYPE\_WAY
   3. SEGMENTTYPE\_1, SEGMENTTYPE\_2, SEGMENTTYPE\_3

## Segment Network related features

Four network is constructed:

[Segment as node(SgAsNd) vs. segment as edge(SgAsEg)] \* [Directed(d) vs. undirected(ud)]

Features:

1. Float: d\_btw\_cntr\_SgAsEg, ud\_btw\_cntr\_SgAsEg, d\_auth\_score\_SgAsNd, d\_btw\_cntr\_SgAsNd, d\_clo\_cntr\_SgAsNd, d\_far\_cntr\_SgAsNd, d\_hub\_score\_SgAsNd, d\_in\_deg\_SgAsNd, d\_node\_ecc\_SgAsNd, d\_out\_deg\_SgAsNd, d\_page\_rank\_SgAsNd, ud\_auth\_score\_SgAsNd, ud\_btw\_cntr\_SgAsNd, ud\_clo\_cntr\_SgAsNd, ud\_deg\_cntr\_SgAsNd, ud\_eig\_cntr\_SgAsNd, ud\_far\_cntr\_SgAsNd, ud\_hub\_score\_SgAsNd, ud\_node\_ecc\_SgAsNd, ud\_page\_rank\_SgAsNd
2. Binary: ud\_bridge\_SgAsEg, ud\_bridge\_SgAsNd, ud\_art\_pt\_SgAsNd(articulation points)

## Car related: moving/parking/crashes

Car related data are all timestamped and count-based features.

The coordinate of each event is spatially joint with segments. The distance of intersection join is 5 meters and the distance of closest join is 20 meters. Intersection join means if there are multiple segments within 5 meters of the coordinate, then this event will be counted once for each segment (the event is considered to happen in the intersection of these segments). If there isn’t any segment within 5 meters, closest join will be applied to find the closest segment within 20 meters. If there isn’t any segment within 20 meters, this event is not counted for any segment.

NA is filled with the mean count values of the other segments with the same STREETTYPE.

### Moving Violation

Data source: [monthly mov violation](http://opendata.dc.gov/datasets/moving-violations-issued-in-february-2016)

Features, such as:

DISTRACTED DRIVING USING CELL PHONE, OTHER DEVICE, FAIL TO DISPLAY PROOF OF VEHICLE INSURANCE, FAIL TO PAY ATTENTION WHILE OPERATING A VEHICLE, FAIL TO STOP PER REGULATIONS FACING RED SIGNAL, FAIL TO YIELD RIGHT OF WAY, OWNER OPERATE OR PERMIT OPERATION OF UNINSRD VEH, PASSING STOP SIGN WITHOUT COMING TO A FULL STOP, SEAT BELT REGULATION VIOLATION…

### Parking Violation

Data source: [monthly park violation](http://opendata.dc.gov/datasets/parking-violations-issued-in-march-2016)

Features, such as:

NO PARKING STREET CLEANING, PARK WITH LEFT WHEEL TO THE CURB, PARKED IN DRIVEWAY OR ALLEY TO OBSTRUCT SIDEWALK, PARKED WITHIN 25 FEET OF A STOP SIGN, RELOCATE TOW FEE, RESIDENTIAL PERMIT PKING BEYOND LIMIT W/O PERMIT…

### Crashes

Data source: [crashes](http://opendata.dc.gov/datasets/crashes-in-the-district-of-columbia)

Features, such as:

crash\_evt\_COLLISION\_WITH\_FIXED, crash\_evt\_NON\_COLLISION, crash\_evt\_PENDING\_INVESTIGATION, crash\_1stharm\_Hit and Run…

## Neighborhood: crime/vision zero/311

Neighborhood related features are timestamped and count-based features.

The distance of intersection join and closest join are 5 and 20 meters respectively for vision zero and 311. In order to account for the larger influence range of crime incidents, the distance of intersection join is set as 150 meters. This means that a crime incident will be counted once for each segment within 150 meters.

NA is filled with 0.

### Crime

Data source: [yearly crime](http://opendata.dc.gov/datasets?q=crime%20incidents)

Features: crime\_mtd\_KNIFE, crime\_mtd\_OTHERS, crime\_ofn\_ASSAULT W/DANGEROUS WEAPON, crime\_ofn\_BURGLARY, crime\_ofn\_HOMICIDE, crime\_ofn\_MOTOR VEHICLE THEFT, crime\_ofn\_ROBBERY, crime\_ofn\_SEX ABUSE, crime\_ofn\_THEFT F/AUTO, crime\_ofn\_THEFT/OTHER…

### Vision zero

Data source: [vision zero](http://opendata.dc.gov/datasets/vision-zero-safety)

Features: v0ur\_Biker, v0ur\_Car Driver, v0ur\_Pedestrian, v0rq\_Accessibility Issue, v0rq\_Blocking the bikebox, v0rq\_Blocking the crosswalk, v0rq\_Cyclist behavior, v0rq\_Double parking, v0rq\_Failure to stop for pedestrians, v0rq\_Jaywalking, v0rq\_Long distance to cross, v0rq\_Long wait to cross…

### 311 request

Data source: [cityworks-service-requests](http://opendata.dc.gov/datasets/cityworks-service-requests)

Features: 311\_ALLEY REPAIR, 311\_ALLEYLIGHT REPAIR, 311\_ANC RESOLUTION, 311\_BICYCLE SERVICES, 311\_BICYCLES, 311\_BRIDGE MAINTENANCE R, 311\_BULB OUT, 311\_BUS AND/OR RAIL ISSU, 311\_BUS SHELTERS, 311\_CHILD SAFETY SEAT PR, 311\_CURB GUTTER REPAIR, 311\_DDOT CITATION, 311\_FLASHER MALFUNCTION, 311\_FLASHER MODIFICATION, 311\_LIGHT-INFRASTRUCTURE, 311\_LIGHT-LIGHT POLE, 311\_LIGHT-OHGS…

## Point of Interests: OSM/FS

Data source: [frsq venues search api](https://developer.foursquare.com/docs/venues/search), [OSM](http://www.openstreetmap.org/)

Manually built mapping between OSM and categories and mapping between FS taxonomy and categories.

Count-based features:

art, outdoors and recreation, retail shop, professional service, food, nightlife spot, residence, schools&university, cycling facilities, transportation.

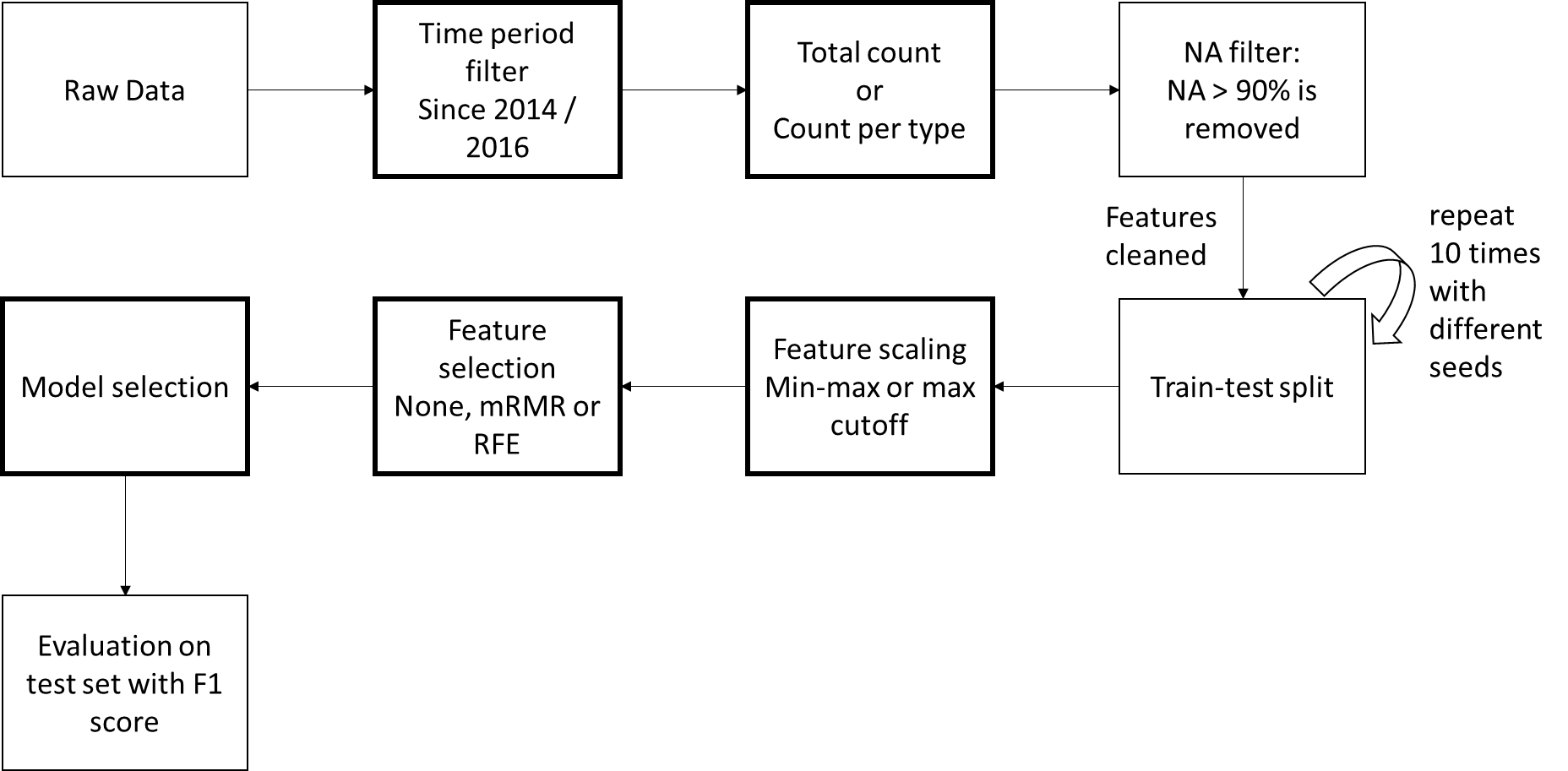
## Bike facilities: DC/OSM

[Street Right of Way for Bicycle](http://opendata.dc.gov/datasets/street-right-of-way-for-bicycle-lanes): (count-based features) Bus/Bike Lane, Climbing Lane, Contraflow Bike Lane, Cycle Track, Existing Bike Lane, Shared Lane

[OSM](http://www.openstreetmap.org/): (binary features) is\_shared, cycle\_lane\_both, cycle\_lane\_one, cycle\_lane\_right, cycle\_way\_both, cycle\_way\_one, cycle\_way\_right, side\_walk\_both, side\_walk\_left, side\_walk\_no, side\_walk\_right, bikable\_no, bikable\_yes

# Experiment

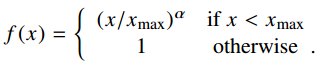
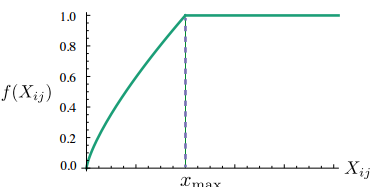
## Experiment pipeline



1. Time period: since 2014; since 2016. This affects car related and neighborhood related features
2. Total count or count per type. This affects count-based features (car related, neighborhood related features and POI)
3. Feature scaling: min-max scaling; max-cutoff

Min-max scaling: x\_scaled = (x-min)/(max-min);

Max-cutoff: This is inspired by GloVe[[1]](#footnote-1) paper. This is applied to count-based features

1. Feature selection: None (no selection is applied), mRMR, rfecv\_linsvc (recursive feature elimination with linear SVC as estimator).
2. Model selection
   1. Models include:
      1. regressions: OLS, ridge, lasso;
      2. classification: logistics;
      3. both regression and classification: decision tree, random forest, adaboost, bagging, gradient boost, multi-layer perception, SVM, linear SVM.
   2. Parameters of each model is tuned with grid search + cross validation on training set.
      1. Ols: None.
      2. Ridge and lasso: {'alpha': np.logspace(0, 2, 10)}
      3. Logistics: {'C': np.logspace(-4, 2, 4), 'penalty': ['l1', 'l2']}
      4. Decision Tree: {'max\_depth': [3, 5, 10, 30, 50], 'max\_features': [0.1, 0.3, 0.5, 1.]}
      5. Random Forest: {'n\_estimators': [10, 100, 500], 'max\_features': ['sqrt', 'log2'], 'min\_samples\_leaf': [1, 2]}
      6. Adaboost: {'n\_estimators': [10, 30, 50, 100, 256, 500], 'learning\_rate': np.logspace(-4, 1, 5)}
      7. Bagging: {'n\_estimators': [10, 30, 50, 100, 256, 500], 'max\_features': [0.4, 0.7, 1.0]}
      8. Gradient boost: {'n\_estimators': [10, 50, 100], 'max\_features': ['sqrt', 'log2'], 'learning\_rate': np.logspace(-4, 1, 3), 'max\_depth': [3, 10, 50]}
      9. Multi-layer perception: {'hidden\_layer\_sizes': [(100,), (5, 2), (20, 5), (100, 20), (100, 20, 5)], 'learning\_rate': ['constant', 'adaptive'] }
      10. SVM/SVR: {‘kernel’: ['rbf', 'sigmoid', 'poly'], 'c': np.logspace(-4, 2, 3), 'gamma': [1e-5, 1e-3, 1e-1]}
      11. Linear SVM: {'C': np.logspace(-4, 2, 3), 'loss': ['hinge', 'squared\_hinge']}
      12. Linear SVR: {'C': np.logspace(-4, 2, 3), 'loss': ['epsilon\_insensitive', 'squared\_epsilon\_insensitive'], 'epsilon': [0, 0.1, 1]}

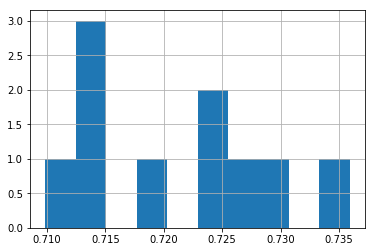
## Results

In sum:

1. Using recent (since 2016) car related and neighborhood related features has a slightly better chance to perform better than longer period (since 2014).
2. For count-based feature, count per type is always better than total count.
3. Max cutoff has a slightly better chance to perform better than min-max.
4. In most cases, no feature selection is better.
5. Gradient boosting classification is the best among all models. Train errors of tuned gradient boosting in all 10 runs are 0.
6. Best test f1 score range between 0.71 and 0.73.

### Parameters of the best test f1 score of each run:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| seed | time period | Total or by type | data scaling | Feature selection | # features | model | test\_f1 |
| 0 | since 2014 | by\_type | max\_cut | rfecv\_linsvc | 194 | GDBcls | 0.723005 |
| 100 | since 2016 | by\_type | max\_cut | None | 178 | GDBcls | 0.72797 |
| 972 | since 2016 | by\_type | min-max | rfecv\_linsvc | 137 | GDBcls | 0.735932 |
| 5258 | since 2014 | by\_type | max\_cut | None | 230 | GDBcls | 0.724184 |
| 7821 | since 2016 | by\_type | max\_cut | None | 178 | GDBcls | 0.719101 |
| 40918 | since 2016 | by\_type | max\_cut | None | 178 | GDBcls | 0.714291 |
| 57852 | since 2016 | by\_type | min-max | rfecv\_linsvc | 140 | GDBcls | 0.709868 |
| 168352 | since 2014 | by\_type | min-max | None | 230 | GDBcls | 0.712624 |
| 291592 | since 2014 | by\_type | max\_cut | mrmr | 176 | GDBcls | 0.714525 |
| 789729423 | since 2016 | by\_type | max\_cut | None | 178 | GDBcls | 0.729038 |



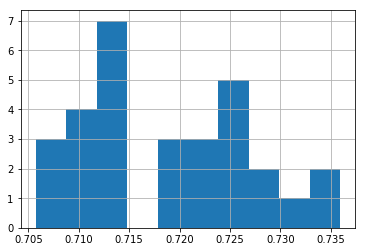
Distribution of test f1 score of 10 runs.

All 10 runs are using count by type and GDBcls (Gradient boosting classification).

The distribution of other 3 parameters are shown below

### Parameters of the best 3 test f1 scores of each run:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| seed | rank | total or by type | time period | data scaling | feature selection | keep | model | test\_f1 |
| 0 | 1 | by type | since 2014 | max\_cut | rfecv\_linsvc | 194 | GDBcls | 0.723005 |
| 0 | 2 | by type | since 2014 | min-max | None | 230 | GDBcls | 0.718645 |
| 0 | 3 | by type | since 2014 | max\_cut | None | 230 | GDBcls | 0.718372 |
| 100 | 1 | by type | since 2016 | max\_cut | None | 178 | GDBcls | 0.72797 |
| 100 | 2 | by type | since 2016 | min-max | None | 178 | GDBcls | 0.726508 |
| 100 | 3 | by type | since 2016 | max\_cut | rfecv\_linsvc | 147 | GDBcls | 0.724759 |
| 972 | 1 | by type | since 2016 | min-max | rfecv\_linsvc | 137 | GDBcls | 0.735932 |
| 972 | 2 | by type | since 2014 | min-max | None | 230 | GDBcls | 0.734606 |
| 972 | 3 | by type | since 2016 | min-max | None | 178 | GDBcls | 0.731876 |
| 5258 | 1 | by type | since 2014 | max\_cut | None | 230 | GDBcls | 0.724184 |
| 5258 | 2 | by type | since 2014 | max\_cut | mrmr | 176 | GDBcls | 0.722683 |
| 5258 | 3 | by type | since 2014 | min-max | None | 230 | GDBcls | 0.721128 |
| 7821 | 1 | by type | since 2016 | max\_cut | None | 178 | GDBcls | 0.719101 |
| 7821 | 2 | by type | since 2016 | max\_cut | rfecv\_linsvc | 145 | GDBcls | 0.714755 |
| 7821 | 3 | by type | since 2016 | min-max | None | 178 | GDBcls | 0.713201 |
| 40918 | 1 | by type | since 2016 | max\_cut | None | 178 | GDBcls | 0.714291 |
| 40918 | 2 | by type | since 2016 | min-max | None | 178 | GDBcls | 0.710432 |
| 40918 | 3 | by type | since 2016 | max\_cut | rfecv\_linsvc | 139 | GDBcls | 0.706616 |
| 57852 | 1 | by type | since 2016 | min-max | rfecv\_linsvc | 140 | GDBcls | 0.709868 |
| 57852 | 2 | by type | since 2016 | max\_cut | rfecv\_linsvc | 131 | GDBcls | 0.708479 |
| 57852 | 3 | by type | since 2016 | max\_cut | None | 178 | GDBcls | 0.705741 |
| 168352 | 1 | by type | since 2014 | min-max | None | 230 | GDBcls | 0.712624 |
| 168352 | 2 | by type | since 2016 | min-max | rfecv\_linsvc | 135 | GDBcls | 0.711996 |
| 168352 | 3 | by type | since 2016 | max\_cut | None | 178 | GDBcls | 0.711532 |
| 291592 | 1 | by type | since 2014 | max\_cut | mrmr | 176 | GDBcls | 0.714525 |
| 291592 | 2 | by type | since 2014 | max\_cut | None | 230 | GDBcls | 0.71398 |
| 291592 | 3 | by type | since 2016 | max\_cut | mrmr | 150 | GDBcls | 0.710028 |
| 789729423 | 1 | by type | since 2016 | max\_cut | None | 178 | GDBcls | 0.729038 |
| 789729423 | 2 | by type | since 2016 | min-max | rfecv\_linsvc | 178 | GDBcls | 0.726843 |
| 789729423 | 3 | by type | since 2014 | max\_cut | None | 230 | GDBcls | 0.725844 |



Distribution of top 3 test f1 score of 10 runs.

All 10 runs are using count by type and GDBcls (Gradient boosting classification).

The distribution of other 3 parameters are shown below:

1. <https://nlp.stanford.edu/pubs/glove.pdf> page 4 [↑](#footnote-ref-1)