

Do the genders commit different violations?

ANALYZING POLICE ACTIVITY WITH PANDAS



Kevin Markham
Founder, Data School

Counting unique values (1)

- `.value_counts()` : Counts the unique values in a Series
- Best suited for categorical data

```
ri.stop_outcome.value_counts()
```

```
Citation          77091
Warning           5136
Arrest Driver     2735
No Action         624
N/D               607
Arrest Passenger  343
Name: stop_outcome, dtype: int64
```

Counting unique values (2)

```
ri.stop_outcome.value_counts().sum()
```

```
86536
```

```
ri.shape
```

```
(86536, 13)
```

Expressing counts as proportions

```
ri.stop_outcome.value_counts()
```

```
77091/86536
```

```
0.8908546731995932
```

Citation	77091
Warning	5136
Arrest Driver	2735
No Action	624
N/D	607
Arrest Passenger	343

```
ri.stop_outcome.value_counts(  
    normalize=True)
```

Citation	0.890855
Warning	0.059351
Arrest Driver	0.031605
No Action	0.007211
N/D	0.007014
Arrest Passenger	0.003964

Filtering DataFrame rows

```
ri.driver_race.value_counts()
```

White	61870
Black	12285
Hispanic	9727
Asian	2389
Other	265

```
white = ri[ri.driver_race == 'White']  
white.shape
```

```
(61870, 13)
```

Comparing stop outcomes for two groups

```
white.stop_outcome.value_counts(  
    normalize=True)
```

Citation	0.902263
Warning	0.057508
Arrest Driver	0.024018
No Action	0.007031
N/D	0.006433
Arrest Passenger	0.002748

```
asian = ri[ri.driver_race ==  
           'Asian']  
asian.stop_outcome.value_counts(  
    normalize=True)
```

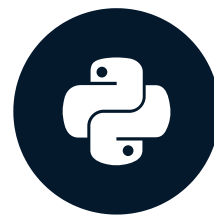
Citation	0.922980
Warning	0.045207
Arrest Driver	0.017581
No Action	0.008372
N/D	0.004186
Arrest Passenger	0.001674

Let's practice!

ANALYZING POLICE ACTIVITY WITH PANDAS

Does gender affect who gets a ticket for speeding?

ANALYZING POLICE ACTIVITY WITH PANDAS



Kevin Markham

Founder, Data School

Filtering by multiple conditions (1)

```
female = ri[ri.driver_gender == 'F']  
female.shape
```

```
(23774, 13)
```

Filtering by multiple conditions (2)

```
female_and_arrested = ri[(ri.driver_gender == 'F') &  
                          (ri.is_arrested == True)]
```

- Each condition is surrounded by parentheses
- Ampersand (&) represents the `and` operator

```
female_and_arrested.shape
```

```
(669, 13)
```

- Only includes female drivers who were arrested

Filtering by multiple conditions (3)

```
female_or_arrested = ri[(ri.driver_gender == 'F') |  
                        (ri.is_arrested == True)]
```

- Pipe (`|`) represents the `or` operator

```
female_or_arrested.shape
```

```
(26183, 13)
```

- Includes all females
- Includes all drivers who were arrested

Rules for filtering by multiple conditions

- Ampersand (`&`): only include rows that satisfy both conditions
- Pipe (`|`): include rows that satisfy either condition
- Each condition must be surrounded by parentheses
- Conditions can check for equality (`==`), inequality (`!=`), etc.
- Can use more than two conditions

Correlation, not causation

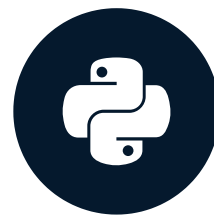
- Analyze the relationship between gender and stop outcome
 - Assess whether there is a correlation
- Not going to draw any conclusions about causation
 - Would need additional data and expertise
 - Exploring relationships only

Let's practice!

ANALYZING POLICE ACTIVITY WITH PANDAS

Does gender affect whose vehicle is searched?

ANALYZING POLICE ACTIVITY WITH PANDAS



Kevin Markham
Founder, Data School

Math with Boolean values

```
ri.isnull().sum()
```

```
stop_date      0
stop_time      0
driver_gender   0
driver_race     0
violation_raw   0
...
```

- `True` = 1, `False` = 0

```
import numpy as np
np.mean([0, 1, 0, 0])
```

```
0.25
```

```
np.mean([False, True,
          False, False])
```

```
0.25
```

- Mean of Boolean Series represents percentage of `True` values

Taking the mean of a Boolean Series

```
ri.is_arrested.value_counts(normalize=True)
```

```
False    0.964431  
True     0.035569
```

```
ri.is_arrested.mean()
```

```
0.0355690117407784
```

```
ri.is_arrested.dtype
```

```
dtype('bool')
```

Comparing groups using groupby (1)

- Study the arrest rate by police district

```
ri.district.unique()
```

```
array(['Zone X4', 'Zone K3', 'Zone X1', 'Zone X3',  
      'Zone K1', 'Zone K2'], dtype=object)
```

```
ri[ri.district == 'Zone K1'].is_arrested.mean()
```

```
0.024349083895853423
```

Comparing groups using groupby (2)

```
ri[ri.district == 'Zone K2'].is_arrested.mean()
```

```
0.030800588834786546
```

```
ri.groupby('district').is_arrested.mean()
```

```
district
Zone K1    0.024349
Zone K2    0.030801
Zone K3    0.032311
Zone X1    0.023494
Zone X3    0.034871
Zone X4    0.048038
```

Grouping by multiple categories

```
ri.groupby(['district', 'driver_gender']).is_arrested.mean()
```

district	driver_gender	
Zone K1	F	0.019169
	M	0.026588
Zone K2	F	0.022196
...

```
ri.groupby(['driver_gender', 'district']).is_arrested.mean()
```

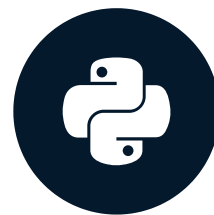
driver_gender	district	
F	Zone K1	0.019169
	Zone K2	0.022196
...

Let's practice!

ANALYZING POLICE ACTIVITY WITH PANDAS

Does gender affect who is frisked during a search?

ANALYZING POLICE ACTIVITY WITH PANDAS



Kevin Markham
Founder, Data School

```
ri.search_conducted.value_counts()
```

False	83229
True	3307

```
ri.search_type.value_counts(dropna=False)
```

NaN	83229
Incident to Arrest	1290
Probable Cause	924
Inventory	219
Reasonable Suspicion	214
Protective Frisk	164
Incident to Arrest, Inventory	123
...	

- `.value_counts()` excludes missing values by default
- `dropna=False` displays missing values

Examining the search types

```
ri.search_type.value_counts()
```

```
Incident to Arrest      1290
Probable Cause          924
Inventory                219
Reasonable Suspicion    214
Protective Frisk        164
Incident to Arrest,Inventory  123
Incident to Arrest,Probable Cause  100
...
```

- Multiple values are separated by commas
- 219 searches in which "Inventory" was the only search type
- Locate "Inventory" among multiple search types

Searching for a string (1)

```
ri['inventory'] = ri.search_type.str.contains('Inventory', na=False)
```

- `str.contains()` returns `True` if string is found, `False` if not found
- `na=False` returns `False` when it finds a missing value

Searching for a string (2)

```
ri.inventory.dtype
```

```
dtype('bool')
```

- `True` means inventory was done, `False` means it was not

```
ri.inventory.sum()
```

```
441
```

Calculating the inventory rate

```
ri.inventory.mean()
```

```
0.0050961449570121106
```

- 0.5% of all traffic stops resulted in an inventory

```
searched = ri[ri.search_conducted == True]  
searched.inventory.mean()
```

```
0.13335349259147264
```

- 13.3% of searches included an inventory

Let's practice!

ANALYZING POLICE ACTIVITY WITH PANDAS