

Crossfilter.js

Manipulando dados

Crossfilter.js



“Crossfilter é uma biblioteca JavaScript para explorar grandes conjuntos de dados no browser. Crossfilter suporta consultas extremamente rápidas (<30ms), mesmo para conjuntos de dados contendo milhões de registros. Esta biblioteca foi contruída e usada pelo time de analytics da Square Register, que permite lojistas explorarem (slice and dice) o seu histórico de pagamentos de maneira fluida.”

<https://github.com/crossfilter/crossfilter>

Crossfilter.js



- Biblioteca de manipulação de dados
 - Filtros e Agrupamentos
- Normalmente utilizada em conjunto com bibliotecas de visualização como D3.js ou DC.js
- Constrói um índice de dados
 - Construção (relativamente) lenta
 - Consultas extremamente rápidas

Dataset



257

Homicide Reports, 1980-2014

Can you develop an algorithm to detect serial killer activity?



Murder Accountability Project • updated 2 years ago (Version 1)

Data (11 MB)



Data Sources

database.csv 24 columns

About this file

Homicide reports from FBI and FOIA requests

Columns

Record ID
A Agency Code
A Agency Name
A Agency Type
A City
A State
Year 2000
A Month
Incident

Primeiros Passos

- Importar o dataset
- Importar a biblioteca:

```
<script src="crossfilter.min.js"></script>
```

- Constrói um objeto crossfilter:

```
var cf = crossfilter(csv);
```

- Primeiro teste:

```
cf.size();
```

Dimensões

- Representam uma certa propriedade dos dados podendo ser
 - Propriedade primitiva (ex: idade da vítima)
 - Propriedade derivada (ex: diferença de idade entre a vítima e o autor do crime)
- Podemos pensar em dimensão como uma coluna da tabela do nosso dataset
- São usadas para definir filtros no dataset e valores que serão agrupados

Dimensões

- Para criar dimensões, usamos a função dimension

```
var victimAgeDimension = cf.dimension(function(d) {  
  return d["Victim Age"];  
});
```

- Podemos criar dimensões derivadas

```
var ageDifferenceDimension = cf.dimension(function(d) {  
  return d["Perpetrator Age"] - d["Victim Age"];  
});
```


Dimensões - Top

- A função top retorna as top-k entradas da tabela de acordo com a dimensão selecionada

```
victimAgeDimension.top(5);
```

(. . .)	State	Victim Age	Victim Count	(. . .)
	Wyoming	998	0	
	North Carolina	998	0	
	Mississippi	998	0	
	Minnesota	998	0	
	New York	998	0	

Dimensões - Bottom

- A função top retorna as bottom-k entradas da tabela de acordo com a dimensão selecionada

```
victimAgeDimension.bottom(5);
```

(. . .)	State	Victim Age	Victim Count	(. . .)
	New Jersey	0	0	
	Alaska	0	0	
	Alaska	0	0	
	Wyoming	0	0	
	North Carolina	0	0	

Dimensões - Filtros

- Como resolver o problema anterior dos dados poluídos?
- A função filter filtra as entradas de acordo com uma função

```
victimAgeDimension.filter(function(d) {  
    return d > 0 && d < 130  
});
```

- Todas as operações posteriores são afetadas por este filtro
- A função filterAll() remove o filtro da dimensão

Dimensões - Filtro + top

- Execute novamente o comando top após ter incluído o filtro
- Qual o resultado?

(. . .)	State	Victim Age	Victim Count	(. . .)
	Michigan	99	0	
	Georgia	99	0	
	Tennessee	99	0	
	Michigan	99	0	
	Virginia	99	0	

Grupos

- Agregações criadas a partir de dimensões
- Similar a um “group by”

```
var weaponDimension = cf.dimension(function(d) {  
    return d.Weapon;  
});
```

```
var weaponGrouping = weaponDimension.group();
```

Grupos - Contagem

- Usando a função `all()` do grupo podemos saber quantos elementos estão presentes em cada grupo

```
weaponGrouping.all();
```

Key	Value
Blunt object	1690
Drowning	25
Drugs	36
Explosives	9
Fall	8

Key	Value
Fire	144
Firearm	1038
Gun	55
Handgun	7567
Knife	1908

Key	Value
Poison	10
Rifle	454
Shotgun	528
Strangulation	166
Suffocation	95

Grupos + Filtros

- Quais são as três armas mais utilizadas nos crimes cometidos por mulheres?
- Podemos usar simultaneamente agrupamentos e filtros

```
var perpetratorSexDimension = cf.dimension(function(d) {  
  return d["Perpetrator Sex"];  
});  
perpetratorSexDimension.filter(function(d) {  
  return d == "Female";  
});  
var weaponDimension = cf.dimension(function(d) {  
  return d["Weapon"];  
});  
var weaponGrouping = weaponDimension.group();  
weaponGrouping.top(3);
```

Key	Value
Handgun	300
Knife	272
Blunt Object	161

Grupos

- Grupos também suportam agregações mais gerais?
- Exemplo: Contando o número total de vítimas extras em todos as entradas

```
cf.groupAll().reduceSum(function(d) {  
    return d["Victim Count"];  
}).value();
```

1740

Exercícios

- Número de homicídios por estado.
- Quantas assassinações aconteceram cometidos por homens (e por mulheres) no estado do Alaska?