Filtering

How We Find Desired Information from Dataset

Filtering

- In general, filtering means throwing out irrelevant stuff (could be harmful) from something. For example, water filtering, air filtering, etc.

- Similarly in data science, filtering means-
  - Refining datasets into simply what a user needs
  - Excluding other data that are irrelevant or sensitive

Different types of strategies can be used to filter out data depending on the user's needs.

Filtering

A real world example of filtering can be preventing access to sensitive information.

For example, a data filter may remove Social Security numbers or credit card numbers from client data before an employee can start working on them.

UserName	credit amount	Gender	Billing Date	CreditCard	SSN
Mason	5000	M	18th	3827383728	3242742
Andrew	3.1	M	22nd	7492748833	9873892
Trevor	3.7	M	11th	2618193385	7847284
Anthony	3.33	M	13th	8901284722	5729175
Robinson	3.45	M	18th	4729108379	4110394
Alice	3.18	F	20th	7191039091	8918383
Katie	3.25	F	7th	4739378748	1129483

Filtering

- After filtering, this data will be sent to an employee as shown

UserName	credit amount	Gender	Billing Date
Mason	5000	M	18th
Andrew	3.1	M	22nd
Trevor	3.7	M	11th
Anthony	3.33	M	13th
Robinson	3.45	M	18th
Alice	3.18	F	20th
Katie	3.25	F	7th

The Need for Filtering Data in a Dataset

- Keep the Relevant Data:

  In data science, a lot of the time data scientists have to work with huge datasets.

  But for a specific analysis, not all data is important to use. In those cases, filtering is very helpful.

  It also helps you focus on specific data rather than looking at the huge dataset all the time.

The Need for Filtering Data in a Dataset

Example:

- Suppose you have a population dataset where individual population data are stored like age, height, weight, hair_color, eye_color, and BMI.
- You are asked to find out how a person's height changes with respect to their age.
- For this scenario, you actually don't need any other information except the height and age columns. So, we can just keep the height, age columns and do further analysis.

Discussion Activity

Suppose we have some data on GPA, GRE and gender.

We want to see the student's performance based on gender (male/female).

GRE	GPA	Gender
316	3.4	M
308	3.1	M
327	3.7	F
310	3.33	F
305	3.45	M
322	3.18	F
316	3.25	M
300	3.4	F
310	3.6	F

Discussion Activity

- What is the average GRE score in both the gender groups? Which group has a better average score?

- What is the average GPA in both the gender groups? Which group has a better average GPA?

GRE	GPA	Gender
316	3.4	M
308	3.1	M
327	3.7	F
310	3.33	F
305	3.45	M
322	3.18	F
316	3.25	M
300	3.4	F
310	3.6	F

Discussion Activity

- What is the maximum GPA from the males?

- What is the minimum GRE score among the females?

GRE	GPA	Gender
316	3.4	M
308	3.1	M
327	3.7	F
310	3.33	F
305	3.45	M
322	3.18	F
316	3.25	M
300	3.4	F
310	3.6	F

How to Filter a Dataset

1. Read CSV Data into Pandas Dataframe
   - Import Pandas Library
   - Read CSV data and Save in Variable
   - Display Dataframe Contents

2. Filter Data Based on Requirement
   - Look at all the features (columns)
   - Keep the features (columns) which are required
   - Filter data as per requirement

Filering in Jupyter Lab and Blockly

- Suppose we want to find the cars which:
  - have only 4 cylinders
  - name is "toyota corolla"
  - model_year is after 1980

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	name
0	18.0	8	307.0	130.0	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165.0	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150.0	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150.0	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140.0	3449	10.5	70	1	ford torino
...	...	...	...	...	...	...	...	...	...
393	27.0	4	140.0	86.0	2790	15.6	82	1	ford mustang gl
394	44.0	4	97.0	52.0	2130	24.6	82	2	vw pickup
395	32.0	4	135.0	84.0	2295	11.6	82	1	dodge rampage
396	28.0	4	120.0	79.0	2625	18.6	82	1	ford ranger
397	31.0	4	119.0	82.0	2720	19.4	82	1	chevy s-10

Filtering in Jupyter Lab and Blockly

- Filter the required features from the dataset

```
car_data = car_data[['cylinders', 'name', 'model_year']]

car_data
```

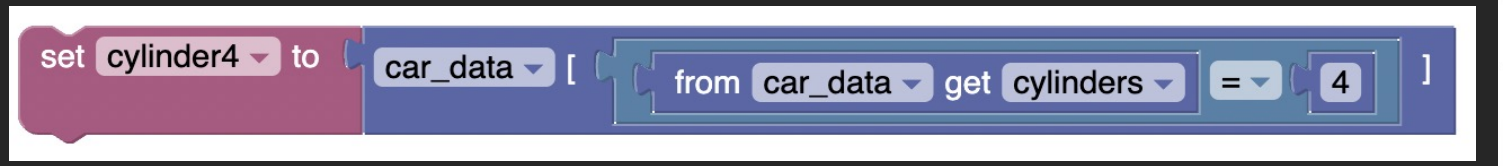
	cylinders	name	model_year
0	8	chevrolet chevelle malibu	70
1	8	buick skylark 320	70
2	8	plymouth satellite	70
3	8	amc rebel sst	70
4	8	ford torino	70
...	...	...	...
393	4	ford mustang gl	82
394	4	vw pickup	82
395	4	dodge rampage	82
396	4	ford ranger	82
397	4	chevy s-10	82

398 rows × 3 columns

Filtering in Jupyter Lab and Blockly

- Filter the cars with 4 cylinders

```
cylinder4 = car_data[(car_data.cylinders == 4)]
```




```
cylinder4
```

	cylinders	name	model_year
14	4	toyota corona mark ii	70
18	4	datsun pl510	70
19	4	volkswagen 1131 deluxe sedan	70
20	4	peugeot 504	70
21	4	audi 100 ls	70
...	...	...	...
393	4	ford mustang gl	82
394	4	vw pickup	82
395	4	dodge rampage	82
396	4	ford ranger	82
397	4	chevy s-10	82

204 rows × 3 columns

Filtering in Jupyter Lab and Blockly

- Filter the cars which name is "toyota corolla"

```
toyota_cars = cylinder4[(cylinder4.name == 'toyota corolla')]
```


```
toyota_cars
```

	cylinders	name	model_year
167	4	toyota corolla	75
205	4	toyota corolla	76
321	4	toyota corolla	80
356	4	toyota corolla	81
382	4	toyota corolla	82

Filtering in Jupyter Lab and Blockly

- Filter the cars which model_year is after 1980 (in our model_year column 80 represents 1980)

```
filtered_cars = toyota_cars[(toyota_cars.model_year > 80)]
```


```
filtered_cars
```

	cylinders	name	model_year
356	4	toyota corolla	81
382	4	toyota corolla	82

Finally found 2 cars out of 398

Summary:

- Initial list contained 398 cars.
- After filtering by our requirements, we have a list of only 2 cars with the required features.
- By using filtering in a big dataset, we can more easily find the data we care about

Reference notebook:

- FilterData_Ex.ipynb