

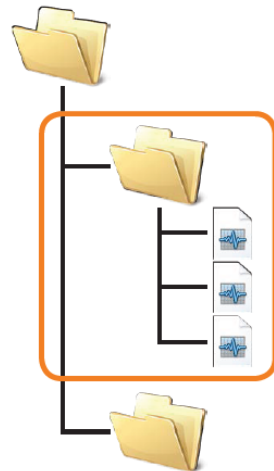
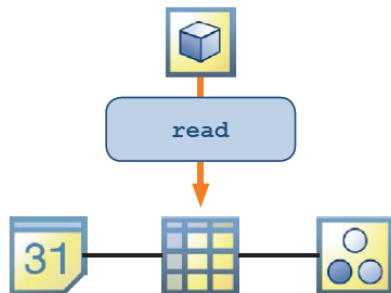
Datastores

MATLAB[®] for Data Processing and Visualization

Duy NGUYEN
Engineering Development Group

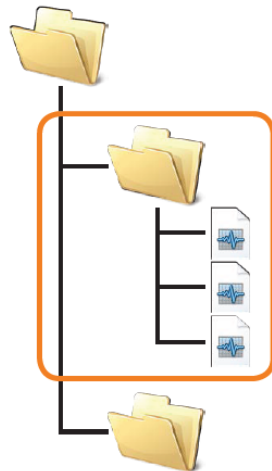
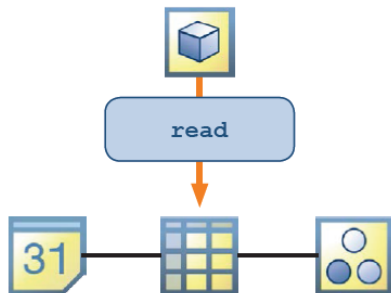
What will you learn today?

- Import data efficiently in MATLAB using datastores



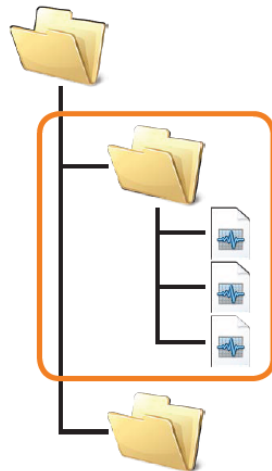
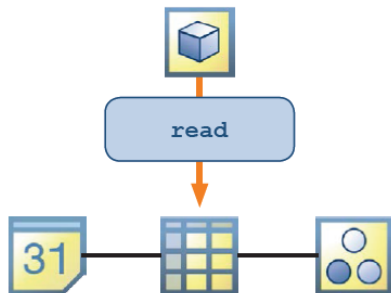
What will you learn today?

- Import data efficiently in MATLAB using datastores
- Specify variables and types to read



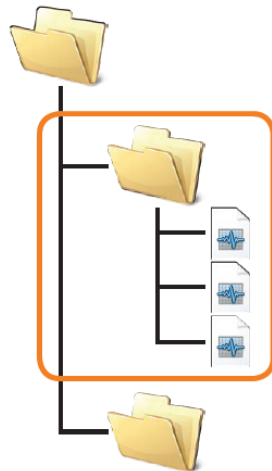
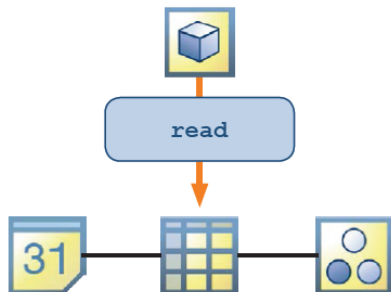
What will you learn today?

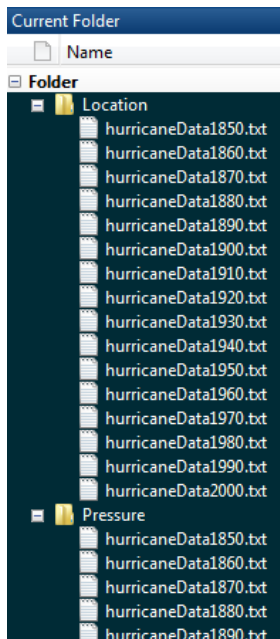
- Import data efficiently in MATLAB using datastores
- Specify variables and types to read
- Import and merge data from multiple sources



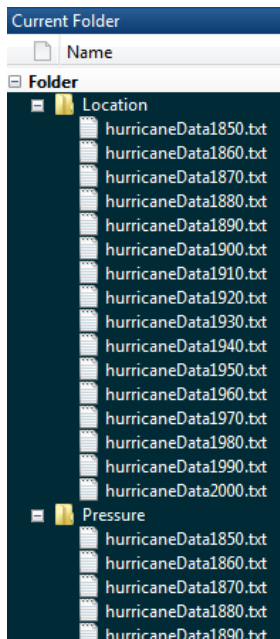
What will you learn today?

- Import data efficiently in MATLAB using datastores
- Specify variables and types to read
- Import and merge data from multiple sources
- Visualize data with 2- and 3-dimensional customized plots





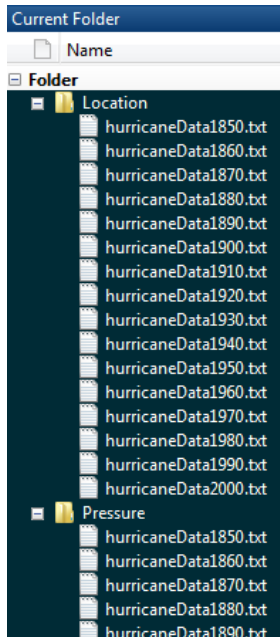
How do you currently import text files in MATLAB?



How do you currently import text files in MATLAB?

Underlying issues

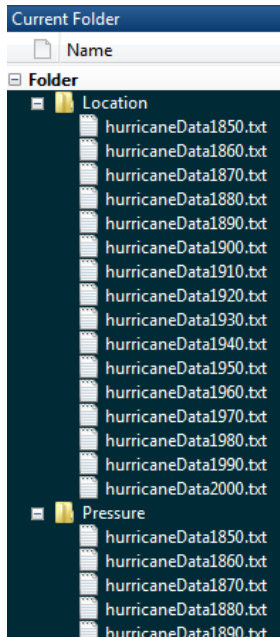
- Huge file sizes



How do you currently import text files in MATLAB?

Underlying issues

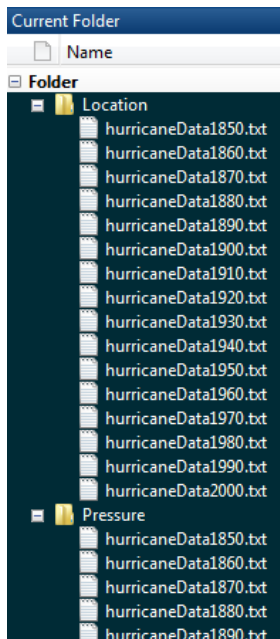
- Huge file sizes
- Thousands of files in multiple subfolders



How do you currently import text files in MATLAB?

Underlying issues

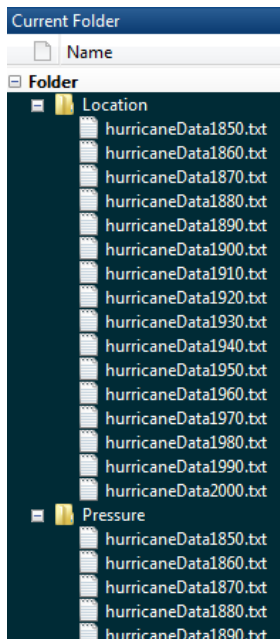
- Huge file sizes
- Thousands of files in multiple subfolders
- Arbitrary, non sequential file names



How do you currently import text files in MATLAB?

Underlying issues

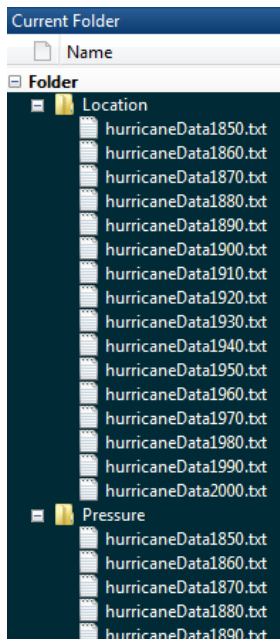
- Huge file sizes
- Thousands of files in multiple subfolders
- Arbitrary, non sequential file names
- Header lines, lines that are comments



How do you currently import text files in MATLAB?

Underlying issues

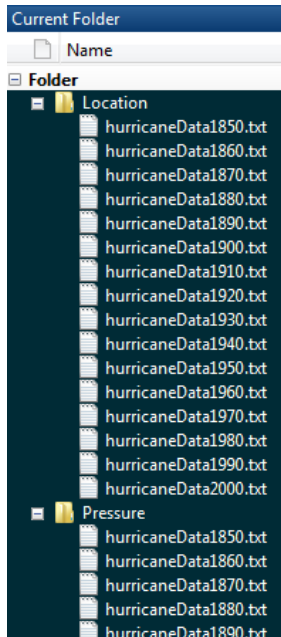
- **Huge file sizes**
- **Thousands of files in multiple subfolders**
- **Arbitrary, non sequential file names**
- Header lines, lines that are comments
- Different delimiters



How do you currently import text files in MATLAB?

Underlying issues

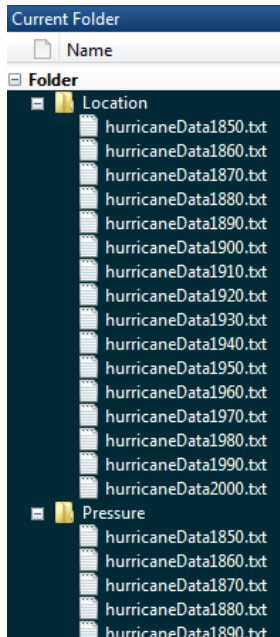
- Huge file sizes
- Thousands of files in multiple subfolders
- Arbitrary, non sequential file names
- Header lines, lines that are comments
- Different delimiters
- Different data types



How do you currently import text files in MATLAB?

Underlying issues

- **Huge file sizes**
- **Thousands of files in multiple subfolders**
- **Arbitrary, non sequential file names**
- Header lines, lines that are comments
- Different delimiters
- Different data types
- Whitespaces



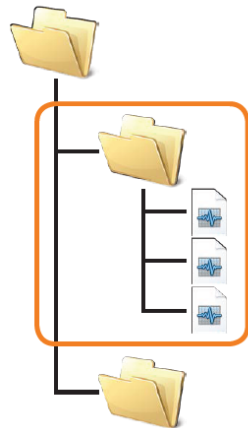
How do you currently import text files in MATLAB?

Underlying issues

- **Huge file sizes**
- **Thousands of files in multiple subfolders**
- **Arbitrary, non sequential file names**
- Header lines, lines that are comments
- Different delimiters
- Different data types
- Whitespaces
- ...

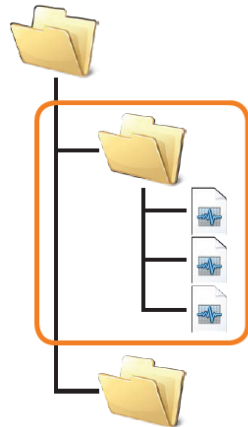
What is a datastore?

- A datastore is a particular type of MATLAB object



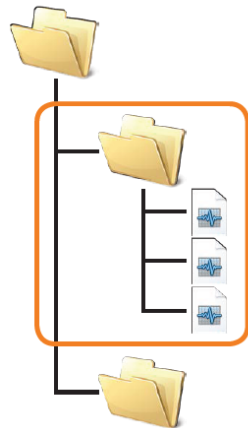
What is a datastore?

- A datastore is a particular type of MATLAB object
 - ▶ It is used to read a **single file** or a **collection of files** or **data**



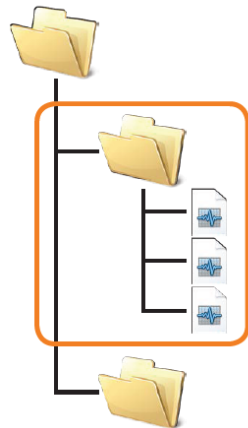
What is a datastore?

- A datastore is a particular type of MATLAB object
 - ▶ It is used to read a **single file** or a **collection of files** or **data**
- A datastore acts as a **reference to a data source**



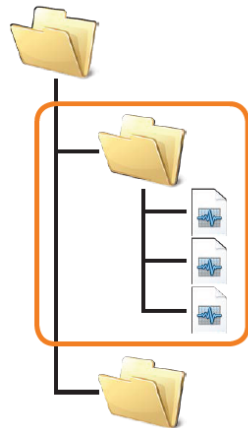
What is a datastore?

- A datastore is a particular type of MATLAB object
 - ▶ It is used to read a **single file** or a **collection of files** or **data**
- A datastore acts as a **reference to a data source**
 - ▶ Data must have the same structure and formatting



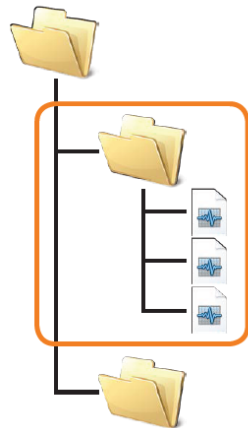
What is a datastore?

- A datastore is a particular type of MATLAB object
 - ▶ It is used to read a **single file** or a **collection of files** or **data**
- A datastore acts as a **reference to a data source**
 - ▶ Data must have the same structure and formatting
- Datastores are especially useful when:



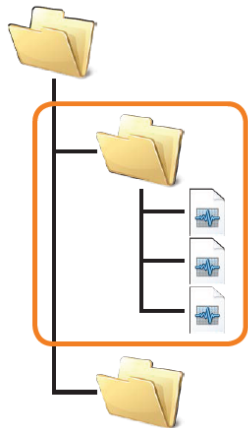
What is a datastore?

- A datastore is a particular type of MATLAB object
 - ▶ It is used to read a **single file** or a **collection of files** or **data**
- A datastore acts as a **reference to a data source**
 - ▶ Data must have the same structure and formatting
- Datastores are especially useful when:
 - ▶ Each file in the collection might be **too large** to fit in memory



What is a datastore?

- A datastore is a particular type of MATLAB object
 - ▶ It is used to read a **single file** or a **collection of files** or **data**
- A datastore acts as a **reference to a data source**
 - ▶ Data must have the same structure and formatting
- Datastores are especially useful when:
 - ▶ Each file in the collection might be **too large** to fit in memory
 - ▶ Files in the collection have **arbitrary names**



[illegible]

- **TabularTextDatastore** **Text files containing column-oriented data**

What are the different types of datastores?

Year	Month	Day	Time	Origin	Destination	Delay	Dist
2000	1	2	10:00	LAX	SEA	10	1
2000	1	3	10:00	LAX	SEA	10	1
2000	1	4	10:00	LAX	SEA	10	1
2000	1	5	10:00	LAX	SEA	10	1
2000	1	6	10:00	LAX	SEA	10	1
2000	1	7	10:00	LAX	SEA	10	1
2000	1	8	10:00	LAX	SEA	10	1
2000	1	9	10:00	LAX	SEA	10	1
2000	1	10	10:00	LAX	SEA	10	1
2000	1	11	10:00	LAX	SEA	10	1
2000	1	12	10:00	LAX	SEA	10	1
2000	1	13	10:00	LAX	SEA	10	1
2000	1	14	10:00	LAX	SEA	10	1
2000	1	15	10:00	LAX	SEA	10	1
2000	1	16	10:00	LAX	SEA	10	1
2000	1	17	10:00	LAX	SEA	10	1
2000	1	18	10:00	LAX	SEA	10	1
2000	1	19	10:00	LAX	SEA	10	1
2000	1	20	10:00	LAX	SEA	10	1
2000	1	21	10:00	LAX	SEA	10	1
2000	1	22	10:00	LAX	SEA	10	1
2000	1	23	10:00	LAX	SEA	10	1

■ `TabularTextDatastore` Text files containing column-oriented data

■ `ImageDatastore` Image files

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	
1	Year	Month	Day	Hour	Minute	Second	CRSDepArrTime	CRSArrTime	FlightNum	TailNum	ActualDepArrTime	ActualCRSDepArrTime	Airline	Delay	DepDelay	Q	R		
2	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
3	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
4	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
5	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
6	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
7	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
8	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
9	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
10	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
11	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
12	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
13	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
14	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
15	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
16	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
17	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
18	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
19	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
20	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
21	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
22	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
23	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
24	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
25	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
26	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
27	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
28	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
29	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
30	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
31	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
32	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
33	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
34	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
35	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
36	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
37	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
38	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
39	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
40	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
41	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
42	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
43	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
44	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
45	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
46	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
47	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
48	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
49	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
50	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
51	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
52	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
53	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
54	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
55	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
56	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
57	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
58	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
59	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
60	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
61	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
62	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
63	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
64	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
65	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
66	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
67	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
68	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
69	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
70	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
71	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
72	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
73	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
74	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
75	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
76	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
77	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
78	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
79	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
80	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
81	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
82	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
83	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
84	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
85	2000	3	2	4	1641	1635	1831	188	WN	170	1726	NA	50	55	41	1	6.5P	LBB	
86	2000	3	2	4	1641	1635	1												

- 5 / 28

What are the different types of datastores?

The image shows a complex spreadsheet with multiple overlapping sheets. The visible sheets contain columns for Year, Month, Day, and various flight-related data points such as flight number, origin, destination, and status. The data is organized in a structured, tabular format, typical of a database or a large dataset.

- **TabularTextDatastore** **Text files containing column-oriented data**
- **ImageDatastore** **Image files**
- **KeyValueDatastore** **Key-value pair data**
- **DatabaseDatastore** **Data from a relational database**

Course example: Hurricane data

- US National Hurricane Center: <http://weather.unisys.com/hurricane/atlantic>



Course example: Hurricane data

Number	Timestamp	Latitude	Longitude	Country
## 1 ALLISON M = 55 SNBR = 1303				
1	05-Jun-2001 12:00	27.5	95	N/A
1	05-Jun-2001 18:00	28.5	95.3	N/A
1	06-Jun-2001 00:00	29.3	95.3	United States
1	06-Jun-2001 06:00	30.1	95.2	United States
		⋮		
## 2 BARRY M = 24 SNBR = 1304				
2	02-Aug-2001 12:00	25.7	84.8	N/A
2	02-Aug-2001 18:00	26.2	84.9	N/A
2	03-Aug-2001 00:00	26.4	85.6	N/A
		⋮		

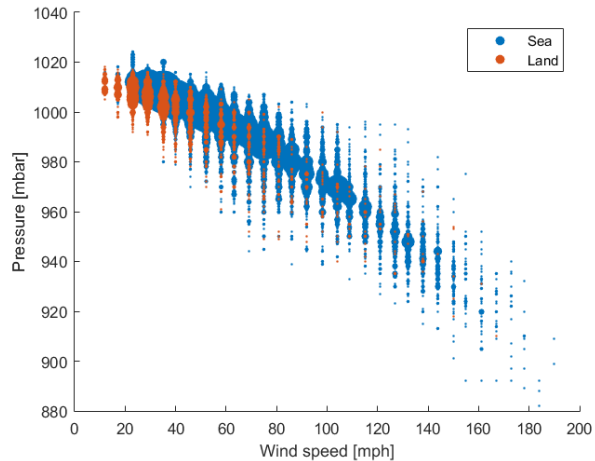
Course example: Hurricane data

Number	Timestamp	Pressure
## 1 ALLISON M = 55 SNBR = 1303		
1	05-Jun-2001 12:00	1007
1	05-Jun-2001 18:00	1002
1	06-Jun-2001 00:00	1003
1	06-Jun-2001 06:00	1006
	⋮	
## 2 BARRY M = 24 SNBR = 1304		
2	02-Aug-2001 12:00	1011
2	02-Aug-2001 18:00	1010
2	03-Aug-2001 00:00	1007
	⋮	

Course example: Hurricane data

Number	Timestamp	Windspeed
## 1 ALLISON M = 55 SNBR = 1303		
1	05-Jun-2001 12:00	46
1	05-Jun-2001 18:00	58
1	06-Jun-2001 00:00	52
1	06-Jun-2001 06:00	35
	⋮	
## 2 BARRY M = 24 SNBR = 1304		
2	02-Aug-2001 12:00	35
2	02-Aug-2001 18:00	52
2	03-Aug-2001 00:00	46
	⋮	

Course example: Hurricane data



Course example: Hurricane data

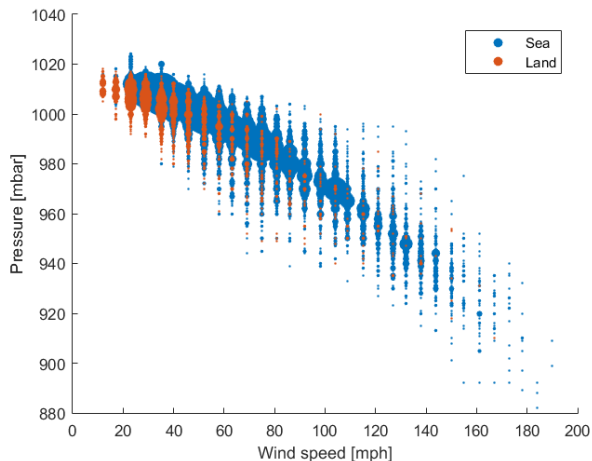
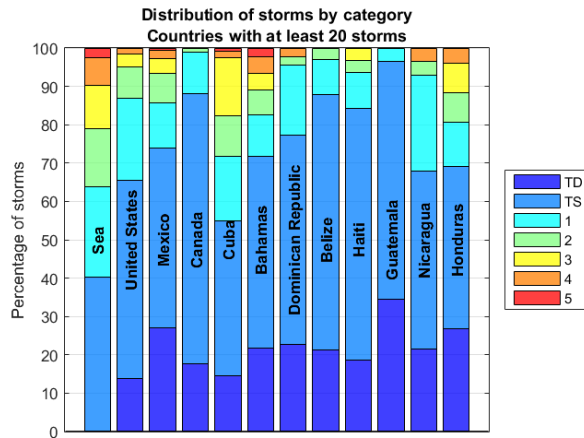
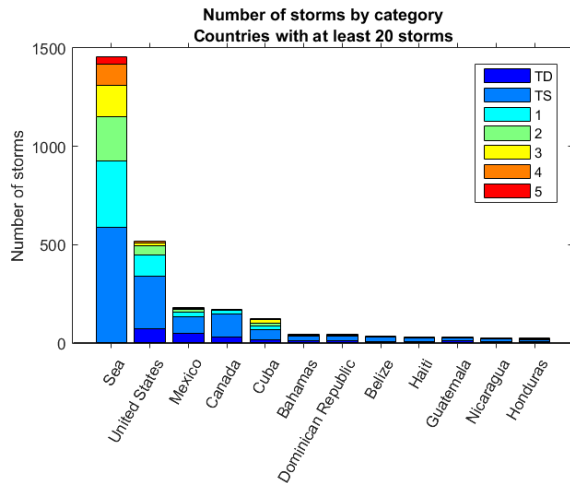


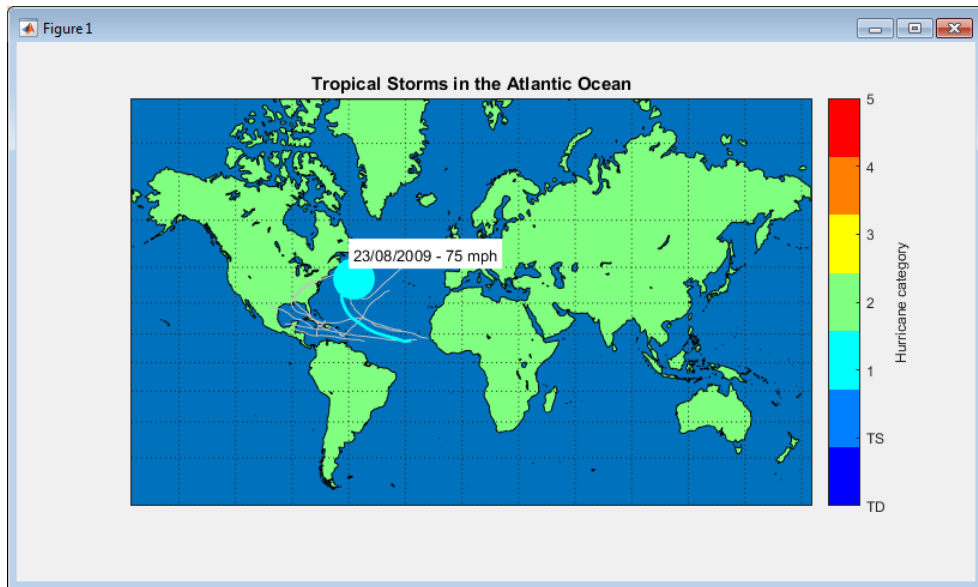
Table: Saffir-Simpson hurricane scale

Category	Wind speed
Tropical Depression (TD)	0 – 38 mph
Tropical Storm (TS)	39 – 73 mph
Category 1 Hurricane	74 – 95 mph
Category 2 Hurricane	96 – 110 mph
Category 3 Hurricane	111 – 130 mph
Category 4 Hurricane	131 – 155 mph
Category 5 Hurricane	≥ 156 mph

Course example: Hurricane data



Course example: Hurricane data



How do I create a datastore?

```
>> src = datastore('myLocation');
```



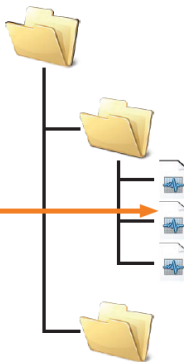
Delimiter

CommentStyle

VariableNames

TextscanFormats

...



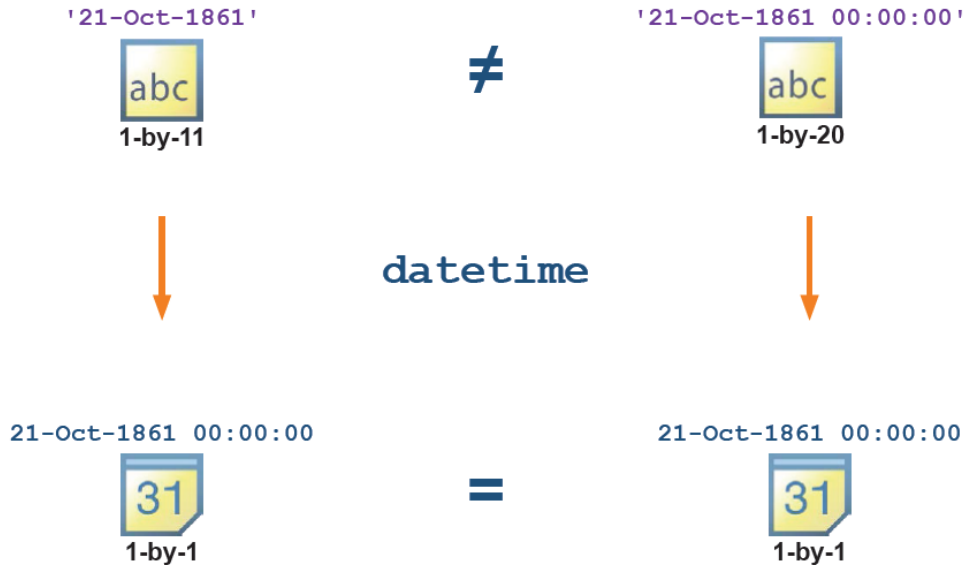
preview()

read()

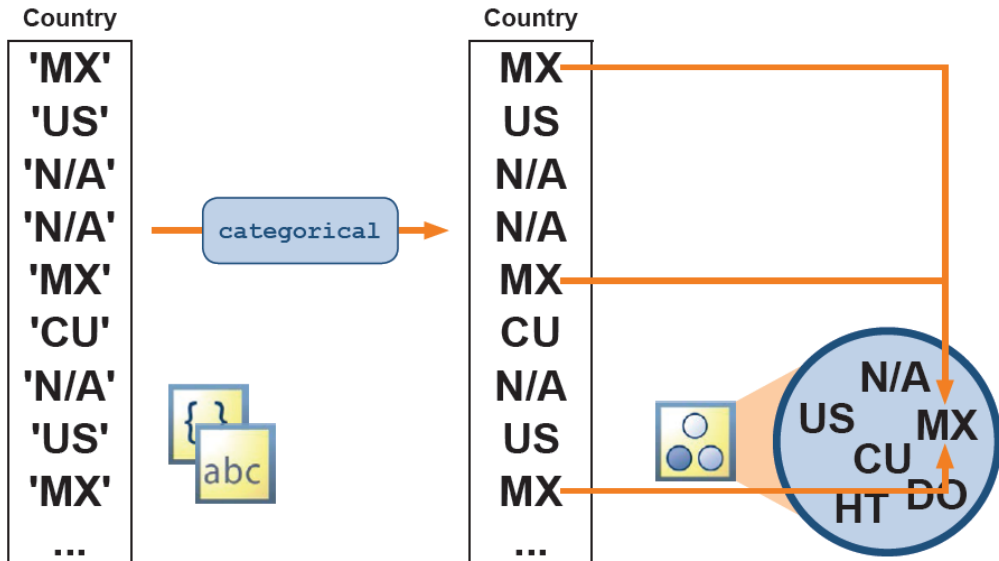


```
11 04-Jan-1955 12:00:00 13.1 64.9 N/A
11 04-Jan-1955 18:00:00 14.7 64.4 N/A
11 05-Jan-1955 00:00:00 14.4 64.0 N/A
11 05-Jan-1955 06:00:00 14.0 63.6 N/A
11 05-Jan-1955 12:00:00 13.6 63.3 N/A
11 05-Jan-1955 18:00:00 13.3 63.1 N/A
11 06-Jan-1955 00:00:00 13.0 63.0 N/A
11 06-Jan-1955 06:00:00 12.8 63.0 N/A
## 1 ENDROW M = 11 ENBA = 846
1 31-Jul-1955 18:00:00 27.5 88.4 N/A
1 01-Aug-1955 00:00:00 28.2 88.6 N/A
1 01-Aug-1955 06:00:00 28.8 88.8 N/A
1 01-Aug-1955 12:00:00 29.3 89.0 N/A
1 01-Aug-1955 18:00:00 29.9 89.5 United States
1 02-Aug-1955 00:00:00 30.6 90.4 United States
1 02-Aug-1955 06:00:00 31.1 91.6 United States
1 02-Aug-1955 12:00:00 31.5 92.8 United States
1 02-Aug-1955 18:00:00 31.8 93.0 United States
1 03-Aug-1955 00:00:00 32.1 95.0 United States
1 03-Aug-1955 06:00:00 32.5 96.2 United States
## 2 COMPLETE M = 49 ENBA = 847
2 03-Aug-1955 06:00:00 15.3 35.5 N/A
```

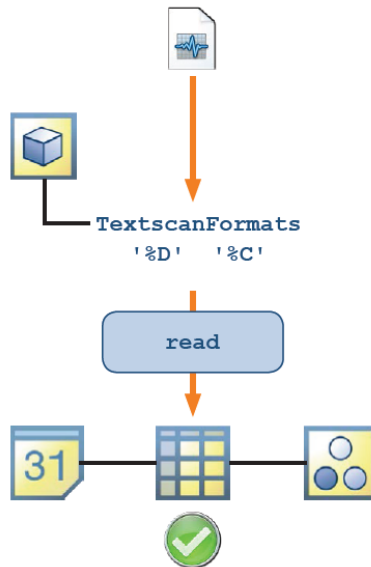
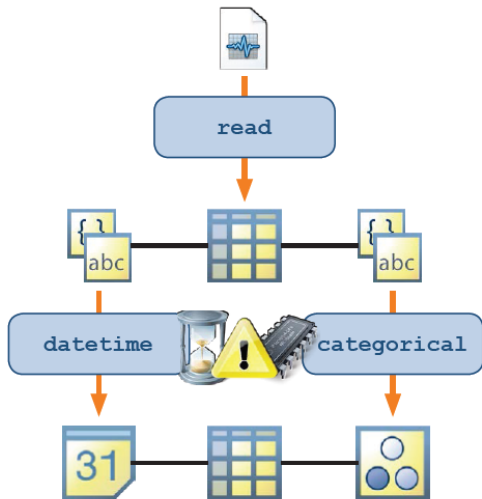
Use `datetime` to compare dates and times stored as strings



Use categorical variables to represent discrete categories



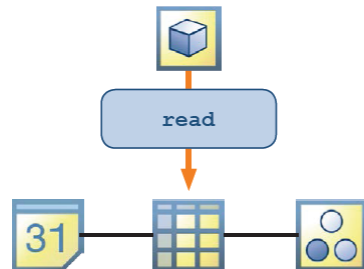
How do I import data types directly?



Most commonly used format strings for the `TextscanFormats` property

Conversion specifiers

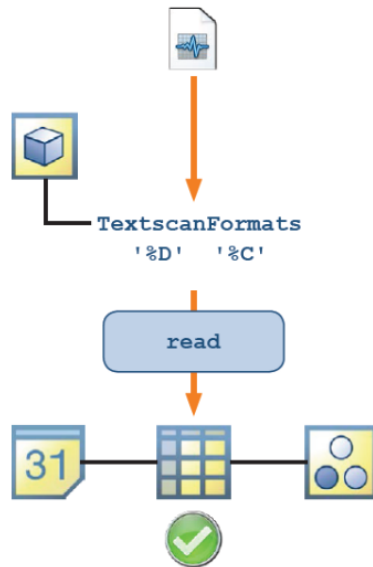
- '%f' Floating point number (double)
- '%q' String (cell array of char)
- '%D' Date (datetime)
- '%C' Category (categorical)
- '%d' Signed integer (int32)
- '%u' Unsigned integer (uint32)
- '%c' Single character (char)



Select appropriate data types to reduce memory consumption

Table: Memory consumption

	Before	After	Memory (%)
Using datetime:	726 KB	76 KB	89%
Using categorical arrays:	573 KB	8 KB	98%
Whole table (4892 × 5):	1416 KB	201 KB	86%



How can I select the columns of data I want to import?

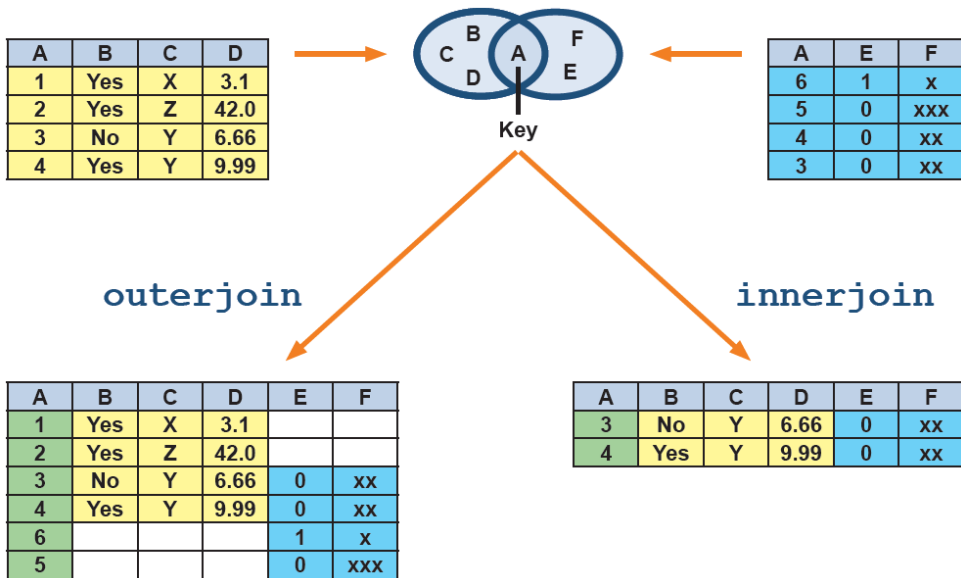


VariableNames

SelectedVariableNames

1	2	3	4	5
Number	Timestamp	Latitude	Longitude	Country
2	12-Aug-1955 12:00:00	34.8000	76.2000	N/A
2	12-Aug-1955 18:00:00	35.6000	76.2000	United States
2	13-Aug-1955 00:00:00	36.6000	75.9000	N/A
2	13-Aug-1955 06:00:00	37.9000	75.9000	N/A
2	13-Aug-1955 12:00:00	39.2000	76.4000	N/A
2	13-Aug-1955 18:00:00	40.7000	77.4000	United States
2	14-Aug-1955 00:00:00	42.9000	80.2000	Canada
2	14-Aug-1955 06:00:00	43.7000	81.6000	Canada
2	14-Aug-1955 12:00:00	44.5000	82.6000	United States
2	14-Aug-1955 18:00:00	45.4000	83.4000	United States
2	15-Aug-1955 00:00:00	46.3000	83.4000	Canada
3	07-Aug-1955 06:00:00	17.1000	43.1000	N/A
3	07-Aug-1955 12:00:00	17.1000	44.1000	N/A
3	07-Aug-1955 18:00:00	17.2000	45.1000	N/A
3	08-Aug-1955 00:00:00	17.3000	46.1000	N/A
3	08-Aug-1955 06:00:00	17.5000	46.9000	N/A
3	08-Aug-1955 12:00:00	17.7000	47.8000	N/A
3	08-Aug-1955 18:00:00	17.9000	49.1000	N/A
3	09-Aug-1955 00:00:00	18.1000	50.2000	N/A

How do I merge data?



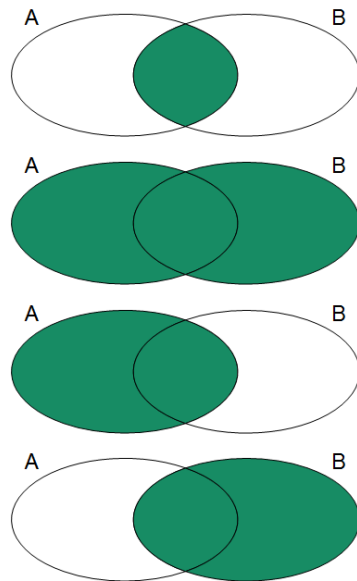
What are the differences between `outerjoin` and `innerjoin` functions?

```
C = innerjoin(A, B)
```

```
C = outerjoin(A, B, 'MergeKeys', true)
```

```
C = outerjoin(A, B, 'MergeKeys', true, 'Type', 'Left')
```

```
C = outerjoin(A, B, 'MergeKeys', true, 'Type', 'Right')
```

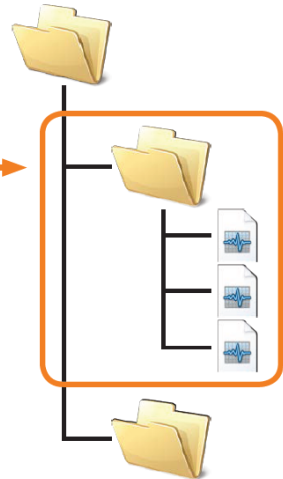


How do I import data from different files?

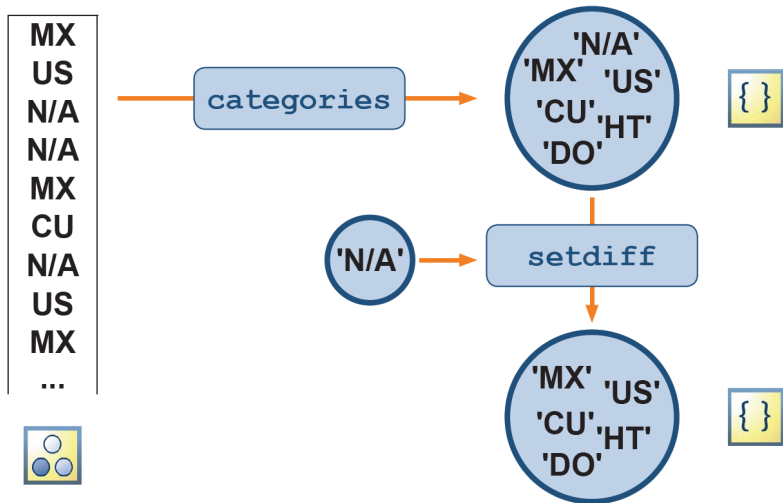
```
>> src = datastore('myLocation');
```



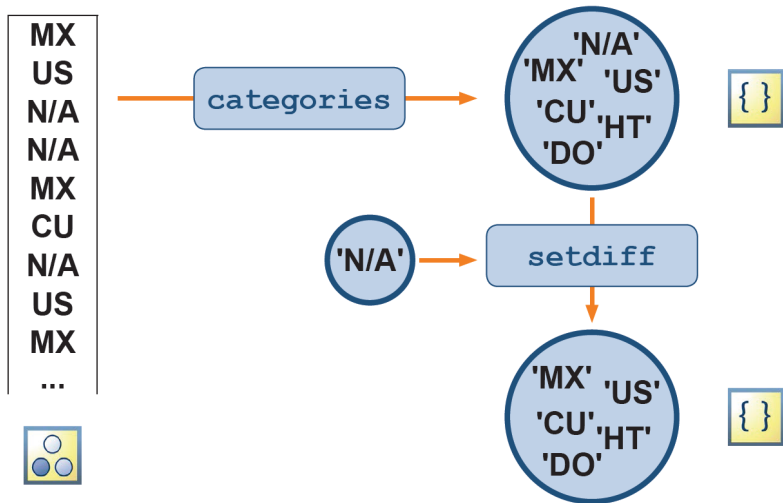
```
read  
readall
```



Categories and set operations

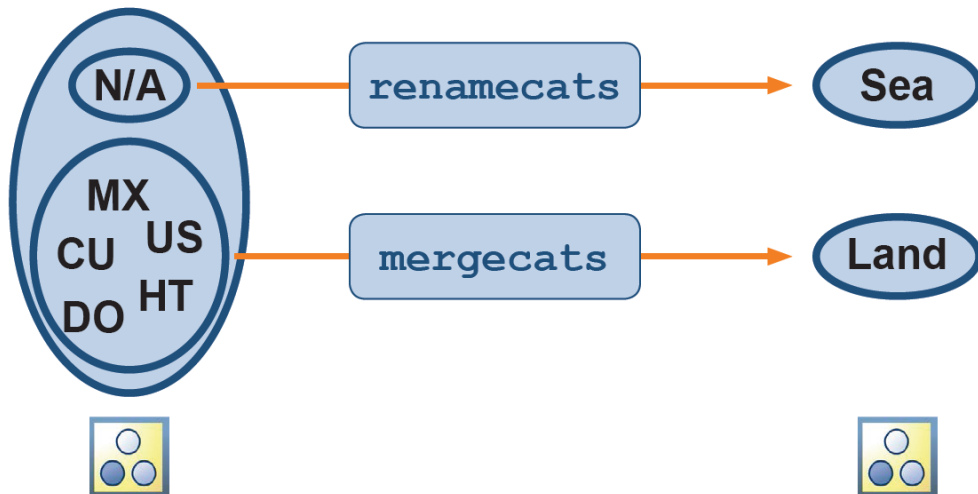


Categories and set operations

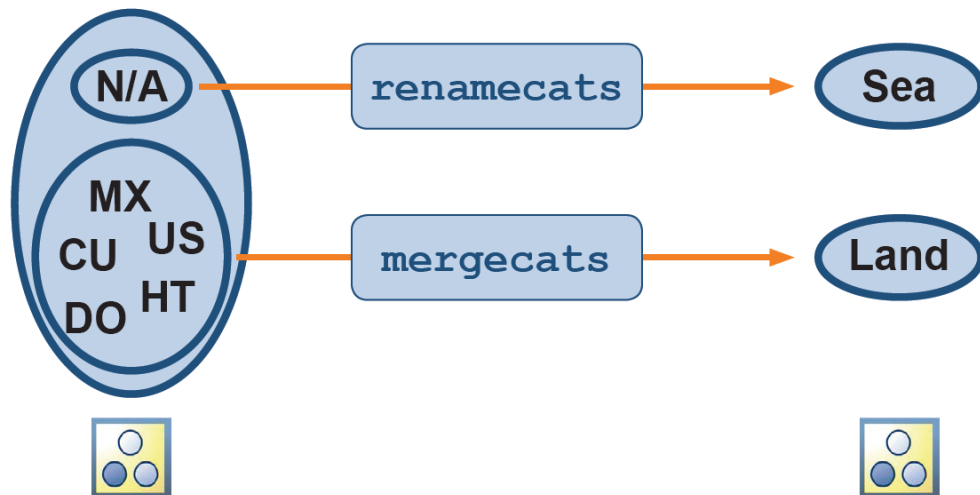


```
■ countries = setdiff(categories(data.Country), 'N/A')
```

How do I rename and merge categories?



How do I rename and merge categories?



```
■ data.Location = mergocats(data.Country,ctry, 'Land')  
  data.Location = renamecats(data.Location, 'N/A', 'Sea')
```

How do I perform calculations with missing data?

1 Number	2 Timestamp	3 Country	4 Windspeed	5 Pressure	6 Location
3	12-Aug-1955 06:00:00	N/A	104	NaN	Sea
2	12-Aug-1955 12:00:00	N/A	81	965	Sea
3	12-Aug-1955 12:00:00	N/A	121	975	Sea
2	12-Aug-1955 18:00:00	United ...	75	962	Land
3	12-Aug-1955 18:00:00	N/A	121	NaN	Sea
2	13-Aug-1955 00:00:00	N/A	69	969	Sea
3	13-Aug-1955 00:00:00	N/A	121	NaN	Sea
2	13-Aug-1955 06:00:00	N/A	58	974	Sea
3	13-Aug-1955 06:00:00	N/A	121	969	Sea
2	13-Aug-1955 12:00:00	N/A	52	982	Sea
3	13-Aug-1955 12:00:00	N/A	121	NaN	Sea
2	13-Aug-1955 18:00:00	United ...	40	995	Land
3	13-Aug-1955 18:00:00	N/A	121	NaN	Sea
2	14-Aug-1955 00:00:00	United ...	35	998	Land
3	14-Aug-1955 00:00:00	N/A	121	NaN	Sea
2	14-Aug-1955 06:00:00	Canada	29	002	Land
3	14-Aug-1955 06:00:00	N/A	121	NaN	Sea
2	14-Aug-1955 12:00:00	Canada	29	006	Land

```
>> mean(data.Pressure)
```

NaN

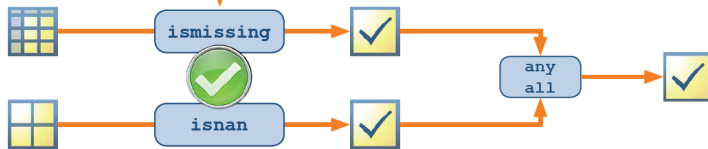
```
>> mean(data.Pressure, 'omitnan')
```

991.7681

How do I locate and remove missing data?

1	2	3	4	5	6
Number	Timestamp	Country	Windspeed	Sea	Location
3	12-Aug-1955 06:00:00	N/A	104	NaN	Sea
2	12-Aug-1955 12:00:00	N/A	81	965	Sea
3	12-Aug-1955 12:00:00	N/A	121	975	Sea
2	12-Aug-1955 18:00:00	United ...	75	96	Land
3	12-Aug-1955 18:00:00	N/A	121	NaN	Sea
2	13-Aug-1955 00:00:00	N/A	69	969	Sea
3	13-Aug-1955 00:00:00	N/A	121	NaN	Sea
2	13-Aug-1955 06:00:00	N/A	58	974	Sea
3	13-Aug-1955 06:00:00	N/A	121	969	Sea
2	13-Aug-1955 12:00:00	N/A	52	982	Sea
3	13-Aug-1955 12:00:00	N/A	121	NaN	Sea
2	13-Aug-1955 18:00:00	United ...	40	965	Land
3	13-Aug-1955 18:00:00	N/A	121	NaN	Sea
2	14-Aug-1955 00:00:00	United ...	35	988	Land
3	14-Aug-1955 00:00:00	N/A	121	NaN	Sea
2	14-Aug-1955 06:00:00	Canada	29	100	Land
3	14-Aug-1955 06:00:00	N/A	121	NaN	Sea
2	14-Aug-1955 12:00:00	Canada	29	1006	Land

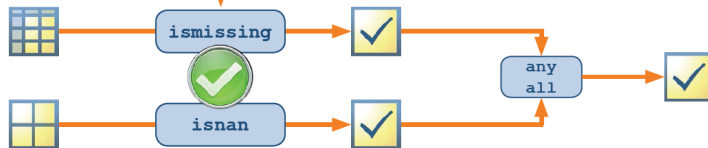
data == NaN



How do I locate and remove missing data?

1	2	3	4	5	6
Number	Timestamp	Country	Windspeed	Seas	Location
3	12-Aug-1955 06:00:00	N/A	104	NaN	Sea
2	12-Aug-1955 12:00:00	N/A	81	965	Sea
3	12-Aug-1955 12:00:00	N/A	121	975	Sea
2	12-Aug-1955 18:00:00	United ...	75	96	Land
3	12-Aug-1955 18:00:00	N/A	121	NaN	Sea
2	13-Aug-1955 00:00:00	N/A	69	960	Sea
3	13-Aug-1955 00:00:00	N/A	121	NaN	Sea
2	13-Aug-1955 06:00:00	N/A	58	974	Sea
3	13-Aug-1955 06:00:00	N/A	121	969	Sea
2	13-Aug-1955 12:00:00	N/A	52	982	Sea
3	13-Aug-1955 12:00:00	N/A	121	NaN	Sea
2	13-Aug-1955 18:00:00	United ...	40	965	Land
3	13-Aug-1955 18:00:00	N/A	121	NaN	Sea
2	14-Aug-1955 00:00:00	United ...	35	988	Land
3	14-Aug-1955 00:00:00	N/A	121	NaN	Sea
2	14-Aug-1955 06:00:00	Canada	29	100	Land
3	14-Aug-1955 06:00:00	N/A	121	NaN	Sea
2	14-Aug-1955 12:00:00	Canada	29	1006	Land

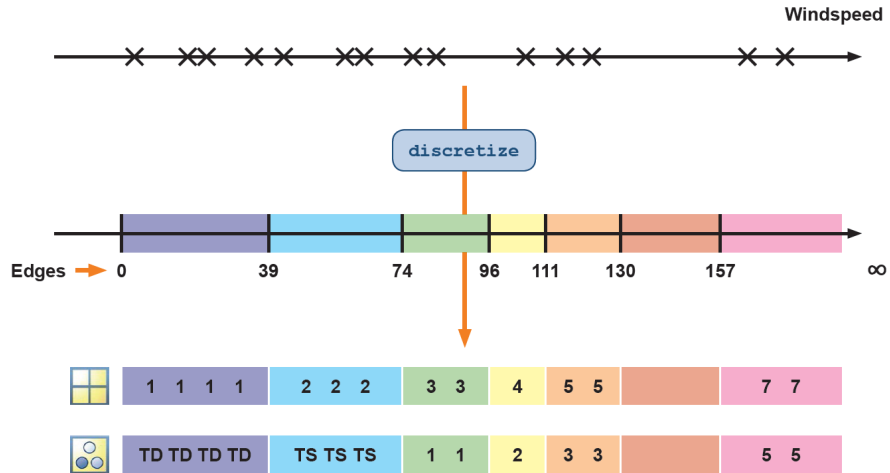
data == NaN



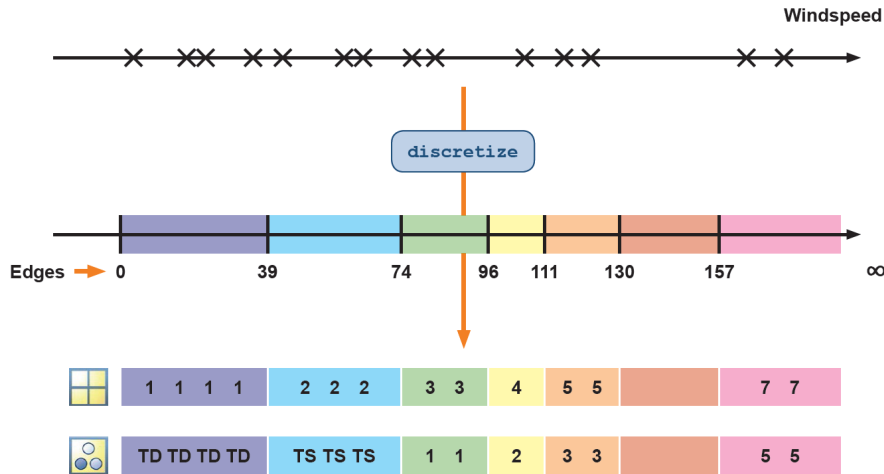
```

■ badrows = any(ismissing(data), 2)
  data(badrows, :) = []
  
```

How do I discretize continuous data?

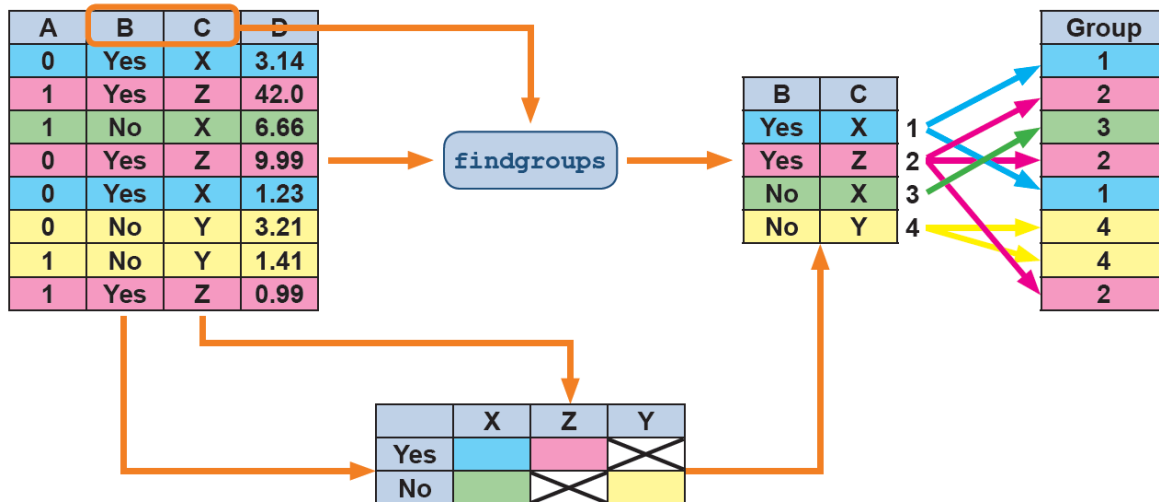


How do I discretize continuous data?

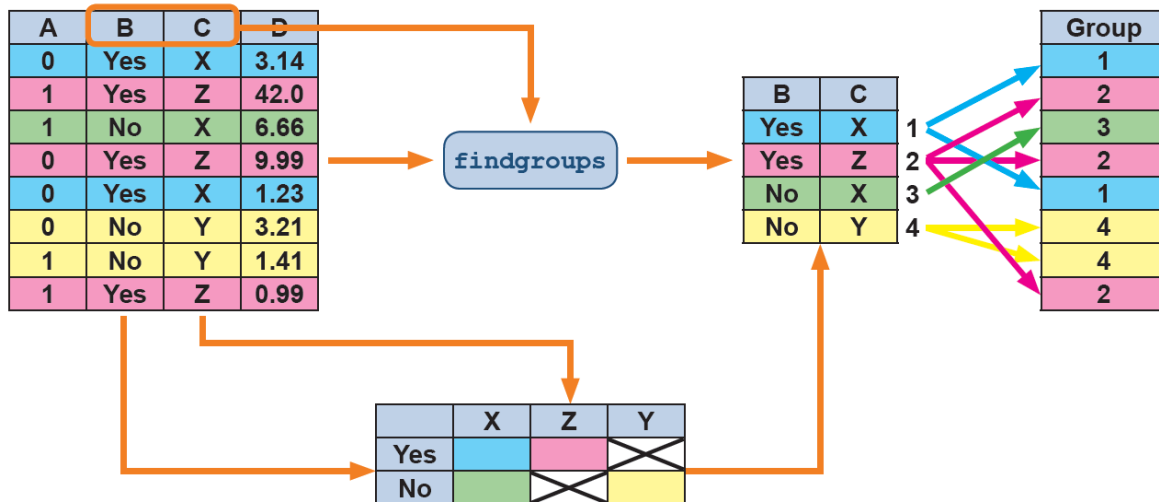


```
■ SSscale = [0 39 74 96 111 130 157 Inf]
  catnames = {'TD', 'TS', '1', '2', '3', '4', '5'}
  data.HurrCat = discretize(data.Windspeed, SSscale, 'Categorical', catnames)
```

How do I find unique groups of data?

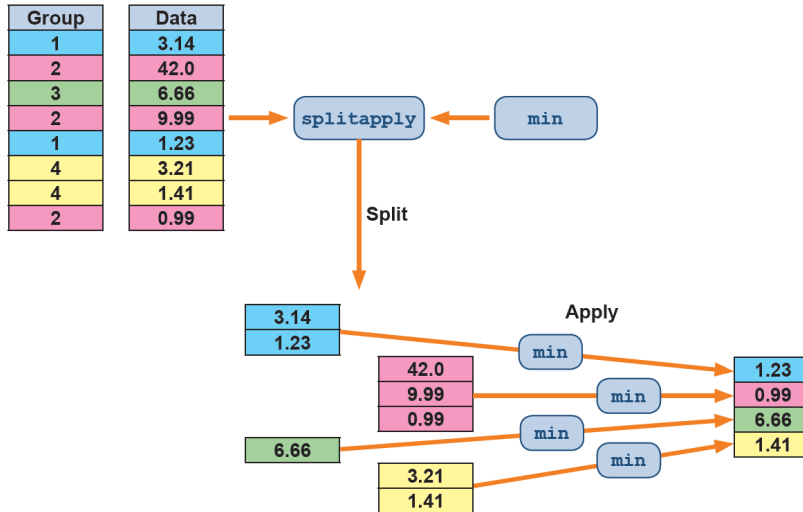


How do I find unique groups of data?

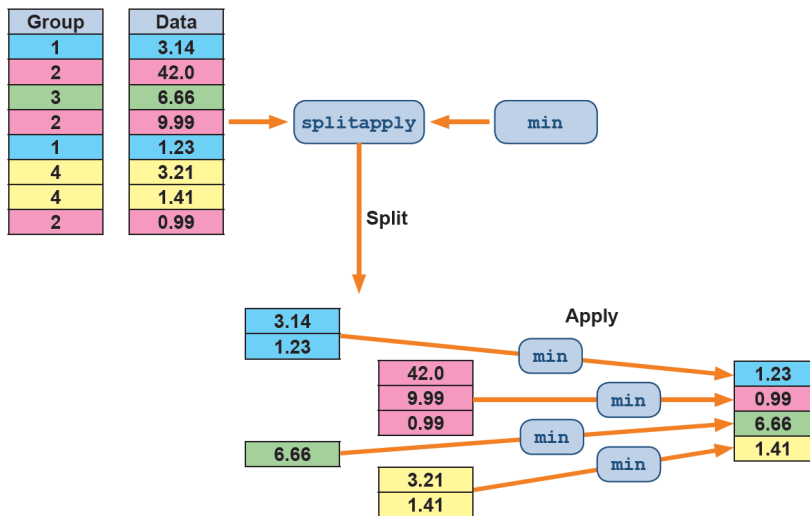


```
■ [Group, BC] = findgroups(mytable(:, {'B', 'C'}))
```

How do I extract useful statistics from my grouped data?



How do I extract useful statistics from my grouped data?



```
■ output = splitapply(@min, mytable(:, 'D'), Group)
```