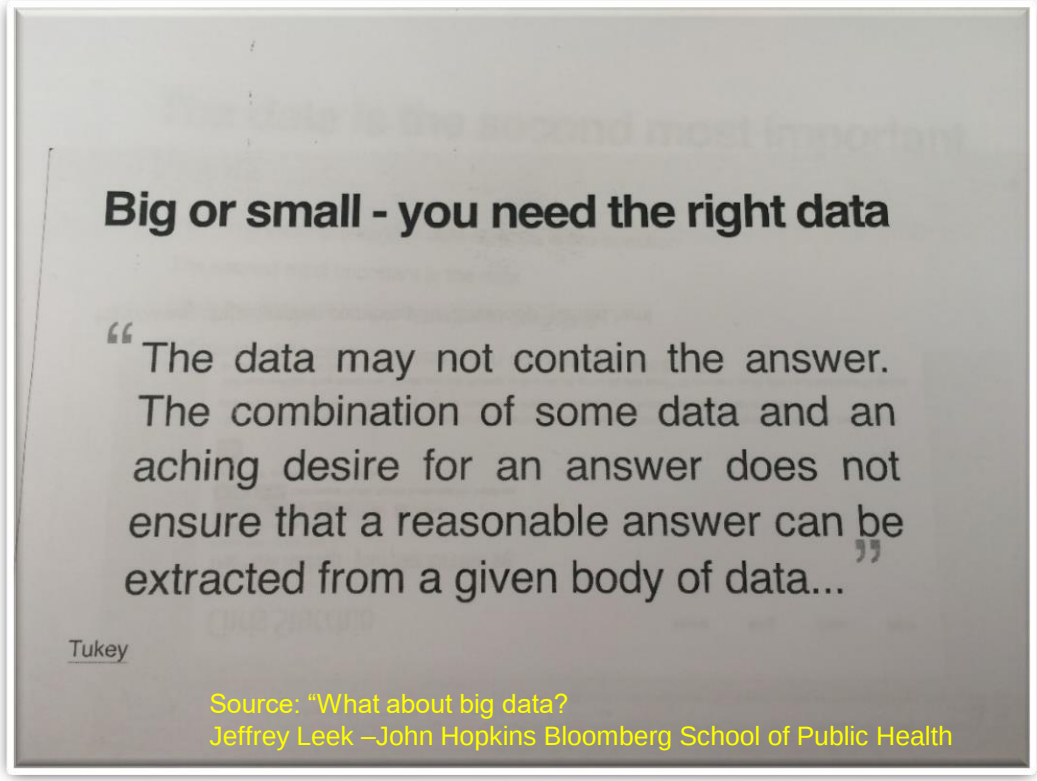
MATLAB Data Type
Esempi di manipolazione di dati testuali

Francesca Perino
Application Engineering Team

© 2017 The MathWorks, Inc.

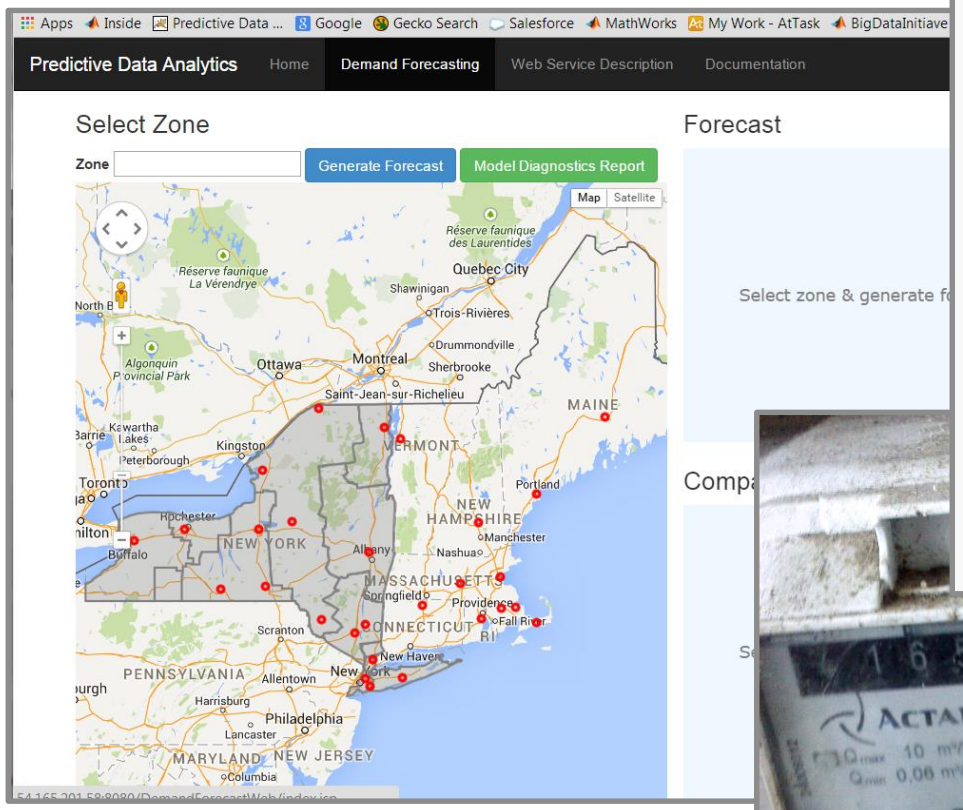
Takeaways

- Different data type
- Combine text with other data sources to take advantage of all your data

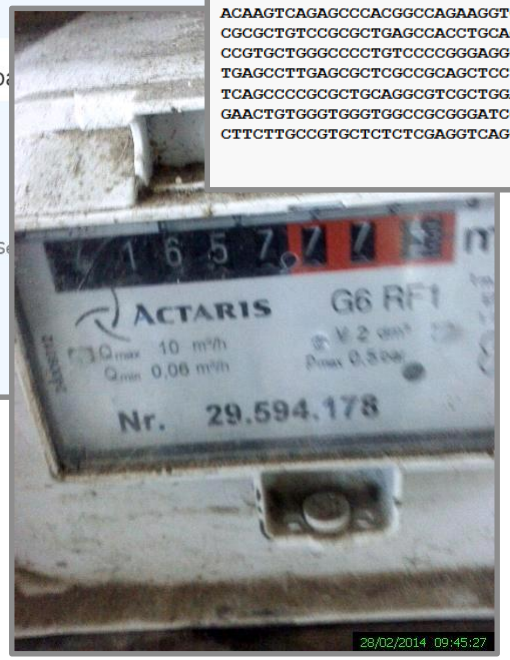


Source: "What about big data?
Jeffrey Leek –John Hopkins Bloomberg School of Public Health

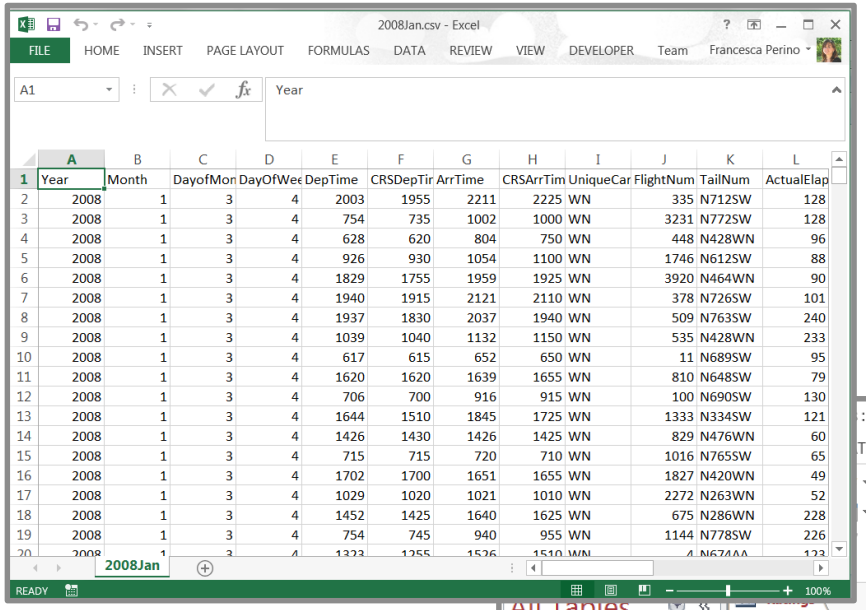
What do data look like?




```
>HSBGPG Human gene for bone gla protein (BGP)
GGCAGATTCCCCCTAGACCCGCCCGCACCATGGTCAGGCATGCCCCTCCTCATCGCTGGGCACAGCCCAGAGGGT
ATAAACAGTGCTGGAGGCTGGCGGGGCAGGCCAGCTGAGTCCTGAGCAGCAGCCCAGCGCAGCCACCGAGACACC
ATGAGAGCCCTCACACTCCTCGCCCTATTGGCCCTGGCCGCACTTTGCATCGCTGGCCAGGCAGGTGAGTGCCCC
CACCTCCCCTCAGGCCGCATTGCAGTGGGGGCTGAGAGGAGGAAGCACCATGGCCCACCTCTTCTCACCCCTTTG
GCTGGCAGTCCCTTTGCAGTCTAACCACCTTGTTGCAGGCTCAATCCATTTGCCCCAGCTCTGCCCTTGCAGAGG
GAGAGGAGGGAAGAGCAAGCTGCCCGAGACGCAGGGGAAGGAGGATGAGGGCCCTGGGGATGAGCTGGGGTGAAC
CAGGCTCCCTTTCCTTTGCAGGTGCGAAGCCCAGCGGTGCAGAGTCCAGCAAAGGTGCAGGTATGAGGATGGACC
TGATGGGTTCCTGGACCCTCCCCTCTCACCCTGGTCCCTCAGTCTCATTCCCCCACTCCTGCCACCTCCTGTCTG
GCCATCAGGAAGGCCAGCCTGCTCCCCACCTGATCCTCCCAAACCCAGAGCCACCTGATGCCTGCCCCTCTGCTC
CACAGCCTTTGTGTCCAAGCAGGAGGGCAGCGAGGTAGTGAAGAGACCCAGGCGCTACCTGTATCAATGGCTGGG
GTGAGAGAAAAGGCAGAGCTGGGCCAAGGCCCTGCCTCTCCGGGATGGTCTGTGGGGGAGCTGCAGCAGGGAGTG
GCCTCTCTGGGTTGTGGTGGGGGTACAGGCAGCCTGCCCTGGTGGGCACCCTGGAGCCCCATGTGTAGGGAGAGG
AGGGATGGGCATTTTGCACGGGGGCTGATGCCACCACGTCGGGTGTCTCAGAGCCCCAGTCCCCTACCCGGATCC
CCTGGAGCCCAGGAGGGAGGTGTGTGAGCTCAATCCGGACTGTGACGAGTTGGCTGACCACATCGGCTTTCAGGA
GGCCTATCGGCGCTTCTACGGCCCGGTCTAGGGTGTCGCTCTGCTGGCCTGGCCGGCAACCCCAGTTCTGCTCCT
CTCCAGGCACCCTTCTTTCCTCTTCCCCTTGCCCTTGCCCTGACCTCCCAGCCCTATGGATGTGGGGTCCCCATC
ATCCCAGCTGCTCCCAAATAAACTCCAGAAG
>HSGLTH1 Human theta 1-globin gene
CCACTGCACTCACCGCACCCGGCCAATTTTTGTGTTTTTAGTAGAGACTAAATACCATATAGTGAACACCTAAGA
CGGGGGGCCTTGGATCCAGGGCGATTCAGAGGGCCCCGGTCGGAGCTGTCGGAGATTGAGCGCGCGCGGTCCCGG
GATCTCCGACGAGGCCCTGGACCCCCGGGCGGCGAAGCTGCGGCGCGGCGCCCCCTGGAGGCCGCGGGACCCCTG
GCCGGTCCGCGCAGGCGCAGCGGGGTCGCAGGGCGCGGCGGGTTCCAGCGCGGGGATGGCGCTGTCCGCGGAGGA
CCGGGCGCTGGTGCGCGCCCTGTGGAAGAAGCTGGGCAGCAACGTCGGCGTCTACACGACAGAGGCCCTGGAAAG
GTGCGGCAGGCTGGGCGCCCCCGCCCCCAGGGGCCCTCCCTCCCCAAGCCCCCCGGACGCGCCTCACCCACGTTC
CTCTCGCAGGACCTTCCTGGCTTTCCCCGCCACGAAGACCTACTTCTCCCACCTGGACCTGAGCCCCGGCTCCTC
ACAAGTCAGAGCCCACGGCCAGAAGGTGGCGGACGCGCTGAGCCTCGCCGTGGAGCGCCTGGACGACCTACCCCA
CGCGCTGTCCGCGCTGAGCCACCTGCACGCGTGCCAGCTGCGAGTGGACCCGGCCAGCTTCCAGGTGAGCGGCTG
CCGTGCTGGGCCCCTGTCCCCGGGAGGGCCCCGGCGGGGTGGGTGCGGGGGGCGTGCGGGGCGGGTGCAGGCGAG
TGAGCCTTGAGCGCTCGCCGCAGCTCCTGGGCCACTGCCTGCTGGTAACCCTCGCCCGGCACTACCCCGGAGACT
TCAGCCCCGCGCTGCAGGCGTCGCTGGACAAGTTCCTGAGCCACGTTATCTCGGCGCTGGTTTCCGAGTACCGCT
GAACTGTGGGTGGGTGGCCGCGGGATCCCCAGGCGACCTTCCCCGTGTTTGAGTAAAGCCTCTCCCAGGAGCAGC
CTTCTTGCCGTGCTCTCTCGAGGTCAGGACGCGAGAGGAAGGCGC
```

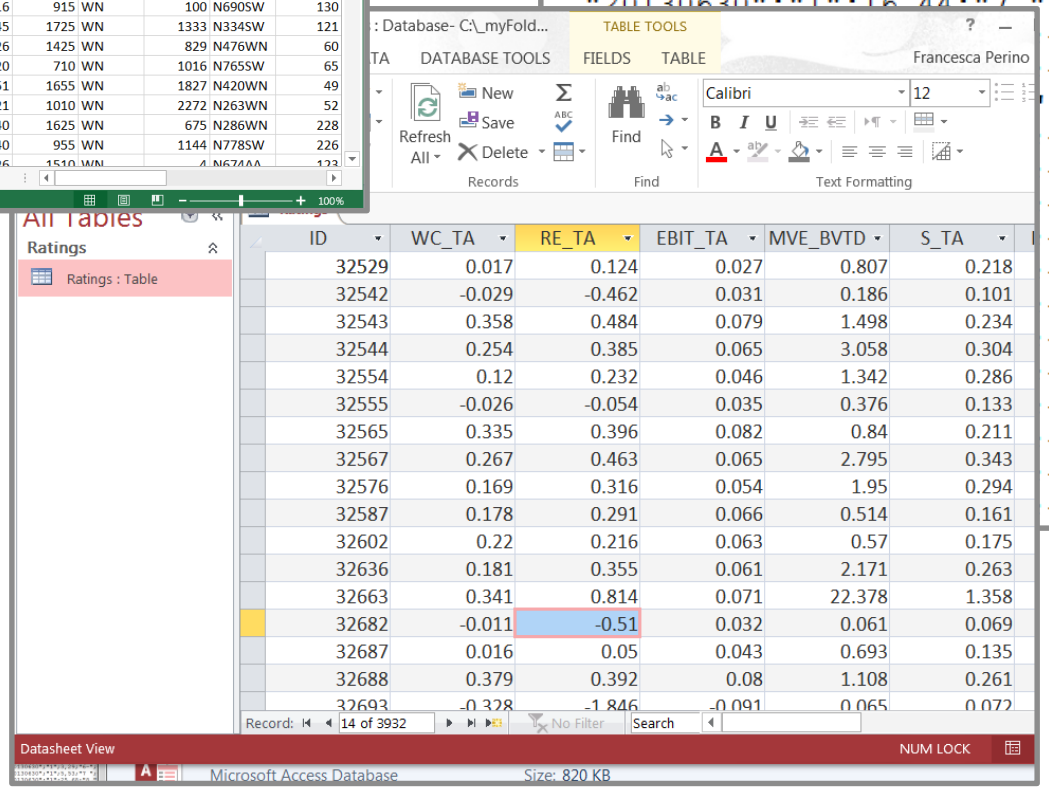


What do data look like?



	A	B	C	D	E	F	G	H	I	J	K	L
1	Year	Month	DayofMon	DayOfWee	DepTime	CRSDepTir	ArrTime	CRSArrTim	UniqueCar	FlightNum	TailNum	ActualElap
2	2008	1	3	4	2003	1955	2211	2225	WN	335	N712SW	128
3	2008	1	3	4	754	735	1002	1000	WN	3231	N772SW	128
4	2008	1	3	4	628	620	804	750	WN	448	N428WN	96
5	2008	1	3	4	926	930	1054	1100	WN	1746	N612SW	88
6	2008	1	3	4	1829	1755	1959	1925	WN	3920	N464WN	90
7	2008	1	3	4	1940	1915	2121	2110	WN	378	N726SW	101
8	2008	1	3	4	1937	1830	2037	1940	WN	509	N763SW	240
9	2008	1	3	4	1039	1040	1132	1150	WN	535	N428WN	233
10	2008	1	3	4	617	615	652	650	WN	11	N689SW	95
11	2008	1	3	4	1620	1620	1639	1655	WN	810	N648SW	79
12	2008	1	3	4	706	700	916	915	WN	100	N690SW	130
13	2008	1	3	4	1644	1510	1845	1725	WN	1333	N334SW	121
14	2008	1	3	4	1426	1430	1426	1425	WN	829	N476WN	60
15	2008	1	3	4	715	715	720	710	WN	1016	N765SW	65
16	2008	1	3	4	1702	1700	1651	1655	WN	1827	N420WN	49
17	2008	1	3	4	1029	1020	1021	1010	WN	2272	N263WN	52
18	2008	1	3	4	1452	1425	1640	1625	WN	675	N286WN	228
19	2008	1	3	4	754	745	940	955	WN	1144	N778SW	226

```
"20130630";"1";2,93;"6 ";-3,49
"20130630";"1";0,72;"5+";-4,91
"20130630";"1";4,84;"6-";-2,97
"20130630";"1";0,17;"4 ";-6,33
"20130630";"1";1,34;"5-";-4,29
"20130630";"1";0,81;"5 ";-4,80
"20130630";"1";12,25;"7 ";-1,96
"20130630";"1";1,60;"6+";-4,11
"20130630";"1";0,61;"5+";-5,07
"20130630";"1";0,33;"4-";-5,68
"20130630";"1";0,23;"4 ";-6,05
"20130630";"1";13,07;"7 ";-1,89
"20130630";"1";16,44;"7 ";-1,62
-3,38
-2,83
;-1,06
-5,05
-2,78
-5,19
-4,97
-5,28
-4,62
-4,88
-3,60
-3,46
-4,83
-6,33
-5,52
```



ID	WC_TA	RE_TA	EBIT_TA	MVE_BVTD	S_TA
32529	0.017	0.124	0.027	0.807	0.218
32542	-0.029	-0.462	0.031	0.186	0.101
32543	0.358	0.484	0.079	1.498	0.234
32544	0.254	0.385	0.065	3.058	0.304
32554	0.12	0.232	0.046	1.342	0.286
32555	-0.026	-0.054	0.035	0.376	0.133
32565	0.335	0.396	0.082	0.84	0.211
32567	0.267	0.463	0.065	2.795	0.343
32576	0.169	0.316	0.054	1.95	0.294
32587	0.178	0.291	0.066	0.514	0.161
32602	0.22	0.216	0.063	0.57	0.175
32636	0.181	0.355	0.061	2.171	0.263
32663	0.341	0.814	0.071	22.378	1.358
32682	-0.011	-0.51	0.032	0.061	0.069
32687	0.016	0.05	0.043	0.693	0.135
32688	0.379	0.392	0.08	1.108	0.261
32693	-0.328	-1.846	-0.091	0.065	0.072

Access Multiple Files to Import Specific Columns

```
tic
for ii = length(D):-1:1   % Takes around 2 minutes to process
    fullname        = [Location '\' D(ii).name];
    [~, name, ~]    = fileparts(fullname); % Strips off ".xls"
    Engine          = importExcelData(fullname);
    [val, index]    = min(Engine.BSFC);
    EngineMin       = Engine(index,:);
    EngineMin.Name  = categorical({name}); % Add the name
    AllEngines(ii,:) = EngineMin;
end
toc
```

```
%% Import the data
data = [xlsread(workbookFile, sheetName, sprintf('A%d:A%d',startRow(1),endRow(1))),...
    xlsread(workbookFile, sheetName, sprintf('G%d:G%d',startRow(1),endRow(1))),...
    xlsread(workbookFile, sheetName, sprintf('U%d:U%d',startRow(1),endRow(1)))];
for block=2:length(startRow)
    tmpDataBlock = [xlsread(workbookFile, sheetName, sprintf('A%d:A%d',startRow(block),endRow(block))),...
        xlsread(workbookFile, sheetName, sprintf('G%d:G%d',startRow(block),endRow(block))),...
        xlsread(workbookFile, sheetName, sprintf('U%d:U%d',startRow(block),endRow(block)))];
    data = [data;tmpDataBlock]; %#ok<AGROW>
end
```

Access Multiple Files to Import Specific Columns

CONTENTS

spreadsheetDatastore

Create SpreadsheetDatastore object for collections of spreadsheet data

Syntax

```
ssds = spreadsheetDatastore(location)
ssds = spreadsheetDatastore(location,Name,Value)
```

Description

`ssds = spreadsheetDatastore(location)` creates a datastore from the collection of data specified by `locatio`
collections of data that are too large to fit in memory. After creating a `SpreadsheetDatstore` object, you can read
ways. See Using SpreadsheetDatastore Objects for more information.

`ssds = spreadsheetDatastore(location,Name,Value)` specifies additional parameters for `ssds` using one or
For example, `spreadsheetDatastore(location,'FileExtentions',{'.xlsx','.xls'})` specifies which files
depending on the file extensions.

```
%% Method #2:
% we use |spreadsheetDatastore|
tic
dsEngine = ...
    spreadsheetDatastore('data','FileExtensions',{'.xls','.xlsm'});
dsEngine.ReadSize = 'file';
%%
dsEngine.SelectedVariableNames = {'SPEED','LOAD','BSFC'};
%%
ii = 1;
while hasdata(dsEngine)
    Engine            = read(dsEngine);
    [val, index]      = min(Engine.BSFC);
    EngineMin         = Engine(index,:);
    [~,name]          = fileparts(dsEngine.Files{ii});
    EngineMin.Name    = categorical({name}); % Add the name
    AllEngines(ii,:)  = EngineMin;
    ii = ii+1;
end
toc
```

Fraud Detection

- Detecting when people intentionally act secretly to deprive another of something of value
- Typically involves processing large amounts of data from many sources, such us
  - Returns Forensics
  - Linguistic Based Cues

Structured Hedge Fund Products

Certificate

NPB New Private Bank Ltd. offers together with the Issuer, Nomura Bank International plc., a leveraged Certificate (3x) with exposure to the well-established Fairfield Sentry Ltd. Fund. Fairfield Sentry Ltd. has a track record of over 15 years with an annual return of 4-6% averaged above the 1-month USD-LIBOR, and a maximum drawdown of -0.64% on a monthly basis. The goal of the Certificate is a significant yield improvement despite a slightly extended volatility.

Strategy

Primarily split-strike conversion strategy: Fairfield Sentry Ltd. invests in shares of the S&P100 Index which are hedged by put options. Additionally, the Investment Manager sells call options on the relevant index.

Liquidity

The Certificate offers daily liquidity.

Security

- Fairfield Sentry Ltd. has a low volatility of 2.58%
- Over 92% positive months (only 13 losing months)
- Widely hedged risk of the equity exposure in the Fund

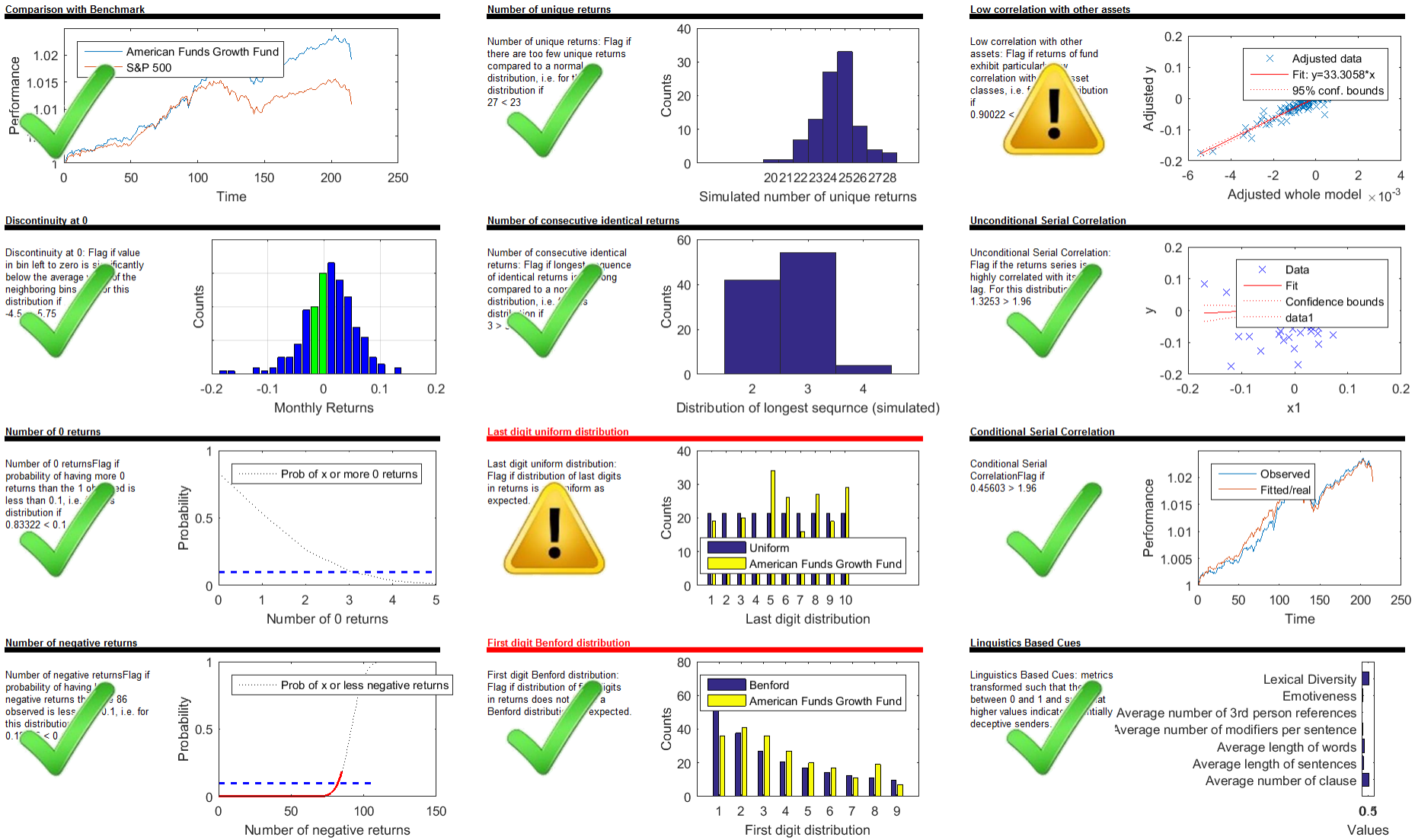
Performance Fairfield Sentry Ltd.

- 15-year track record of the Fund (11.04% p.a. in USD)
- Low correlation to equities, bonds, hedge funds and other asset classes

													Fairfield Sentry Ltd.	Certificate[4]
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Year	Year
1990												2.77%	2.77%	6.79%
1991	3.01%	1.40%	0.52%	1.32%	1.82%	0.30%	1.98%	1.00%	0.73%	2.75%	0.01%	1.56%	17.64%	38.52%
1992	0.42%	2.72%	0.94%	2.79%	-0.27%	1.22%	-0.09%	0.86%	0.33%	1.33%	1.36%	1.36%	13.72%	30.39%
1993	-0.09%	1.86%	1.79%	-0.01%	1.65%	0.79%	0.02%	1.71%	0.28%	1.71%	0.19%	0.39%	10.75%	21.96%
1994	2.11%	-0.44%	1.45%	1.75%	0.44%	0.23%	1.71%	0.35%	0.75%	1.81%	-0.64%	0.60%	10.57%	18.05%
1995	0.85%	0.69%	0.78%	1.82%	1.65%	0.43%	1.02%	-0.24%	1.63%	1.53%	0.44%	1.03%	12.04%	19.67%
1996	1.42%	0.66%	1.16%	0.57%	1.34%	0.15%	1.86%	0.20%	1.16%	1.03%	1.51%	0.41%	12.08%	21.09%
1997	2.38%	0.67%	0.80%	1.10%	0.57%	1.28%	0.68%	0.28%	2.32%	0.49%	1.49%	0.36%	13.10%	23.83%
1998	0.85%	1.23%	1.68%	0.36%	1.66%	1.22%	0.76%	0.21%	0.98%	1.86%	0.78%	0.26%	12.52%	22.32%
1999	1.99%	0.11%	2.22%	0.29%	1.45%	1.70%	0.36%	0.87%	0.66%	1.05%	1.54%	0.32%	13.29%	25.27%
2000	2.14%	0.13%	1.77%	0.27%	1.30%	0.73%	0.58%	1.26%	0.18%	0.86%	0.62%	0.36%	10.67%	14.28%
2001	2.14%	0.08%	1.07%	1.26%	0.26%	0.17%	0.38%	0.94%	0.66%	1.22%	1.14%	0.12%	9.82%	17.79%
2002	-0.04%	0.53%	0.39%	1.09%	2.05%	0.19%	3.29%	-0.14%	0.06%	0.66%	0.10%	0.00%	8.43%	17.74%
2003	-0.35%	-0.05%	1.85%	0.03%	0.90%	0.93%	1.37%	0.16%	0.86%	1.26%	-0.14%	0.25%	7.27%	15.48%
2004	0.88%	0.44%	-0.01%	0.37%	0.59%	1.21%	0.02%	1.26%	0.46%	0.03%	0.79%	0.24%	6.44%	12.20%
2005	0.51%	0.37%	0.85%	0.14%	0.63%	0.46%	0.13%	0.16%	0.89%	1.61%	0.75%	0.54%	7.26%	10.50%
2006	0.70%	0.20%	1.31%	0.94%	0.70%	0.51%	1.06%	0.77%	0.68%	0.42%	0.86%	0.86%	9.38%	13.59%
2007	0.29%	-0.11%	1.64%	0.98%	0.81%	0.34%	0.17%	0.31%	0.97%	0.46%	1.04%	0.23%	7.34%	7.58%
2008	0.63%	0.06%	0.18%	0.93%	0.81%	-0.06%	0.72%	0.71%	0.50%	-0.06%			4.50%	5.40%

http://nakedshorts.typepad.com/files/madoff_fairfieldsentry3x.pdf

The Desired Report



Workflow

Access and Explore Data

- Import heterogeneous data
  - `Excel files`
  - `Datafeeder (e.g. Bloomberg,Yahoo!Finance)`
  - `PDF file`

Preprocess Data

- Merge data and align timestamps
- Screening Funds Data for Signs of Fraud
  - Quantitative Analysis
  - Textual Analysis

Develop Models

- Fraudulent and non-fraudulent funds classifier

Results Sharing

- Report that is automatically generated and pre-flags potential suspicious behavior

Data Type

Numeric Types
Integer and floating-point data

Characters and Strings
Text in character arrays and string arrays

Dates and Time
Arrays of date and time values that can be displayed in different formats

Categorical Arrays
Arrays of qualitative data with values from a finite set of discrete, nonnumeric data

Tables
Arrays in tabular form whose named columns can have different types

Timetables
Time-stamped data in tabular form

Structures
Arrays with named fields that can contain data of varying types and sizes

Cell Arrays
Arrays that can contain data of varying types and sizes

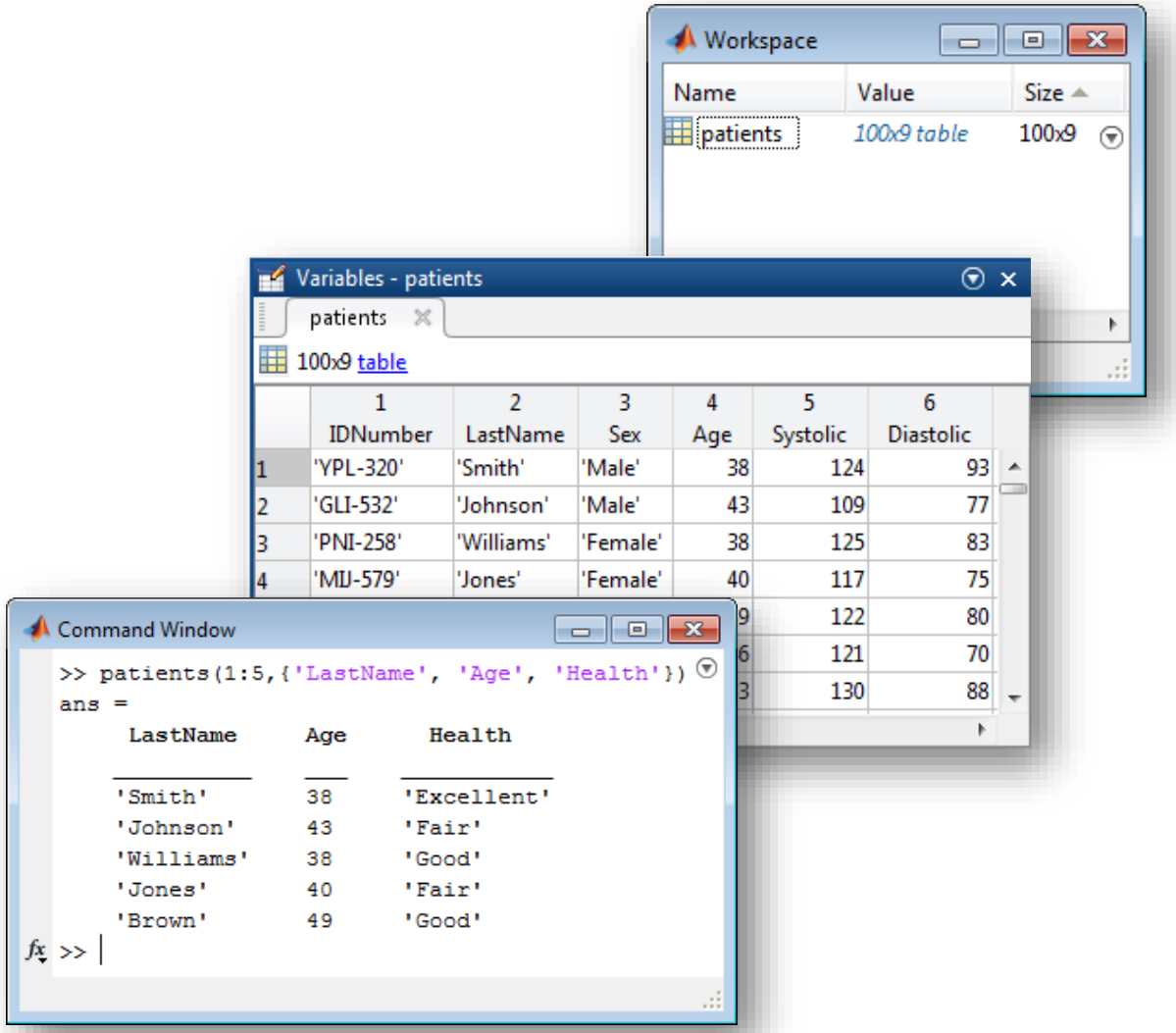
Function Handles
Variables that allow you to invoke a function indirectly

Map Containers
Objects with keys that index to values, where keys need not be integers

Time Series
Data vectors sampled over time

table

- For mixed-type tabular data
  - Holds both data and metadata
- Supports flexible indexing
- Built-in functionality (merge, sort, etc.)
- Database-like functionality
  - Join
  - Stack / Unstack



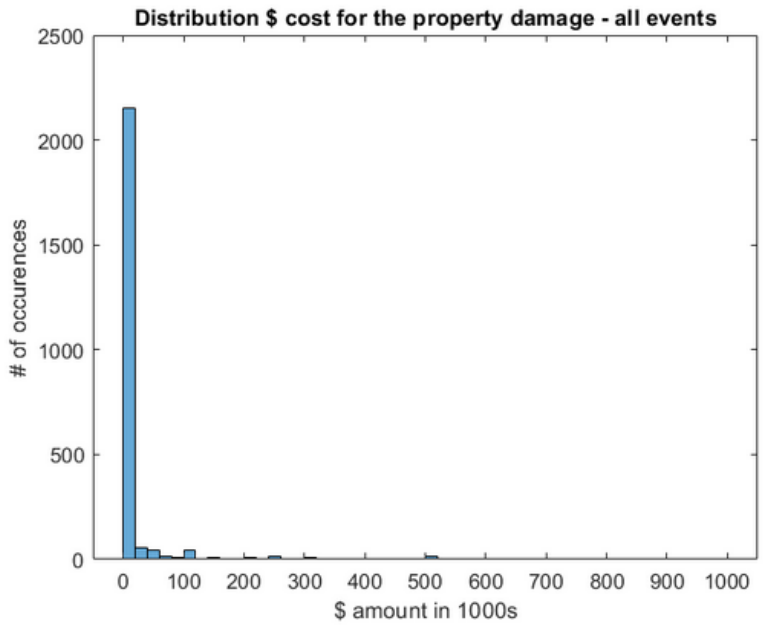
`string`

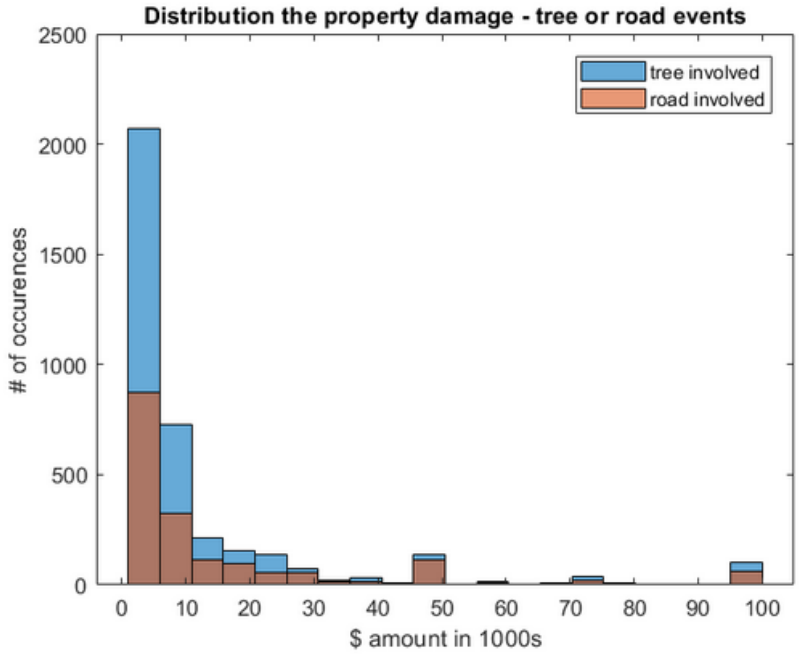
- MATLAB char arrays are arrays of characters
- MATLAB string arrays are arrays of strings
- Library of new functions for manipulating strings

`compose`	Converts data into formatted string arrays
`erase`	Remove substring from string elements
`extractBetween`	Extract bounded substrings
`insertAfter`	Insert substring after a specified position
`join`	Append elements of a string array together
`replace`	Find and replace substring
`replaceBetween`	Replace bounded substrings
`reverse`	Reverse the order of characters
`split`	Split string at delimiter

Weather Report

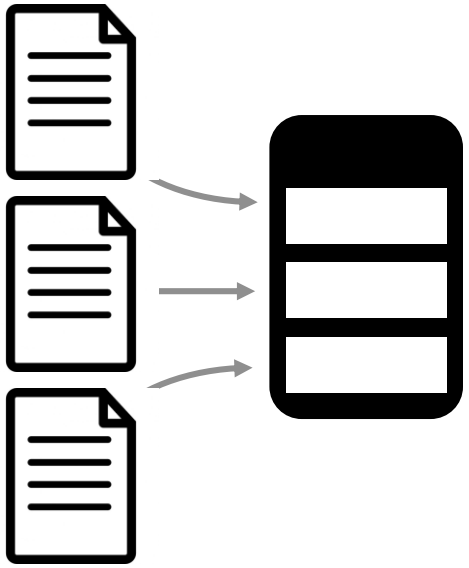
	Time	event_id	state	event_type	damage_prope...	damage_crops	begin_lat	begin_lon	end_lat	end_lon	event_narrative	storm_duration	begin_day	end_day	year	end_timestamp
1	22-Jul-201...	644330	"MISSISSI...	"Thunderst...	""	"0.00K"	34.1398	-88.6298	34.1220	-88.6260	"Large tree...	"00:05:00"	22	22	2016	22-Jul-001...
2	15-Jul-201...	651823	"SOUTH C...	"Heavy Rain"	"2.00K"	"0.00K"	34.9400	-81.0300	34.9400	-81.0300	"One to two...	"00:00:00"	15	15	2016	15-Jul-001...
3	15-Jul-201...	651825	"SOUTH C...	"Thunderst...	"0.00K"	"0.00K"	35.0100	-80.9300	35.0100	-80.9300	"NWS Colu...	"00:00:00"	15	15	2016	15-Jul-001...
4	16-Jul-201...	651828	"NORTH C...	"Thunderst...	"0.00K"	"0.00K"	35.6400	-82.1400	35.6400	-82.1400	"Media rep...	"00:00:00"	16	16	2016	16-Jul-001...
5	15-Jul-201...	643319	"MISSOURI"	"Hail"	""	""	36.4500	-89.9700	36.4500	-89.9700	""	"00:07:00"	15	15	2016	15-Jul-001...
6	15-Jul-201...	643321	"ARKANSAS"	"Thunderst...	""	"0.00K"	35.8500	-90.1000	35.8376	-90.0865	"A few tree ...	"00:09:00"	15	15	2016	15-Jul-001...
7	15-Jul-201...	643431	"TENNESS...	"Thunderst...	"20.00K"	"0.00K"	35.0563	-89.9367	35.0498	-89.9037	"Awning bl...	"00:07:00"	15	15	2016	15-Jul-001...
8	15-Jul-201...	643437	"TENNESS...	"Hail"	""	""	35.3845	-89.7800	35.3845	-89.7800	"Quarter si...	"00:05:00"	15	15	2016	15-Jul-001...





Access and Explore Data → **Preprocess Data** → **Develop Predictive Models**

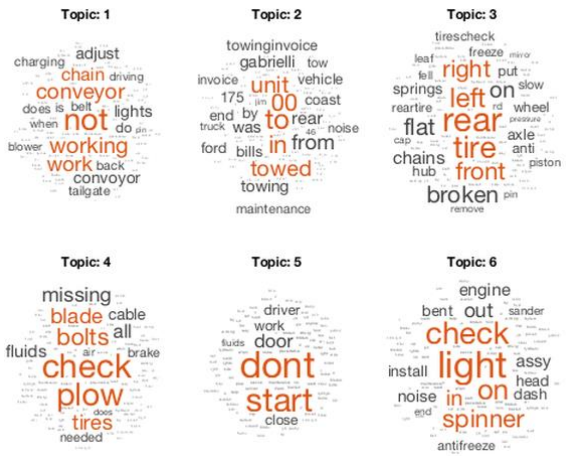
Clean-up Text → **Convert to Numeric**



Media reported two trees blown down along I-40 in the Old Fort area.

media report two tree blown down i40 old fort area

	cat	dog	run	two
doc1	1	0	1	0
doc2	1	1	0	1



- Word Docs
- PDF's
- Text Files

- Stop Words
- Stemming
- Tokenization

- Bag of Words
- TF-IDF
- Word Embeddings

- Latent Dirichlet Allocation
- Latent Semantic Analysis

`String:` *the better way to work with text*

- Manipulate, compare, and store text data efficiently

```
>> "image" + (1:3) + ".png"
   1×3 string array
     "image1.png"   "image2.png"   "image3.png"
```

- Simplified text manipulation functions
  - Example: Check if a string is contained within another string
    - Previously: `if ~isempty(strfind(textdata,"Dog"))`
    - Now: `if contains(textdata,"Dog")`

- Performance improvement
  - Up to 50x faster using **`contains`** with **`string`** than **`strfind`** with **`cellstr`**
  - Up to 2x memory savings using **`string`** over **`cellstr`**

Takeaways

- Different data type
- Combine text with other data sources to take advantage of all your data