

MATLAB EXPO 2017

KOREA

4월 27일, 서울

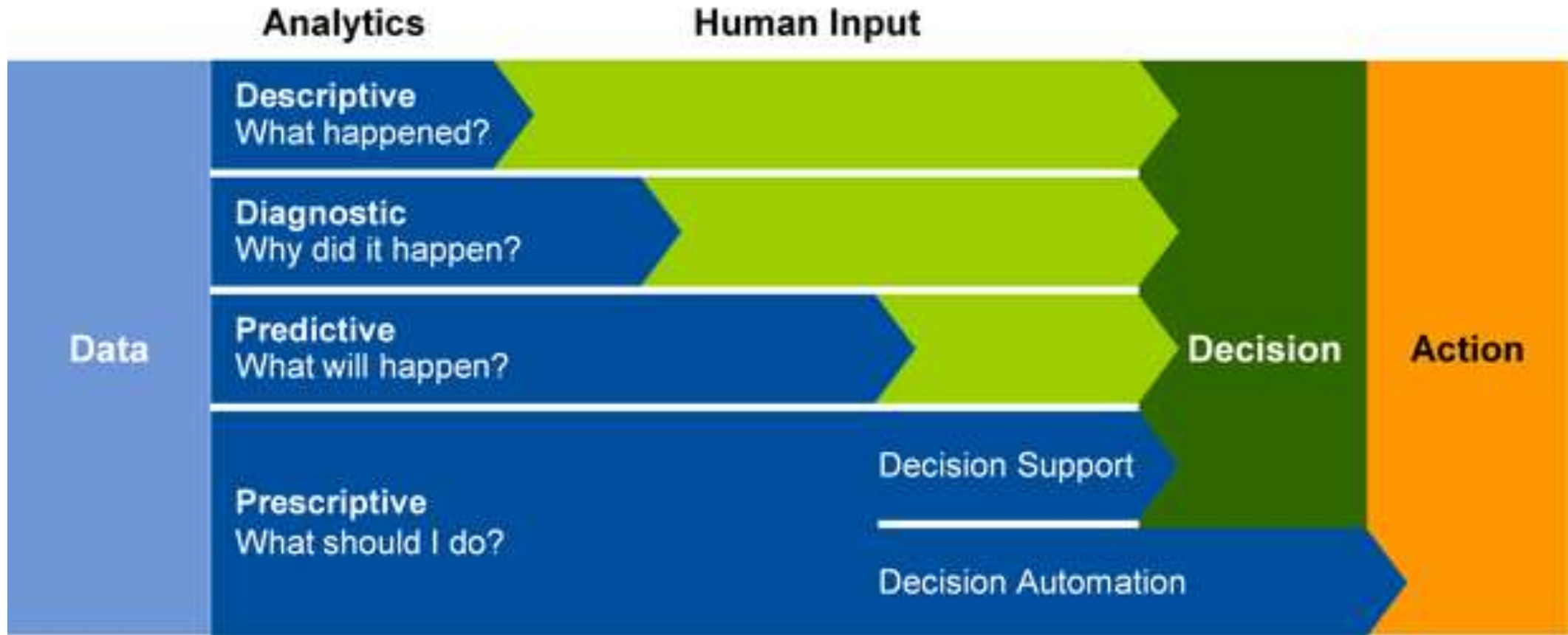
등록 하기 matlabexpo.co.kr

빅데이터 처리 및 머신 러닝 기법

Application Engineer

엄준상 과장

Data Analytics



Turn *large volumes* of complex data into actionable information
source: [Gartner](#)

Data Analytics Workflow

Access and Explore Data

Preprocess Data

Develop Predictive Models

Integrate Analytics with Systems

Files



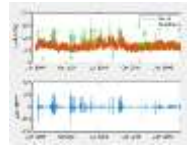
Databases



Sensors



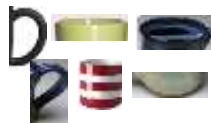
Working with Messy Data



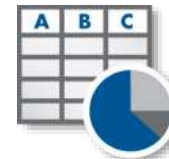
Data Reduction/Transformation



Feature Extraction



Model Creation e.g. Machine Learning



Parameter Optimization



Model Validation



Desktop Apps



Enterprise Scale Systems

MATLAB Excel
.NET C/C++
.exe Java .dll

Embedded Devices and Hardware



Example: Working with Big Data in MATLAB

- **Objective:** Create a model to predict the cost of a taxi ride in New York City
- **Inputs:**
 - Monthly taxi ride log files
 - The local data set is **small** (~20 MB)
 - The full data set is **big** (~25 GB)
- **Approach:**
 - Access Data
 - Preprocess and explore data
 - Develop and validate predictive model (linear fit)
 - Work with subset of data for prototyping
 - Scale to full data set on a cluster



Example: Working with Big Data in MATLAB

tall Arrays for Big Data in MATLAB

Predict Cost of Taxi Ride in New York City

Analyze data from .csv files containing taxi trip information, separated by month. The data set is available from the [City of New York](#).

VendorID	tpep_pickup_datetime	tpep_dropoff_datetime	passenger_count	trip_distance	pickup_longitude	pickup_latitude
2	2015-01-07 07:40:20	2015-01-07 08:04:45	6	9.12	-73.9524536132812	40.78
2	2015-01-21 22:49:50	2015-01-21 23:17:11	6	5.63	-74.0083694458008	40.73
1	2015-01-05 23:04:30	2015-01-05 23:15:00	1	2.9	-73.8632125854492	40.76
1	2015-01-11 22:20:43	2015-01-11 22:23:02	1	0.8	-73.9577560424805	40.76
2	2015-01-24 00:34:59	2015-01-24 00:38:39	1	0.65	-73.9916687011719	40.73
1	2015-01-25 19:09:57	2015-01-25 19:18:02	1	1.5	-73.9983825683594	40.72
1	2015-01-02 23:24:13	2015-01-02 23:27:30	1	1	-73.9963912963867	40.75
2	2015-01-21 06:46:23	2015-01-21 06:47:56	1	0.63	-73.9913635253906	40.77
2	2015-01-23 19:32:33	2015-01-23 19:49:56	1	2.52	-73.988382018043	40.73

Set up execution environment

```
numWorkers = 16;

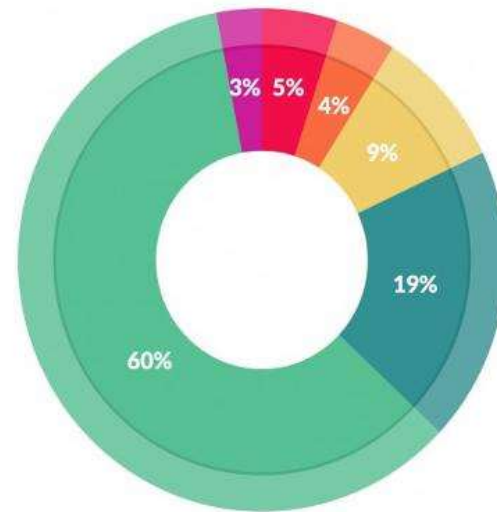
setenv('HADOOP_HOME', '/mathworks/test/hadoop');
setenv('SPARK_HOME', '/mathworks/test/spark');

cluster = parallel.cluster.Hadoop;
cluster.SparkProperties('spark.executor.instances') = num2str(numWorkers);
```

Data Access and Pre-processing – Challenges

Challenges

- Data aggregation
 - Different sources (files, web, etc.)
 - Different types (images, text, audio, etc.)
- Data clean up
 - Poorly formatted files
 - Irregularly sampled data
 - Redundant data, outliers, missing data etc.
- Data specific processing
 - Signals: Smoothing, resampling, denoising, Wavelet transforms, etc.
 - Images: Image registration, morphological filtering, deblurring, etc.
- Dealing with out of memory data (big data)

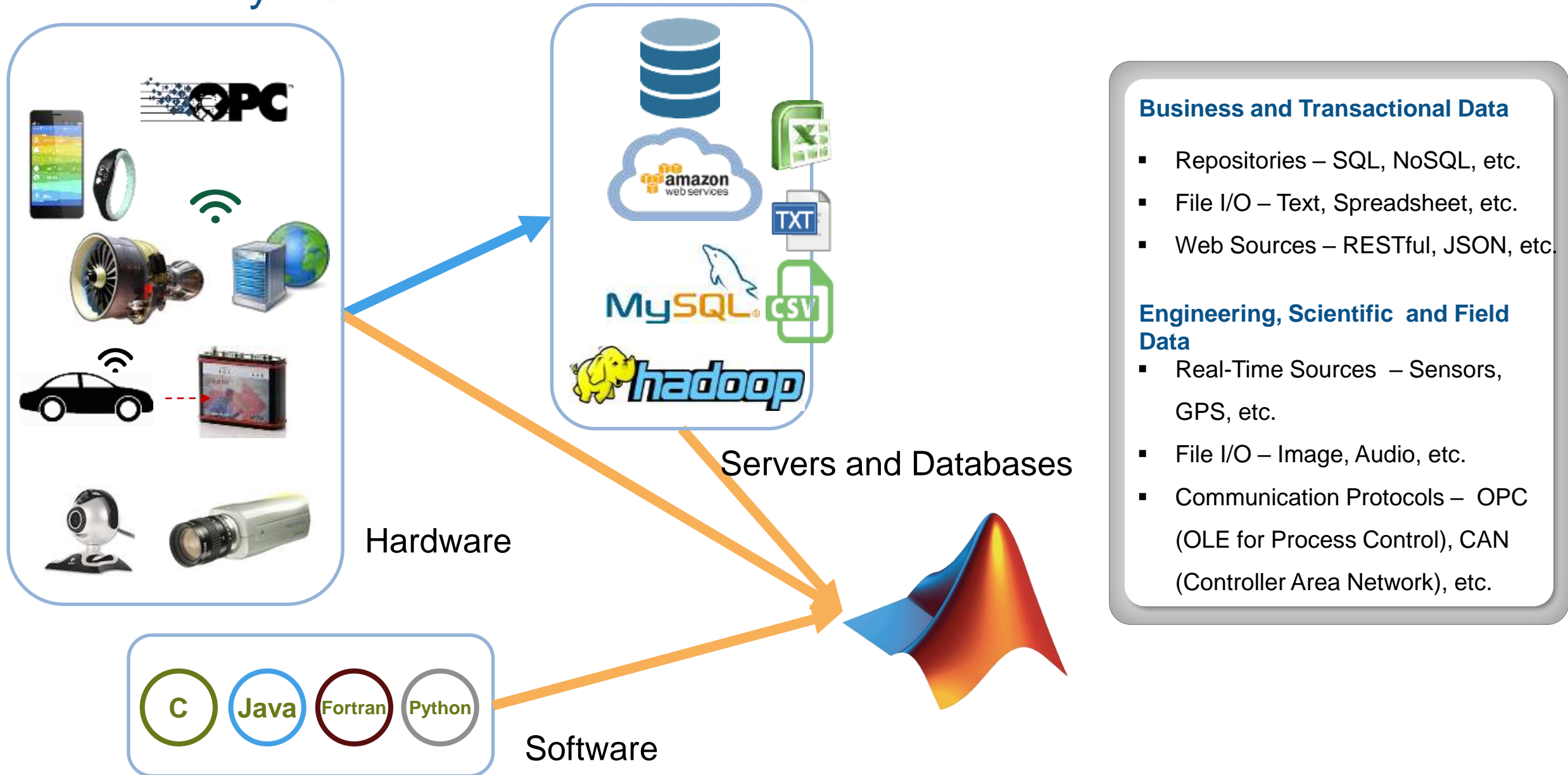


What data scientists spend the most time doing

- Building training sets: 3%
- Cleaning and organizing data: 60%
- Collecting data sets: 19%
- Mining data for patterns: 9%
- Refining algorithms: 4%
- Other: 5%

Data preparation accounts for about **80%** of the work of data scientists - Forbes

Data Analytics Workflow: Data Access



Data Analytics Workflow: Big Data Access and Pre-processing

www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

Data Analytics - Home Discover MATLAB & CRE - Home MATLAB Fleet Data Analysis

2016

2015

January	Yellow	Green	FHV
February	Yellow	Green	FHV
March	Yellow	Green	FHV
April	Yellow	Green	FHV
May	Yellow	Green	FHV
June	Yellow	Green	FHV
July	Yellow	Green	FHV
August	Yellow	Green	FHV
September	Yellow	Green	FHV
October	Yellow	Green	FHV
November	Yellow	Green	FHV
December	Yellow	Green	FHV

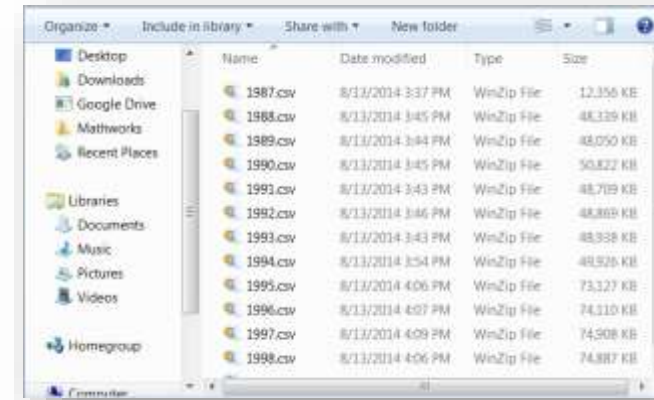
2014

Download 2015 Taxi Data from Web using 'websave' in parallel

```
parfor i=1:12
    fileName = ['taxiData2015_', num2str(i)]
    url      = ['https://s3.amazonaws.com/nyc-tlc/trip+data/yellow_tripdata_2016-0', num2str(i), '.csv']
    websave(fileName, url)
end
```

Big Data in Recent Releases

- **datastore**
 - Tabular text files
 - Images
 - Excel spreadsheets
 - (SQL) Databases
 - HDFS (Hadoop)
 - S3 (Amazon Web Services)
- **MATLAB MapReduce**
 - Scales from Desktop to Hadoop



```
>> preview(ds)
ans =
```

Year	Month	DayofMonth	DayOfWeek
1987	10	21	3
1987	10	26	1
1987	10	23	5
1987	10	23	5

```
airdata = datastore('*.csv');
airdata.SelectedVariables = {'Distance', 'ArrDelay'};

data = read(airdata);
```

Data Analytics Workflow: Big Data Access and Pre-processing

www.nyc.gov/html/tlc/html/about/trip_record_data.shtml

Data Analytics - Home Discover MATLAB & CRE - Home MATLAB Fleet Data Analysis

2016

2015

January	Yellow	Green	FHV
February	Yellow	Green	FHV
March	Yellow	Green	FHV
April	Yellow	Green	FHV
May	Yellow	Green	FHV
June	Yellow	Green	FHV
July	Yellow	Green	FHV
August	Yellow	Green	FHV

above

Create a datastore to represent the data

A `datastore` is a repository for data and allows you to read part of the data, memory.

```
fileLoc = fullfile('taxiData','*.csv');
ds = datastore(fileLoc);
preview(ds)
```

Select variables of interest and give them more intuitive labels.

```
vars = [2:3,5,12:13,16,19];
ds.VariableNames(vars) = {'Pickup','Dropoff','TripDistance',...
    'PaymentType','Fare','Tip','Total'};
ds.SelectedVariableNames = ds.VariableNames(vars);
```

Connect to the database application

```
conn = database('taxiDemo', 'root', 'matlab', ...
    'Vendor', 'MYSQL', ...
    'Server', 'localhost', ...
    'PortNumber', 3306);
```

Create a database datastore and import data of interest

```
sqlquery = ['select pickuptime, dropofftime, trip_distance,...
    'payment_type, fare_amount from taxiData'];
ds = databaseDatastore(conn,sqlquery, 'ReadSize',100000);
```

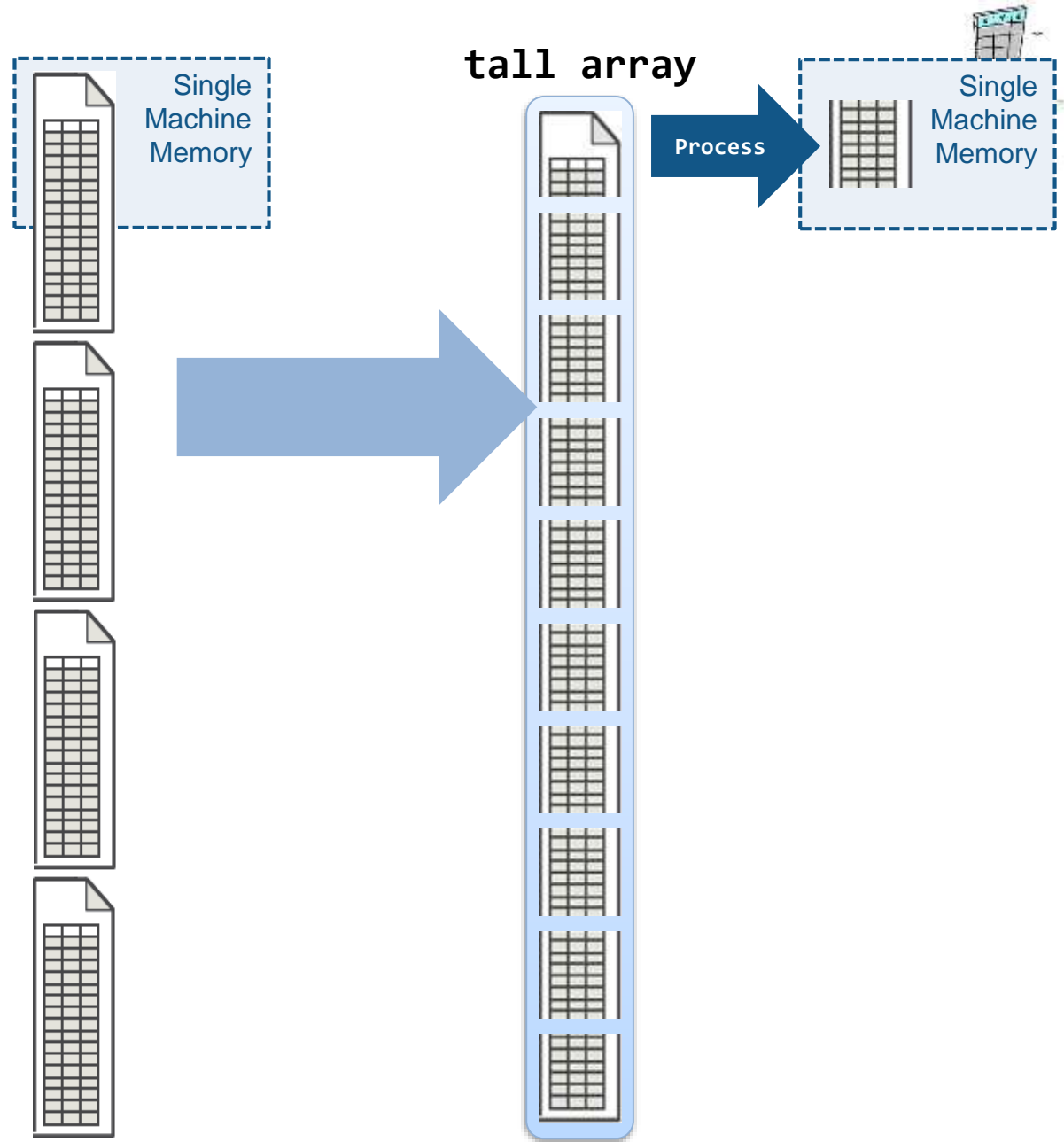
tall arrays in R2016b

- New data type designed for data that doesn't fit into memory
- Lots of observations (hence "tall")
- Looks like a normal MATLAB array
 - Supports numeric types, tables, datetimes, strings, etc...
 - Supports several hundred functions for basic math, stats, indexing, etc.
 - **Statistics and Machine Learning Toolbox** support (clustering, classification, etc.)



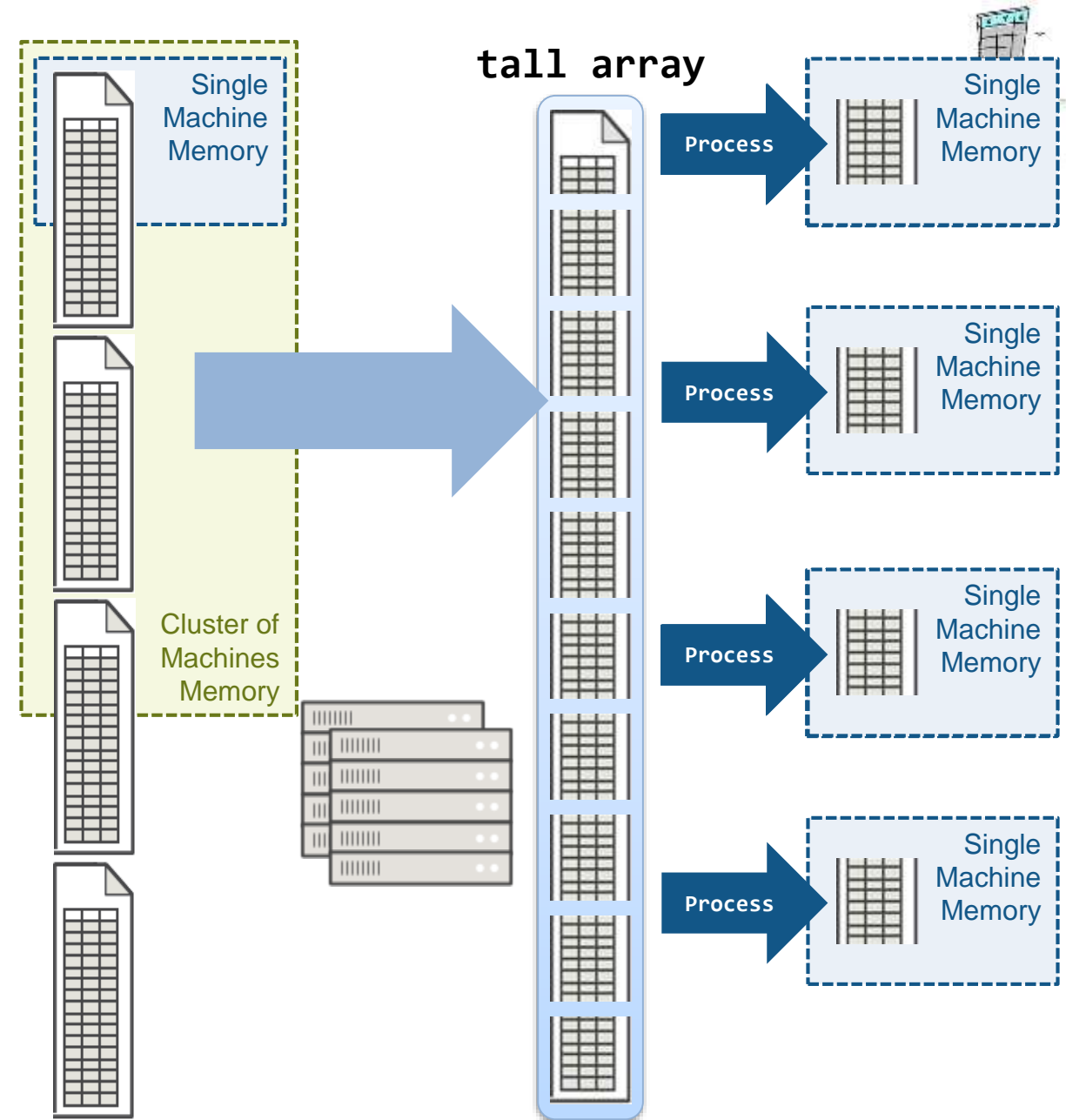
tall arrays R2016b

- Automatically breaks data up into small “chunks” that fit in memory
- Tall arrays scan through the dataset one “chunk” at a time
- Processing code for tall arrays is the same as ordinary arrays

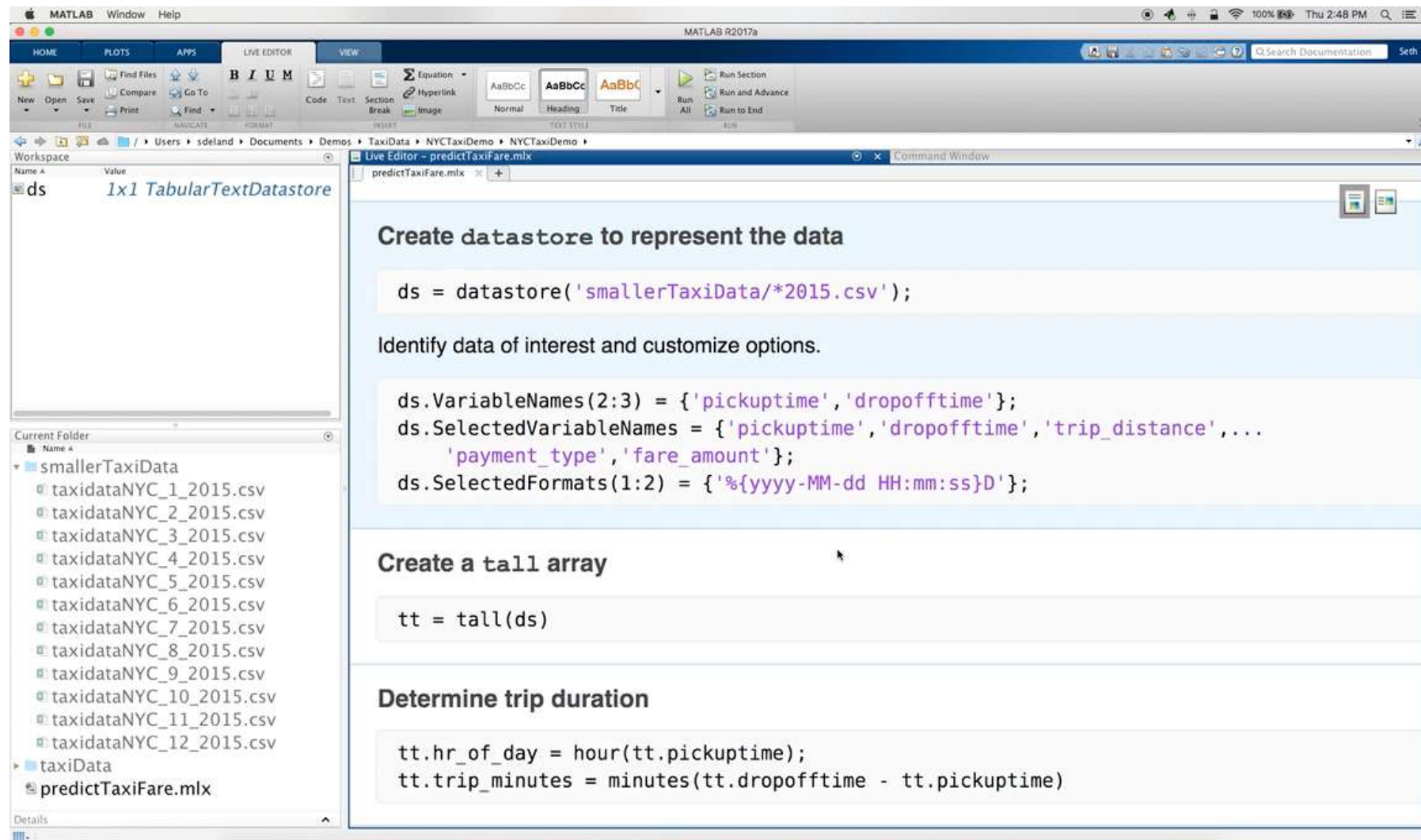


tall arrays R2016b

- With Parallel Computing Toolbox, process several “chunks” at once
- Can scale up to clusters with MATLAB Distributed Computing Server



Demo: Working with Tall Arrays



The image shows the MATLAB Live Editor interface. The top toolbar includes options for HOME, PLOTS, APPS, LIVE EDITOR, and VIEW. The workspace on the left shows a variable 'ds' of type '1x1 TabularTextDatastore'. The current folder on the left shows a directory structure with 'smallerTaxiData' containing 12 CSV files and a 'taxiData' folder. The main editor area contains the following code:

```
Create datastore to represent the data

ds = datastore('smallerTaxiData/*2015.csv');

Identify data of interest and customize options.

ds.VariableNames(2:3) = {'pickuptime', 'dropofftime'};
ds.SelectedVariableNames = {'pickuptime', 'dropofftime', 'trip_distance', ...
    'payment_type', 'fare_amount'};
ds.SelectedFormats(1:2) = {'%{yyyy-MM-dd HH:mm:ss}D'};

Create a tall array

tt = tall(ds)

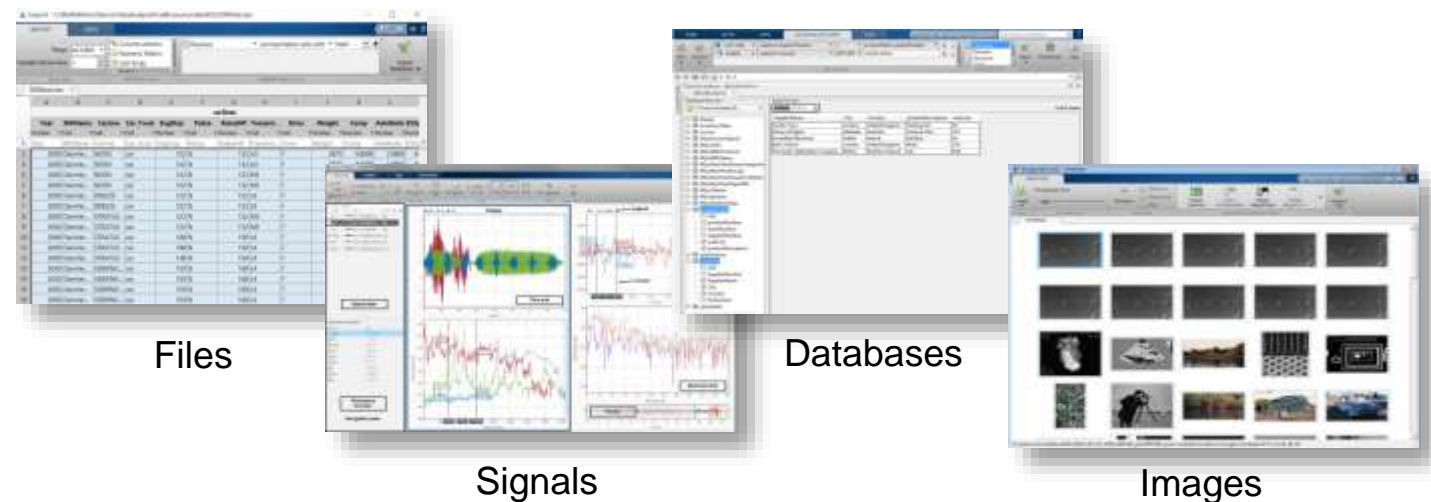
Determine trip duration

tt.hr_of_day = hour(tt.pickuptime);
tt.trip_minutes = minutes(tt.dropofftime - tt.pickuptime)
```

Data Access and pre-processing – challenges and solution

Challenges

- Data aggregation
 - Different sources (files, web, etc.)
 - Different types (images, text, audio, etc.)
- Data clean up
 - Poorly formatted files
 - Irregularly sampled data
 - Redundant data, outliers, missing data etc.
- Data specific processing
 - Signals: Smoothing, resampling, denoising, Wavelet transforms, etc.
 - Images: Image registration, morphological filtering, deblurring, etc.
- Dealing with out of memory data (big data)



- Point and click tools to access variety of data sources
- High-performance environment for **big data**
- Built-in algorithms for data preprocessing including sensor, image, audio, video and other real-time data

Consider Machine/Deep Learning When

Problem is too complex for hand written rules or equations



Speech Recognition



Object Recognition



Engine Health Monitoring

Because algorithms can

learn complex non-linear relationships

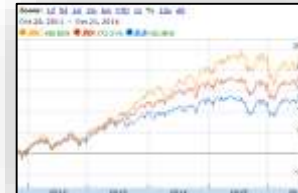
Program needs to adapt with changing data



Weather Forecasting



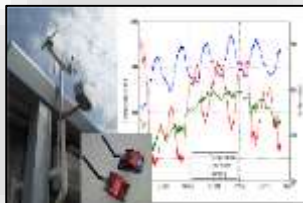
Energy Load Forecasting



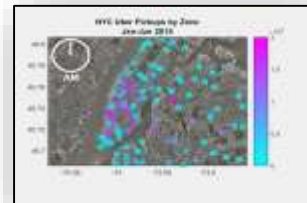
Stock Market Prediction

update as more data becomes available

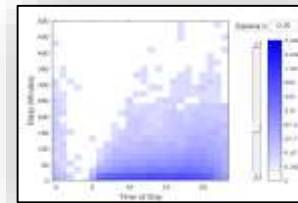
Program needs to scale



IoT Analytics



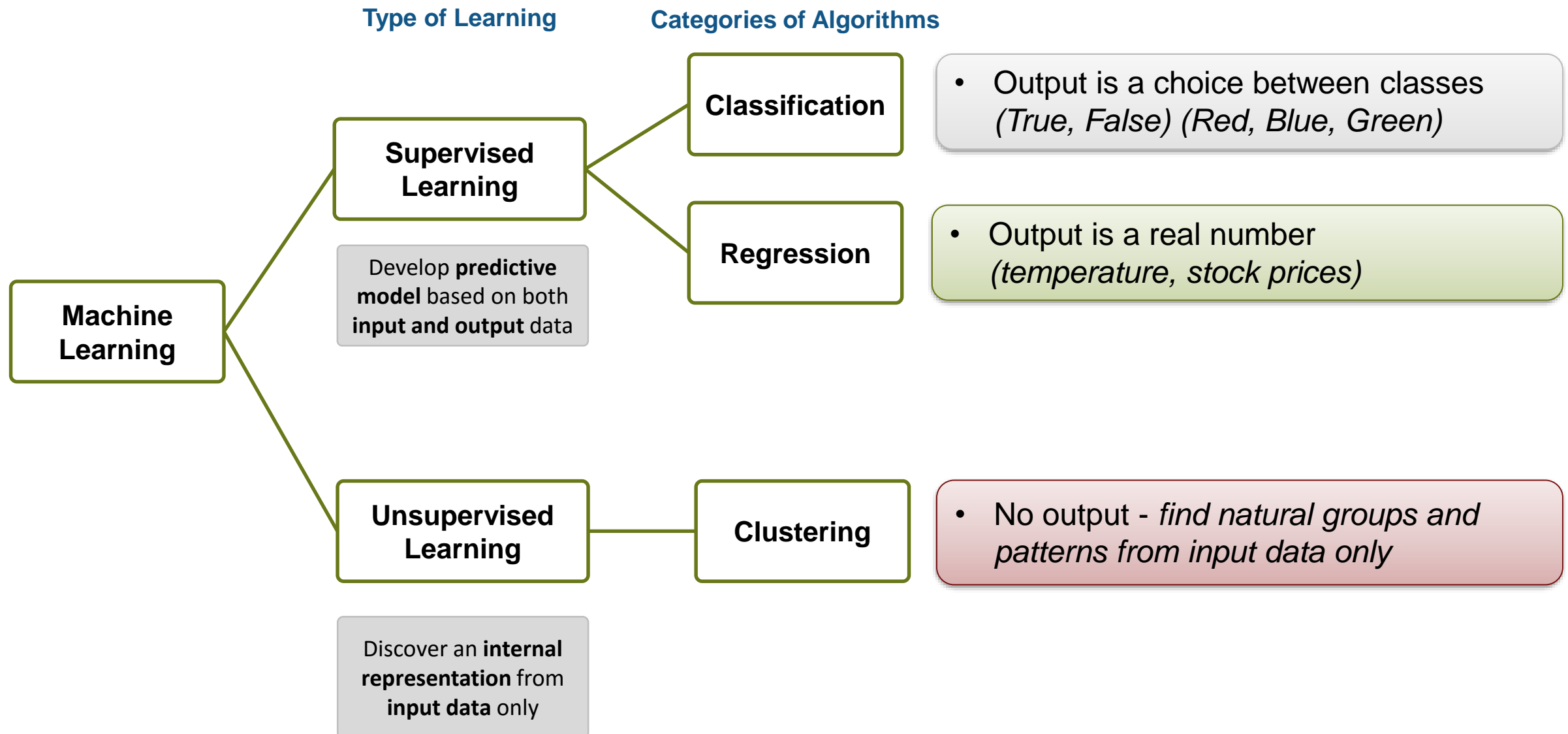
Taxi Availability



Airline Flight Delays

learn efficiently from very large data sets

Different Types of Learning



Machine Learning with Big Data

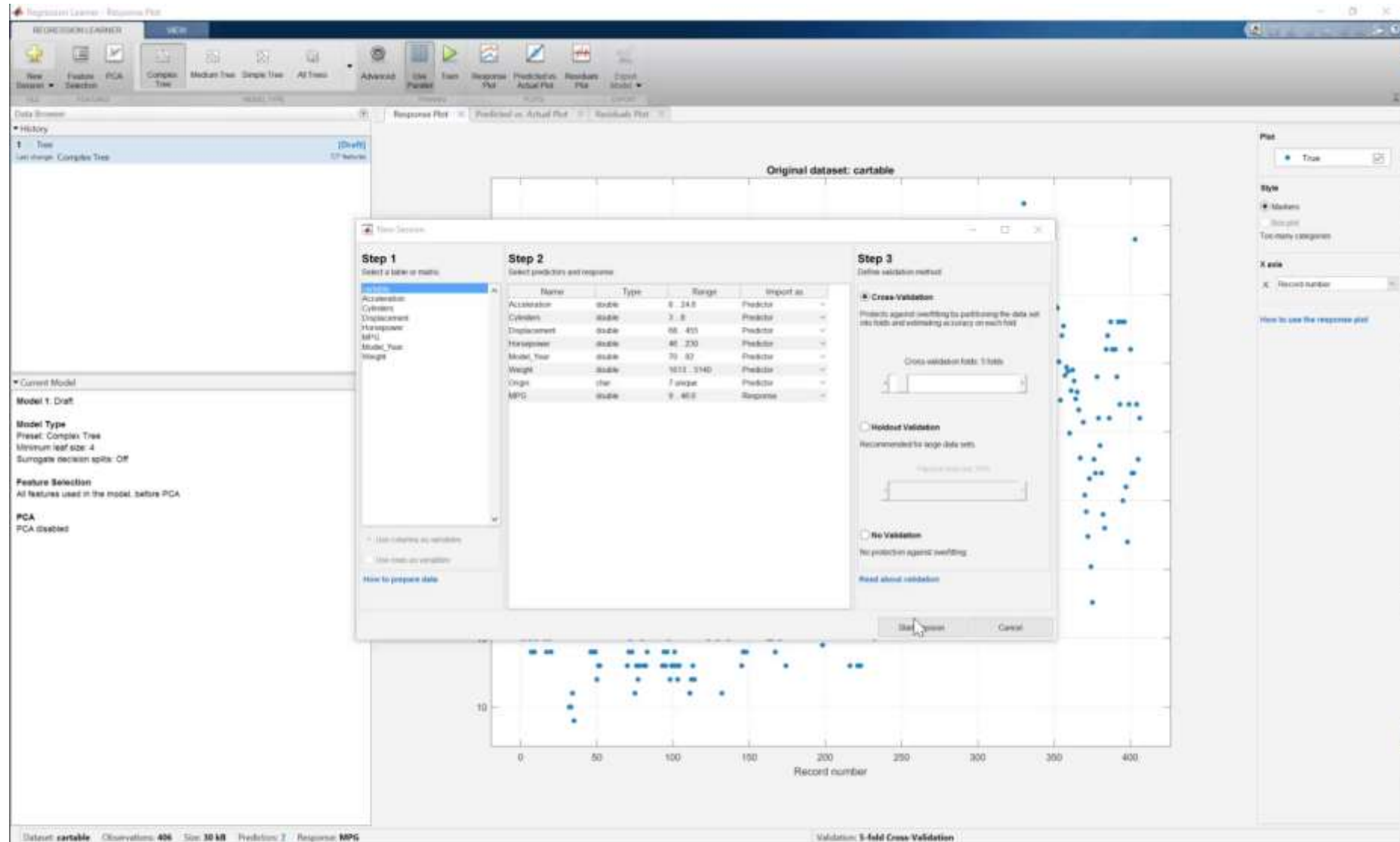
R2016b

- Descriptive statistics (skewness, tabulate, crosstab, cov, grpstats, ...)
- K-means clustering (kmeans)
- Visualization (ksdensity, binScatterPlot; histogram, histogram2)
- Dimensionality reduction (pca, pcacov, factoran)
- Linear and generalized linear regression (fitlm, fitglm)
- Discriminant analysis (fitcdiscr)

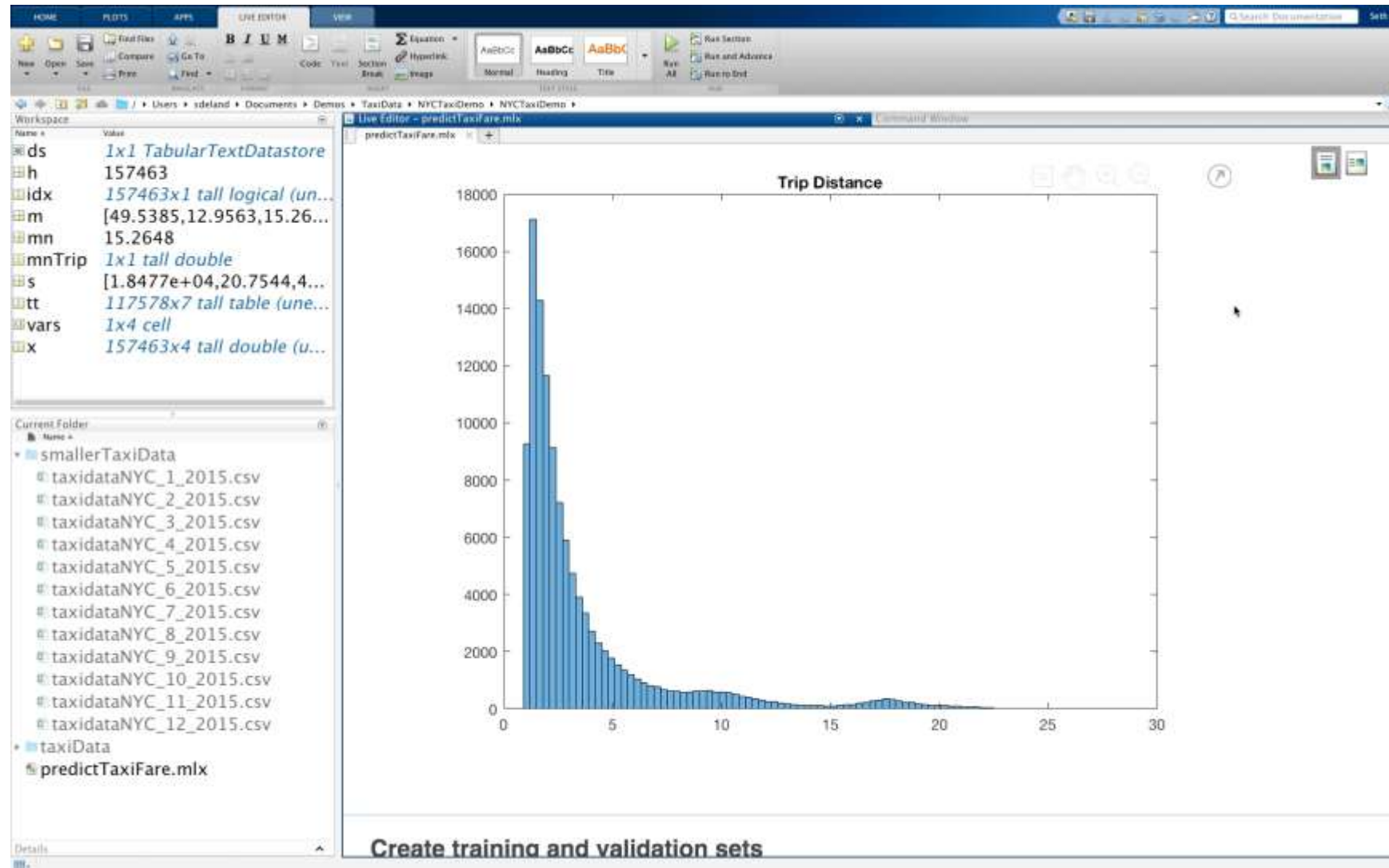
R2017a

- Linear classification methods for SVM and logistic regression (fitclinear)
- Random forest ensembles of classification trees (TreeBagger)
- Naïve Bayes classification (fitcnb)
- Regularized regression (lasso)
- Prediction applied to tall arrays

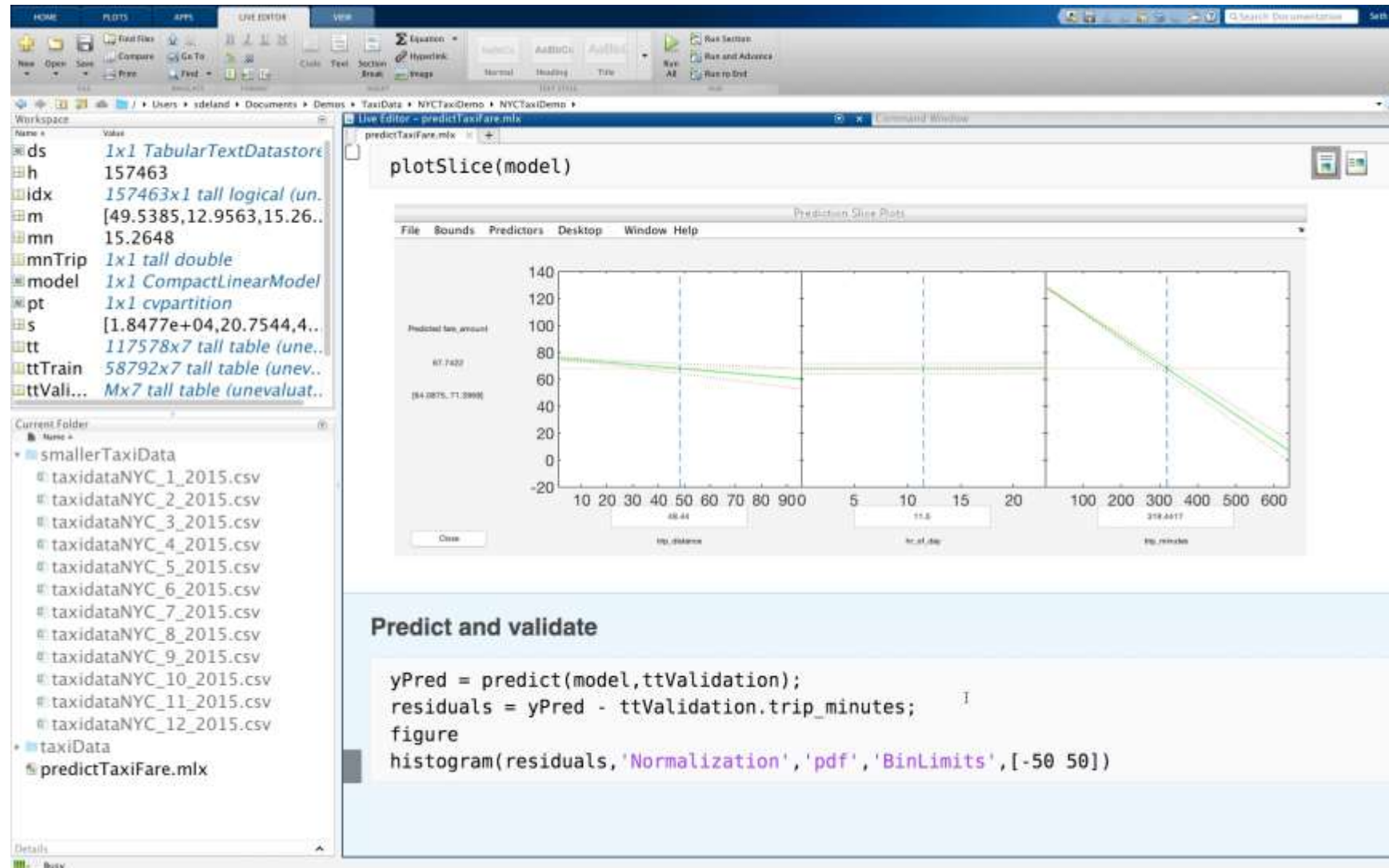
Regression Learner



Demo: Training a Machine Learning Model



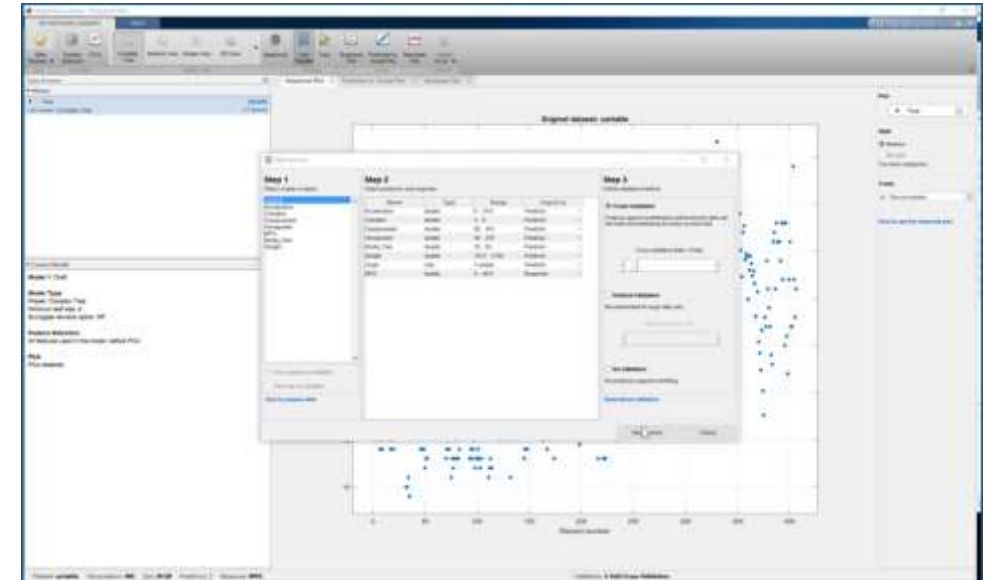
Demo: Training a Machine Learning Model



Regression Learner

App to apply advanced regression methods to your data

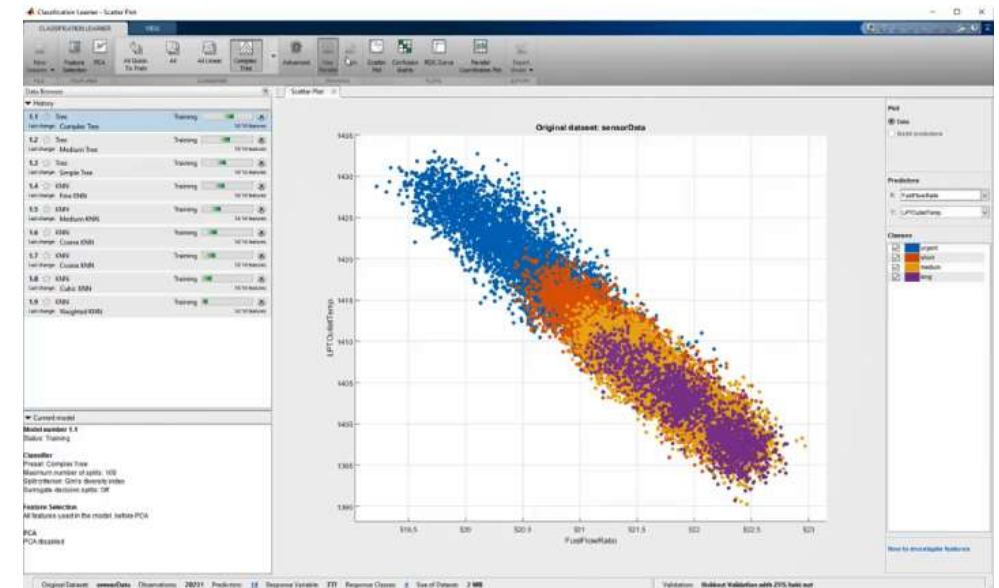
- Added to Statistics and Machine Learning Toolbox in R2017a
- Point and click interface – no coding required
- Quickly evaluate, compare and select regression models
- Export and share MATLAB code or trained models



Classification Learner

App to apply advanced classification methods to your data

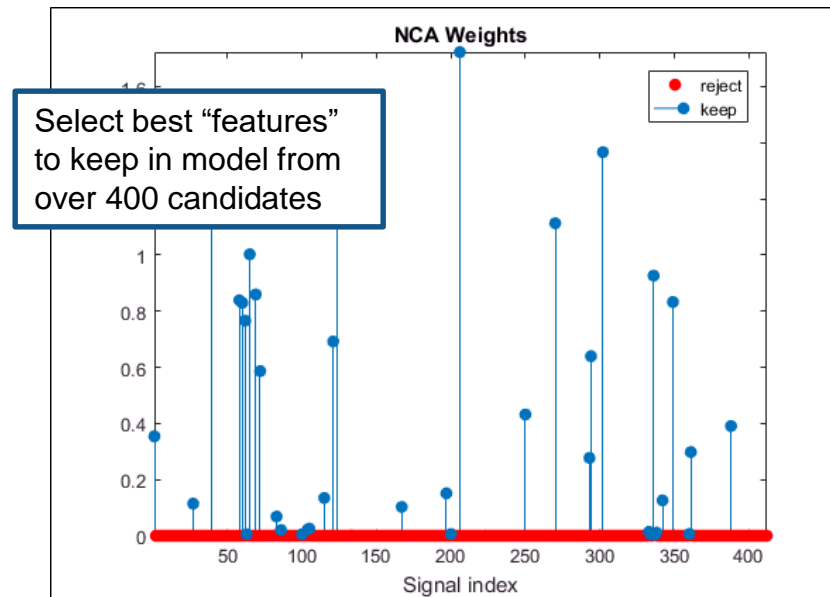
- Added to Statistics and Machine Learning Toolbox in R2014a
- Point and click interface – no coding required
- Quickly evaluate, compare and select classification models
- Export and share MATLAB code or trained models



Tuning Machine Learning Models

Get more accurate models in less time

Automatically select best machine learning “features”



R2016b

NCA: Neighborhood Component Analysis

Automatically fine-tune machine learning parameters

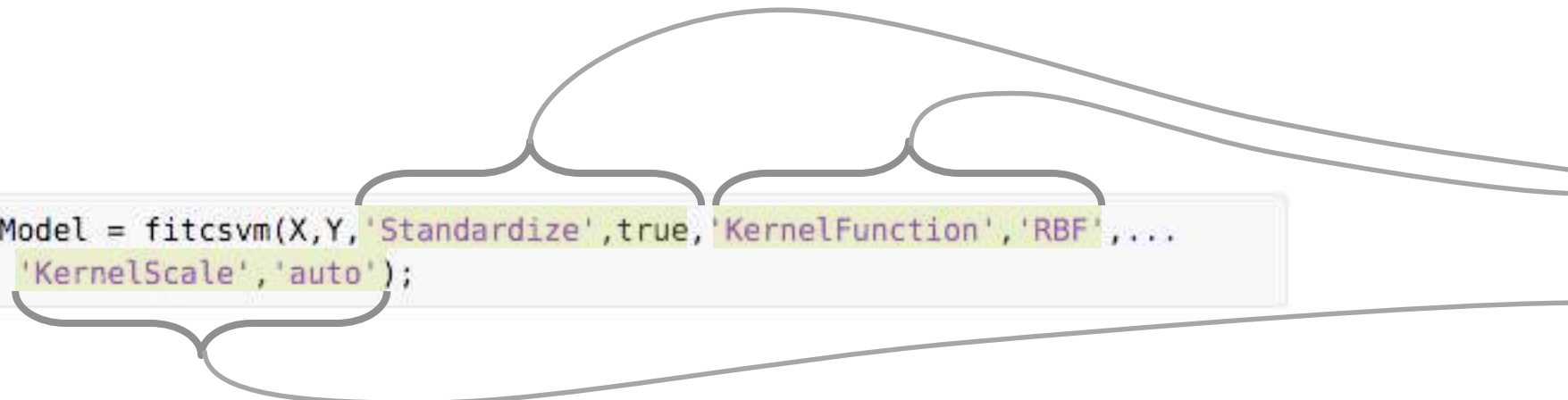


R2016b

Hyperparameter Tuning

Machine Learning Hyperparameters

```
SVMModel = fitcsvm(X,Y,'Standardize',true,'KernelFunction','RBF',...  
    'KernelScale','auto');
```



Hyperparameters

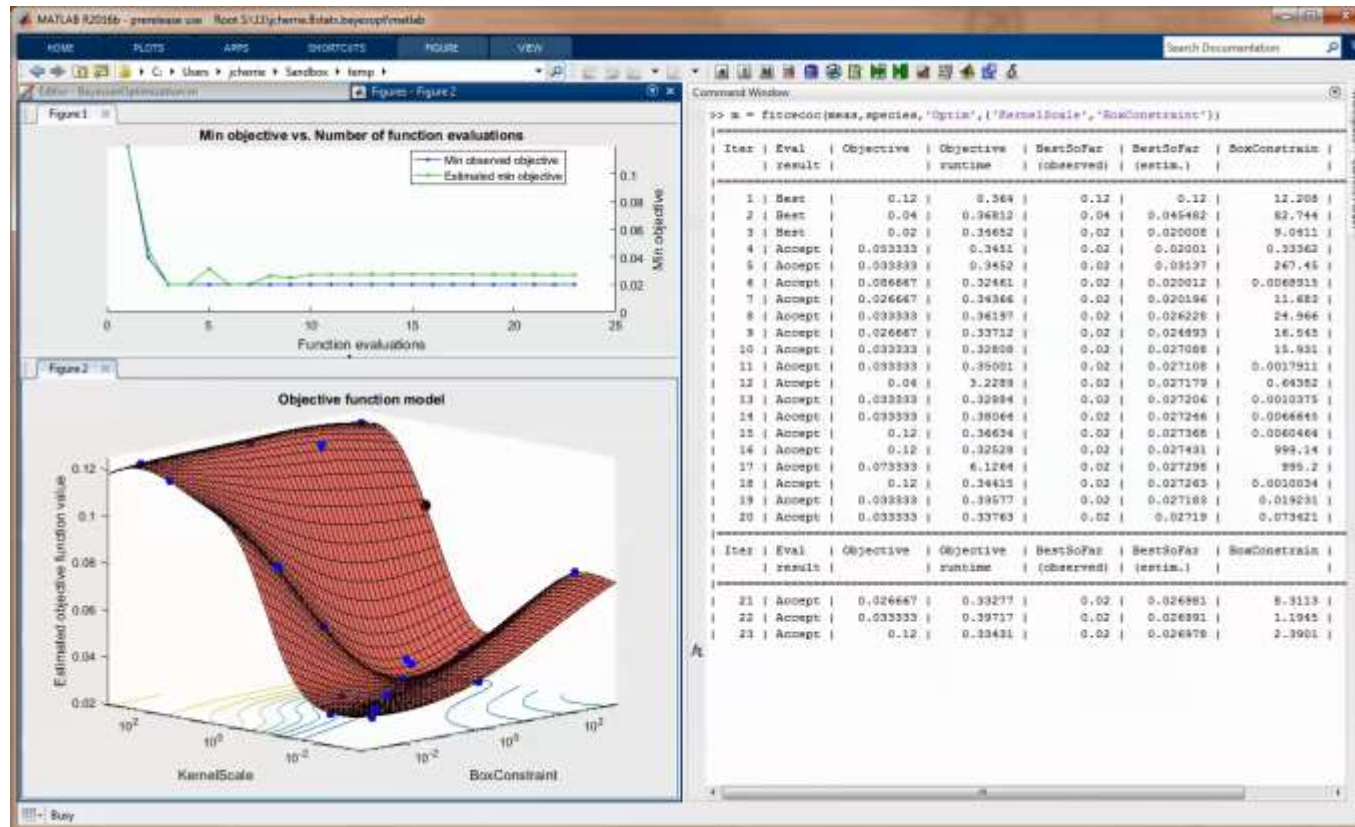
```
SVMModel = fitcsvm(X,Y,'OptimizeHyperparameters','auto');
```

Tune a typical set of hyperparameters for this model

```
SVMModel = fitcsvm(X,Y,'OptimizeHyperparameters','all');
```

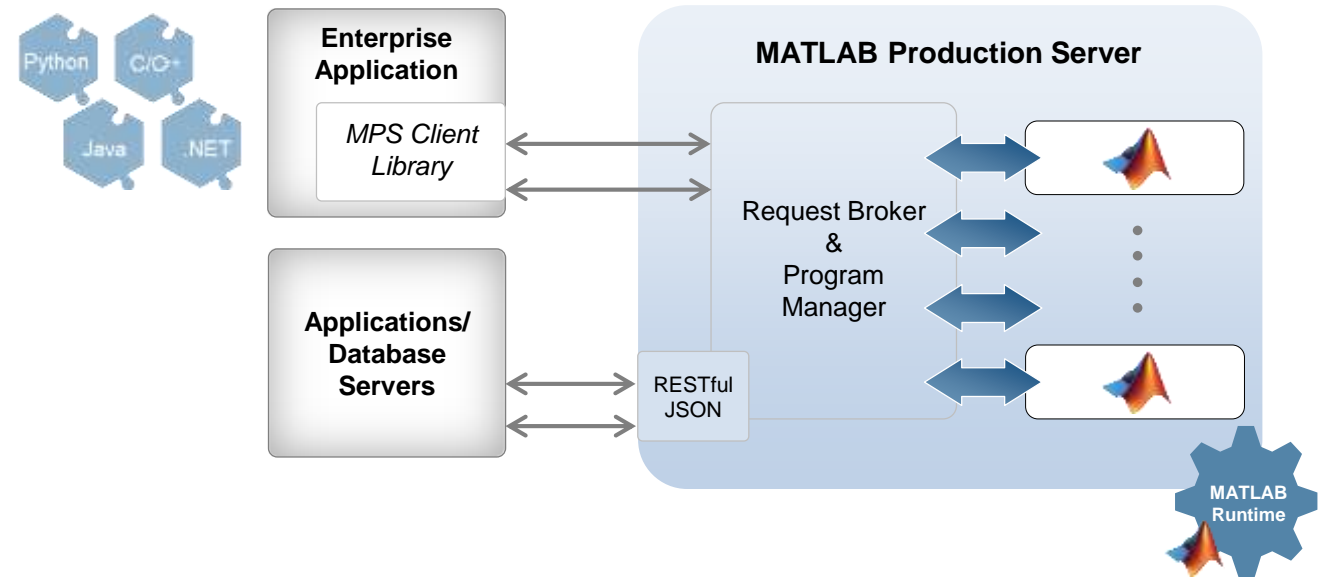
Tune all hyperparameters for this model

Bayesian Optimization in Action

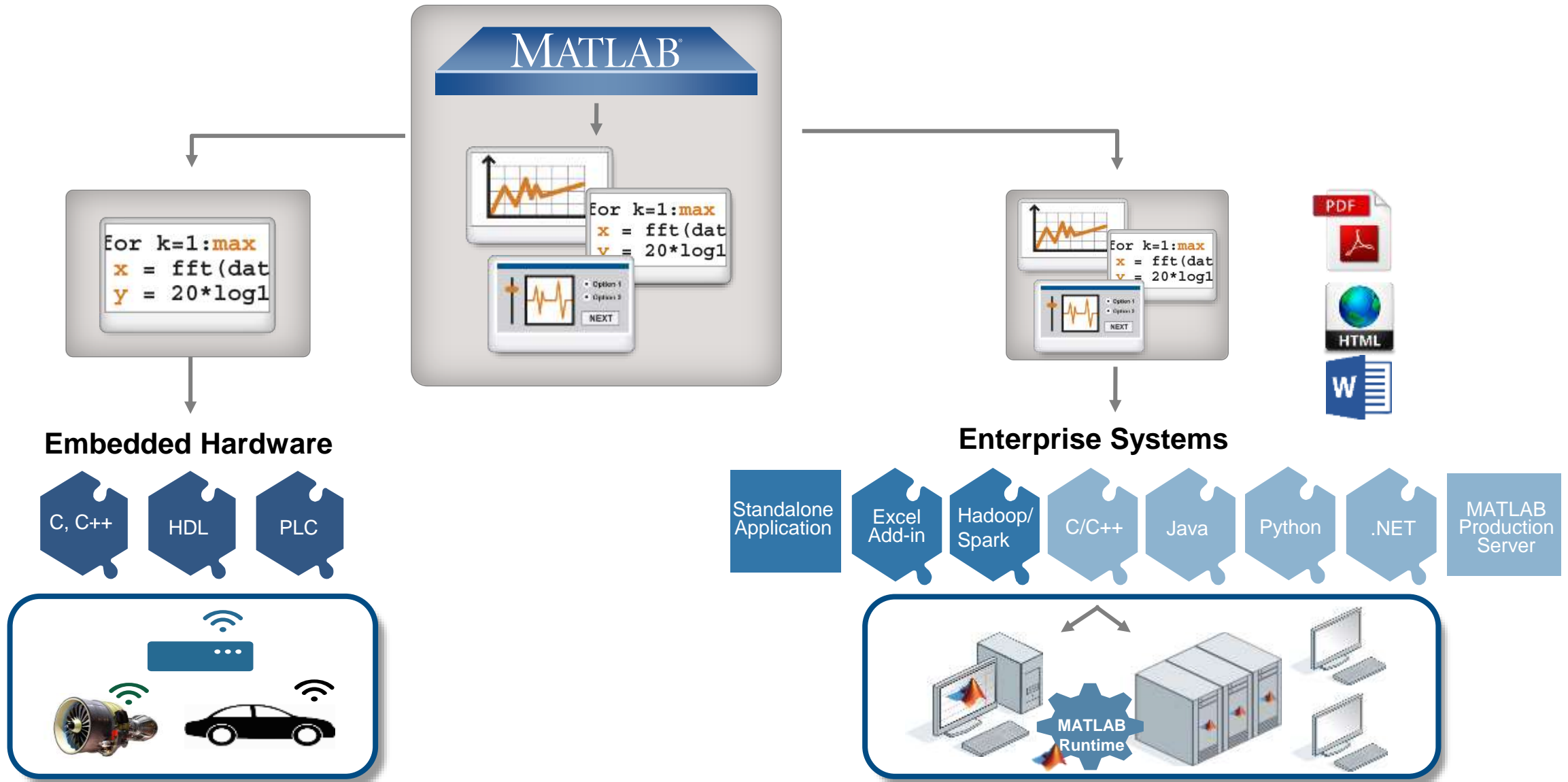


MATLAB Production Server

- Server software
 - Manages packaged MATLAB programs and worker pool
- MATLAB Runtime libraries
 - Single server can use runtimes from different releases
- RESTful JSON interface
- Lightweight client libraries
 - C/C++, .NET, Python, and Java



Integrate analytics with systems



Key Takeaways

MATLAB Analytics work
with **business** and
engineering data

1

- Utilize all of your data.

MATLAB enables
domain experts to do
Data Science

2

- Apply advanced analytics techniques.

MATLAB Analytics
run anywhere

3

- Operationalize analytics to enterprise systems and embedded devices.

Resources to learn and get started

mathworks.com/big-data



mathworks.com/machine-learning



eBook

