

AI를 위한 데이터 활용 및 Machine Learning & Deep Learning Workflow







Artificial Intelligence





MATLAB for Artificial Intelligence



- Machine Learning
- Deep Learning
- Reinforcement Learning
- Predictive Maintenance
- Data Science / Data Analytics
- Signal Processing
- Image Processing
- ...and more



There are two ways to get a computer to do what you want





There are two ways to get a computer to do what you want



📣 MathWorks

Machine Learning is Everywhere

Solution is too complex for hand written rules or equations





Object Recognition



Engine Health Monitoring

learn complex nonlinear relationships

Solution needs to adapt with changing data



Weather Forecasting



Energy Load Forecasting



update as more data becomes available

Solution needs to scale



IoT Analytics



Taxi Availability



Airline Flight Delays

learn efficiently from very large data sets

Agenda 1

- What is Machine Learning?
- Supervised Learning
 - Feature Engineering
 - Model Selection and Training
 - Optimization and AutoML
- Unsupervised Learning
- Deployment
- Resources





Types of Machine Learning





Training a predictive model

Train: Iterate until you find the best model



Predict: Integrate trained models into applications





Machine Learning Workflow





Data Preparation





Demo Example 1: Human Activity Recognition

Classification



Data:

- 3-axial Accelerometer data
- 3-axial Gyroscope data

Dataset courtesy of:

Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra and Jorge L. Reyes-Ortiz. *Human Activity Recognition on Smartphones using a Multiclass Hardware-Friendly Support Vector Machine.* International Workshop of Ambient Assisted Living (IWAAL 2012). Vitoria-Gasteiz, Spain. Dec 2012 <u>http://archive.ics.uci.edu/ml/datasets/Human+Activity+Recognition+Using+Smartphones</u>



What is Machine Learning?

Machine learning uses data and produces a program to perform a task

Task: Human Activity Detection





Machine Learning Workflow – step 1



📣 MathWorks

Machine Learning Workflow – step 2





Feature Engineering

Using domain knowledge to create features for machine learning algorithms

Feature transformation: Reduce dimensionality

Feature selection: Choose subset of most relevant features

Possible feature engineering ideas:

- Additional statistics PCA, NCA etc.
- Signal Processing Techniques power spectral density, wavelets etc.
- Image Processing Techniques bag of words, pixel intensity etc.
- Get creative!





Principal Components Analysis(PCA)





Demo Example 2: PCA





Comparison of Feature Selection Methods

Functions	Predictors	Machine Learning	Training Speed	Types of Models	Accuracy	Caveats
<u>NCA</u>	Continuous	Classification Regression	Medium	KNN SVM (can use for others)	Strong	Requires manual tuning of lambda
<u>MRMR</u>	Continuous Categorical Mix of both	Classification	Fast	Model Independent	Strong	
<u>ReliefF</u>	Continuous Categorical	Classification Regression	Medium	KNN SVM (can still use for others)	Moderate	Unable to differentiate correlated predictors
Sequentialfs	Continuous Categorical	Classification Regression	Very Slow	Model Independent (define custom loss function)	Strong	Doesn't rank all features
<u>F Test</u>	Continuous Categorical Mix of both	Regression	Very Fast	Model Independent	Weak	Unable to differentiate correlated predictors
Chi Squared	Continuous Categorical Mix of both	Classification	Very Fast	Model Independent	Weak	Unable to differentiate correlated predictors



Demo Example 3: Classification Learner App

Classification Learner - Scatter Plot					– 🗆 X
CLASSIFICATION LEARNER VIEW					1 / 4 4 5 C 🗗 🤉 오
🕂 🗉 🗹 🕅 🕅			🕅 🖬 🎵		₽ 🗸
New Feature PCA Linear Quadratic Session - Selection Discriminant	All Logistic Discrimina Regression	Advanced Use Train Parallel	Scatter Confusion ROC Curve Plot Matrix C	Parallel Export Plot Coordinates Plot to Figure	Generate Export Function Model ▼
FILE FEATURES	MODEL TYPE	TRAINING	PLOTS	E	XPORT
Data Browser 🐨	Scatter Plot X Conf	usion Matrix 🛛 🗶 ROC Curv	e 🛛 🛛 Parallel Coordinates Plot	×	
▼ History					Plot
1.1 🏠 Tree Accuracy: 94.7% ^		Predict	ions: model 1.4		
Last change: Fine Tree 4/4 features	4.5				Model predictions
1.2 🟠 Tree Accuracy: 94.7%		•			
Last change: Medium Tree 4/4 features					Correct
1.3 🟠 Tree Accuracy: 95.3%		•			X Incorrect
Last change: Coarse Tree 4/4 features	4 -	•			
1.4 🏠 Linear Discriminant Accuracy: 98.0%		•			Predictors
Last change: Linear Discriminant 4/4 features		• •		• •	X: SepalLength ~
1.5 🕎 Quadratic Discrimi Accuracy: 97.3%		• ••			
Last change: Quadratic Discrimina 4/4 features	•	••	•		Y: SepalWidth
16 Accuracy: 95 3%	3.5	•••			
Last change: Gaussian Naive Bayes 4/4 features	글				Classes Move to Front
17 Naive Paves Accuracy: 05 2%	ž	. I' .			Show Order
Last change: Kernel Naive Bayes 4/4 features	ba	· · · · · ·			setosa
×	<i></i> м ₃ – • •				versicolor
▼ Current Model	Ŭ •	• •	•••••		virginica
Model 1 4: Trained			••ו• •	• •	
Model 1.4. Halled		• • •	* ••		3
Results		• ••	•	•	
Accuracy 98.0%	2.5	• • •••	• •		
Prediction speed ~1900 obs/sec		• •			
Training time 2.161 sec	•	• •	•		
			••		
Model Type	2				
Preset: Linear Discriminant	2				
Covariance structure: Full	4.5	5 5.5	6 6.5 7	7.5 8	How to investigate features
Feature Selection		Se	palLength		
All features used in the model, before PCA					
Data set: fishertable Observations: 150 Size: 2	26 kB Predictors: 4 Re	sponse: Species Respons	e Classes: 3	Validation: 5-fold Cross	-Validation .



Data Augmentation to Improve Accuracy

Label

Count



With augmented image set

	Label	Count
1	jetski	233
2	medium-other	233
3	navigation	233
4	passenger	233





Hyperparameter Tuning



Best: Bayesian Optimization

- Bayesian model indicates impact of change
- Model picks "good" point to try next
- Much more efficient!
- Scale to multi-cores (using PCT) for larger datasets

CLASSI	FICATION LEARNER	VIEW				f	the (Classification/
New Session • FILE	Feature PCA Selection FEATURES	<u>ян</u> <u>ян</u> Misclassification Costs OPTIONS	GET STARTED	8			Regression) Learner
Data Brows	er		All Quick-To	All	All Linear	r	model
1 Tr	86	A	DECISION TREE	ES			
Last chang	e: Disabled PCA			B	ふ	2	
2 🚖 SV Last chang	'M _{le:} Linear SVM	A	Fine Tree	Medium Tree	Coarse Tree	All Trees	Optimizable Tree
3 😭 En Last chang	isemble ie: Bagged Trees	A	DISCRIMINANT	ANALYSIS			
4 C KN	IN Je: Fine KNN	A	×			×.	
	INF		Linear	Quadratic	All Discrimina	Optimizabl Discriminat	e nt



AutoML Machine Learning Workflow





AutoML Workflow in MATLAB

- 1. Generate features by applying Wavelet scattering Note: other (manual) feature generation methods exist!
- 2. Apply Feature Selection techniques
- 3. Select and Optimize Model
 - 3a. Train and optimize various models in Learner App R2019b
 - 3b. Or, automatic model selection **fitcauto**
- 4. Generate C-code or Compile to deploy: codegen

R2()2()a



Model Selection and Optimization

R2019**b**

- 1. Open up Classification (or Regression) Learner App
- 2. Train multiple models
- 3. Perform hyperparameter tuning on top models
- 4. Other advanced optimization maneuvers are manual...

R2020a

- 1. Run fitcauto on your features
- 2. If good enough, DONE. ELSE continue with iterative process above.

Additional Resources:

<u>Tech Talk: Hyperparameter Optimization</u> [4:43 min video] <u>Bayesian Optimization Workflow</u> [doc category page] <u>Hyperparameter Optimization in Classification Learner</u> [doc]



Automated Feature Generation with Wavelet Scattering

Wavelet Scattering Framework [Bruna and Mallat 2013]

- Automatic Feature Extraction
- Great starting point if you don't have a lot of data
- Reduces data dimensionality and provides compact features



Additional Resources:

- <u>Wavelet scattering for ECG</u> [doc example]
- <u>Applying Deep Learning to Signals [3 min video]</u>
- <u>Blog about Wavelet scattering on towardsdatascience.com</u>



Types of Machine Learning





Clustering Motivation 1

- 2명의 대통령 후보가 있는 상황에서 여러분의 당선 전략은?
 - 상대와 나의 여론조사 지지율은 50.2% Vs 47.7% (2.5% p 차이)
 - 현재 결과의 역전을 위해서는 상대편 지지자의 1.3% p 가져와야 함.
- 그렇다면 상대편 지지자 중에서 나를 지지할 수 있는 1.3% p는 누구인가??
 - 유권자들을 세분화하여 군집화
 - 상대편으로부터 나에게 올 수 있는 유권자(군집)에게 집중
 - 즉, 군집을 통해 유권자별 맞춤형 선거전략이 가능
- 유권자들을 적절하게 군집하여 맞춤형 선거전략을 통해 선거 승리 가능함.



Clustering Motivation 2

- 왼쪽 데이터
 - 기존 데이터에 아무 것도 없는 상태(No Labeling)
- 오른쪽 데이터

- 데이터를 군집한 결과(Labeling 가능해짐)





Unsupervised Learning Vs Supervised Learning





Demo Example 1: K-means Clustering Algorithm using MATLAB





Clustering

- 데이터를 다음의 두가지 조건을 만족하는 군집함.
 - 같은 Cluster 내부의 데이터들 간의 유사성: 높음
 - 같은 Cluster 내부의 데이터와 다른 cluster 내부의 데이터 간의 유사성 : 낮음

유사성(similarity)

- 두 사물(데이터)의 다른 점을 구분하는 기준이 되는 지표
- 두 사물(데이터) 사이의 거리(distance, dissimilarity)로 측정
 - 거리를 구하는 방법은 여러가지가 있으나,
 유클리디안 거리 측정을 많이 활용함.
 - 기타 다른 거리 구하는 방식을 활용하여 군집할 수 있음.

 $x = (x_1, x_2, \dots), y = (y_1, y_2, \dots)$

(1) Euclidian distance (dissimilarity)

$$D(x, y) = d(x, y) = \sqrt{\sum_{i} (x_{i} - y_{i})^{2}}$$
(2) Manhattan distance (dissimilarity)

$$D(x, y) = d(x, y) = \sum_{i} |x_{i} - y_{i}|$$
(3) "sup" distance (dissimilarity)

$$D(x, y) = d(x, y) = \max_{i} |x_{i} - y_{i}|$$
(4) Correlation coefficient (similarity)

$$D(x, y) = s(x, y) = \frac{\sum_{i} (x_{i} - \mu_{x})(y_{i} - \mu_{y})}{\sigma_{x}\sigma_{y}}$$
(5) Cosine similarity (similarity)

$$D(x, y) = \cos(x, y) = \frac{x \cdot y}{\|x\| \|y\|} = \frac{\sum_{i} x_{i} y_{i}}{\sqrt{\sum_{i} x_{i}^{2}} \sqrt{\sum_{i} y_{i}}}$$



K-means Clustering Algorithm

개념

- 입력 값으로 군집할 개수(K)가 필요
- 결정된 군집 수에 따른 군집 별 중심을 설정
- 군집 별 중심과 데이터 사이의 거리를 구함
- 거리가 작은 중심 군집으로 해당 데이터를 속하게 함
- 특징
 - 거리기반 군집

 - 반복작업: 반복수행을 통해 군집 결과를 개선함.
 - 짧은 계산시간: 직관적으로 이해되는 간단한 알고리즘으로 계산이 빠름
 - 탐색적 기법: 주어진 자료에 대한 사전정보가 필요 없음(거리를 계산해 나가며 군집)



K-means Clustering Algorithm Process





Application Case of K-means Clustering

- 데이터 마이닝에서 데이터 군집 알고리즘으로 활용
- 트랜드 또는 성향이 불분명한 시장의 데이터 분석에 활용
- 실제 활용 사례1
 - 네이버나 카카오 뉴스 검색 클러스터링: k-means clustering 방식 사용 추정
- 실제 활용 사례2
 - 스타트업 기업: 언니의 파우치(https://www.unpa.me/)
 - 국내 화장품에 대한 소비자들의 리뷰 정보를 보유하고 있는 업체
 - 리뷰에 엄격한 기준이 적용되어 퀄리티가 높은 리뷰가 많음.
 - 고객들의 정보가 연령, 피부타입, 고객의 커뮤니티 활동, 구매 정보 등으로 나뉘어 축적되어 있음.



Application Case of K-means Clustering

- 이용자의 11개 변수를 활용해 K-means Clustering 을 통해 5개 cluster로 군집

- ① 화장하기 시작한 중고등학생
- ② 언니의 파우치 주축 활동 멤버로 고등학생과 20대 초반
- ③ VVIP, 언니의 파우치 활동 대장 20대
- ④ 언니들을 보며 배우는 고등학생과 20대 초반
- ⑤ 이벤트만 관심, 조용한 **30대 이상 진짜 언니들**




Application Case of K-means Clustering

- 5개 그룹별로 '언니의 파우치' 스토어에서 구매에 영향을 주는 요인 분석
 - Tier 1인 중고등학생이 다른 그룹보다 앱 내 활동이 구매에 미치는 영향이 컸음.
 - 이들의 구매 유도를 위해 팔로워 및 팔로잉 기능 강화
 - 인적 네트워크 활성화 등 앱 내 활동 활성화 방안이 필요
 - Tier 5인 30대는 구매력은 높으나, 활동이 없는 조용한 고객군으로 분석
 - 분석 결과 이벤트에 민감한 것으로 파악
 - 30대를 위한 타깃 이벤트 기획 필요 안티에이징 화장품 리뷰 이벤트를 진행하고 좋은 반응을 얻음.
- 언니의 파우치 주요 고객층이 10대 후반에서 20대 초반임을 파악
 - 그룹별 고객 분석 전: 주고객이 20대 후반이라는 막연한 추정
 - 이들을 타깃으로 하는 신제품 출시



Application Case of K-means Clustering



['언니의 파우치' 스토어 8~11월 매출액]



Machine Learning Workflow





Integrate Analytics with Systems





Machine Learning for Edge Analytics and Code Deployment

Deploy trained models as standalone C/C++ code

- Apply algorithms to out-ofmemory data using tall arrays
- Generate C/C++ code for predictive models
- Generate fixed-point C/C++ code for SVM models, decision trees, and ensembles of decision trees
- Update deployed models without regenerating code





Using MATLAB with Other Languages

Calling Libraries Written in Another Language From MATLAB



Calling MATLAB from Another Language



- Java
- Python
- C
- C++ _____
- Fortran



- COM components and ActiveX[®] controls
- RESTful, HTTP, and WSDL web services
- Java
- Python
- C/C++
- Fortran
- COM Automation server



Challenges in Machine Learning

Steps	Challenge
Access and Explore data	Data diversity Numeric, Images, Signals, Text – not always tabular
Preprocess Data	Lack of domain tools Filtering and feature extraction Feature selection and transformation
Develop Predictive Models	Time consuming Train and compare several models to find the "best" Select optimal parameters and avoid overfitting
Integrate Analytics with Systems	Platform diversity Translate analytics to production Deploy on different target platforms
Iterate	



MATLAB Strengths for Machine Learning

Challenge	Solution
Data diversity	Extensive data support Work with signal, images, financial, textual, and others formats
Lack of domain tools	High-quality libraries Industry-standard algorithms for Finance, Statistics, Signal, Image processing & more
Time consuming	Interactive, app-driven workflows Focus on machine learning, not programing Select best model and easily fine-tune model parameters
Platform diversity	Run analytics anywhere Code generation for embedded targets Deploy to broad range of enterprise system architectures
	Flexible architecture for customized workflows Complete machine learning platform

MathWorks

Summary: Complete Machine Learning Workflow







Agenda 2

• What is Deep Learning?





Machine Learning vs Deep Learning





Machine Learning vs Deep Learning

Deep learning performs end-to-end learning by learning features, representations and tasks directly from images, text and sound





Machine Learning vs Deep Learning



	Machine Learning	Deep Learning
Training dataset	Small	Large
Choose your own features	Yes	No
# of classifiers available	Many	Few
Training time	Short	Long



Deep learning is part of our everyday lives



Speech Recognition







Automated Driving



Deep learning applications: mainstream vs. engineering

Mainstream



Detecting Objects

Engineering and Science



Identifying Machinery at Shell

Deep Learning Detection



MATLAB Deep Learning used in Industry







ECU Vehicle Control
Denso



Shell



MATLAB Deep Learning used in Research



Predicting gastrointestinal cancer (July 2019)



<u>Converting brain waves to speech to help ALS patients</u> <u>communicate</u> (Nov 2019)



MATLAB is used in many areas of medical imaging



Digital pathology



Radiotherapy planning



Ophthalmology/OCT



Endoscopy



Radiology (MRI, US, X-ray, CT)



Intravascular imaging



Applications of deep learning for images and video



YOLO v2 (You Only Look Once)



Semantic Segmentation using SegNet



Applications of deep learning for signal processing

a · I · 4 I 🗠			
15-			Channel 1 Channel 2 Channel 3
10			
\$ 2) 5			
cceleration (m			
-5	Figure 2: Acquisition Control	- 0 ×	
-10	Pause		
-15	Actual: Laying Estimated: Laying		
o	0.5 1 Time (secs)	15	2 25

Signal Classification using LSTMs





Al-driven system design

Data Preparation



Data cleansing and preparation



 $- \sum$



Simulationgenerated data

AI Modeling



Hardware accelerated training



System simulation



and validation

Simulation & Test

Integration with

complex systems

Deployment



Embedded devices



Enterprise systems



Edge, cloud, desktop



Data preparation represents most of your AI effort...

Transforming raw data for useful modeling and analysis is a critical step.





Spend less time preprocessing and labeling data

Synchronize disparate time series, filter noisy signals, automate labeling of video, and more.

Data Preparation

Image: Image:



᠆ᡗᠵᠴ

Human insight

Simulationgenerated data





Data Preparation





Start with a complete set of algorithms and pre-built models

AI Modeling



Model design and tuning



Hardware accelerated training



Algorithms

Machine learning Trees, Naïve Bayes, SVM...

Deep learning CNNs, GANs, LSTM, MIMO...

Reinforcement learning DQN, A2C, DDPG...

Regression Linear, nonlinear, trees...

Unsupervised learning K-means, PCA, GMM...

Predictive maintenance RUL models, condition indicators...

Bayesian optimization

Pre-built models

Image classification models AlexNet, GoogLeNet, VGG, SqueezeNet, ShuffleNet, ResNet, DenseNet, Inception...

Reference examples

Object detection Vehicles, pedestrians, faces...

Semantic segmentation Roadway detection, land cover classification, tumor detection...

Signal and speech processing Denoising, music genre recognition, keyword spotting, radar waveform classification...

...and more...



Increase productivity using Apps for design and analysis

Use MATLAB Apps to design deep learning networks, explore a wide range of classifiers, train regression models, train an optical character recognition model, and more.





Model design and tuning



Hardware accelerated training

nteroperability



Deep Network Designer app to build, visualize, and edit deep learning networks

					and the second statements						
1111111											60
and a second second		1 811								-	and the second second
	1		1								
Angles +	1.0	of Patter+	100								
() (nonnara) bia	-	OCCUPATION OF	PHILE DEF	187.							
Cesila	1000		mus shirt.	Parati IC							
insura Comission I	+ lives	et flattaire .									
Disease Initial Locations France	1	en Tarra		3178080	(1010.00 PM						(708)
And a second second	1000	Committee and				Corgana 7	A.	Skapael +	0	Drat-	
Tanna Long					6	- Marring 1		Elaborati . #	× 1	Second	
Lagerman Lagreng Rose Range											
Dissess Louring Hale Carly Sile and	Tables	Landon V	Television		Concerning Street, I	Committee and	Concerning Sectors	Include the second	Parameters and the	1 Palator	
Add Core Ball Polic Barro	1.00	annet.	Program		Dapeod Time	when a base	ID-STREET	Thereig Acco.	a management	129.00	1004.84
I may Print that of Penil Dama D Lagar	10	D Davies	_	-	101104-0010	1.000000	0.0	11.1110	4.0	1001	111
There become Spill Produ-	1	Contrast		- 20	Official and the state	1100000-0		311,0943	412		
		O Carrier	_		a la base di se	200					
	<u> </u>			- 200			1.000	11.71.00			
		Concerned in the second		- 342	distance where	i siste a					
	*	6 Comm		-	Gib Street Water		10.000		1.00		
		O Names	1.000	21/2		0.000	8,000	411111			
	8	Li thanks	of Longing	1.175		1.0000-0	5.000				
	10	to thread		111		1.0005-5	8,000				
	10	Lo Question		125		9,000	2.000				
	10	U. Goront		111		2.340	10,000				
	11	to Generi	10 10	115		1100010-6	0.000	1			
	16	II Descel				1.000	0.000	-			
	1.0	And the surgery	-	1 1 1 h		2.000	8.000				

Experiment Manager app to manage multiple deep learning experiments, analyze and compare results and code



Deep Network Designer

📣 Dee	p Network Designer					8 <u>—</u> 8	
DESI	GNER.				\$\$006X		?
New In	Image: Second	Fit C Zoom In Fit C Zoom Out to View	Auto Arrange LAYOUT ANALYSIS TRA	Train Export			Ā
LAYER	LIBRARY	-	-		×	PROPERTIES	
	imageInputLayer	E	reluLayer	reluLayer	inc ma	Number of layers Number of connections Input type Output type	144 170 Image Classification
2 2 2	image3dInputLayer sequenceInputLayer	pn_4a tion2dL	inception_4a convolution2dL	inception_4a convolution2dL			
CONVO	roiInputLayer	on_4a-r	inception_4a-r	inception_4a-r			
民	convolution2dLayer		Y				
	convolution3dLayer		incepti depthC	on_4a oncaten			
	groupedConvolution2dL transposedConv2dLayer					· OVERVIEW	
國	transposedConv3dLayer		inception_4b convolution2dL	inception_4b convolution2dL			
*	fullyConnectedLayer		J			魚書	
SEQUE	INCE	[inception_4b-r	inception_4b-r	inc ma		
Ð	IstmLayer -		Teiurayei	reiurayei			
14							▶1



Hardware acceleration and scaling are critical for training

MATLAB accelerates AI training on GPUs, cloud, and datacenter resources without specialized programming.





MATLAB interoperates with other frameworks

Supports ONNX and can exchange models with PyTorch, TensorFlow, and other frameworks.





Modeling

AI Modeling



Model design and tuning



Hardware accelerated training







Models need to exist within a complete system

In automated driving systems, AI for perception must integrate with algorithms for path planning, braking, acceleration, and other controls.

Simulation & Test



Integration with complex systems



System simulation

— × System verification and validation





Deploy to any processor with best-in-class performance

AI models in MATLAB and Simulink can be deployed on embedded devices, edge devices, enterprise systems, the cloud, or the desktop.





Deploy to enterprise IT infrastructure





Deployment



Deployment





Enterprise systems

Edge, cloud, desktop

REPORT								1 🖗 🤨
Back Forward O Find	Trace	A	Package	Export Report				
	Code	MATLAB	Code 👻	Information				
NAVIGATE 1	TRACE	EDIT	S	HARE				
MATLAB SOURCE	E Fu	inction: yold	v2_detect	- volov2 /	datast(in)			6
Function List	2	Tuncere	outing	= y010v2_t	letect(III)			
Call Tree	3	% Cop	yright 2	018-2019 TH	ne MathWorks, I	nc.		
yolov2_detect.m	4			0.01				
fx yolov2_detect GPU	6	persist	ent yolo	v20bj;				
CudnnApi.p	7	if isem	pty(yolo	v2Obj)				
	8	yol	.ov20bj =	coder.load	dDeepLearningNe	twork('Yolov2Using	ResNet50_ONNX.mat');	
	9	end						
	11	% pass	in input					
	12	[bboxes	,~,label	s] = yolov2	20bj.detect(in,	'Threshold',0.5);		
	13	e/			.1		store for MATLAD	
	14	% execu	ition	OLICAL TADE	ers to cerr and	ay of charactor ve	CLOPS TOP MATLAB	
	16	if code	r.target	('MATLAB')				
GENERATED CODE	17	lab	els = ce	llstr(label	ls);			
Files GPU Kernels	18	end						
Source Files	20	% Annot	ate dete	ctions in t	the image.			
DeepLearningNetwo	21	outImg	= insert	ObjectAnnot	tation(in,'rect	angle',bboxes,labe	els);	
DeepLearningNetwo	22							
++ MWAdditionLayer.cl	SUM	IMARY		ALL MES	SAGES (0)	BUILD LOGS	CODE INSIGHTS (1)	VARIABLES
MVVAdditionLayer.n		Cod	0 000	oration	aucococ			
MWAdditionLayerIn		Cou	e gen	eration	successi	ui		
+ MWBatchNormaliza	G	enerated	17-Sep-	2019 14:21:46				
MWBatchNormaliza	D.	on:						
MWBatchNormaliza	BI	utout file:	C:\Usor	hcuon s\shmitra\\//ork		minar\19h\DecNetImpo	rtValay2\HalparEilesAndEupstions\w	olov2 detect max maxw64
MWBatchNormaliza	Pr	ocessor:	Generic	->MATLAB Ho	st Computer	sinnar (130/ixesivetimpol	renore a repen near-indranduorisiy	
MWCNNI avorimni	1	0000011	Conollo		er e sinpatoi			×.
14								







Demo: Malaria Detection

	Label	Count
1	Parasitized	13779
2	Uninfected	13779

Uninfected

Parasitized

Parasitized

Parasitized



Parasitized

Parasitized

Uninfected



Uninfected

Uninfected







Parasitized



























Uninfected





Parasitized

Parasitized


Demo: Malaria Detection



Parasitized





Parasitized



Uninfected

Uninfected





Parasitized

Parasitized

Parasitized

Parasitized



Parasitized



Uninfected





Uninfected



Parasitized

Uninfected







Parasitized







Demo: Malaria Detection





Demo: Malaria Detection





Target Class



Interactively design and edit neural networks

- Use deep learning for medical imaging tasks such as segmentation, classification and detection
- Interactively create and edit deep learning networks
 - Built-in Deep Network Designer app
- Analyse network architecture to detect errors and layer compatibility issues before training



Interactively build and visualise network structures



Segmentation of brain tumours in 3D images using deep learning



Import pre-trained models for fast implementation

- Access pretrained networks and use them as a starting point for new models
 - Multiple pre-trained networks available online
- Perform transfer learning to use the learned features in the network for a specific task
- Compare the accuracy of pre-trained
 networks for a specific medical imaging task



A list of pretrained networks





Example: A list of pretrained networks **Deep Network Designer**





Track training progress and compare performance

- Train networks under various initial conditions and compare the results
 - Built-in Experiment Manager app
- Easily compare different network
 architectures using the same training data
- Use custom metrics to compare the performances of deep learning modes



Design and run experiments to train and compare deep learning networks



confusion matrix



Example: Manage multiple deep learning experiments **Experiment Manager**

			1	Experiment Mana	ger		
EXPERIMENT MANAGER						1.1	00000
New Duritate	> 🔲 n Stro 1	Plat Confusion Plat	[Filter Export			
FILE ENVIRONMENT	RUN	REVIEW RE	844,TA F	LTER BHARE			a mutata
PHOLECT BHOWSER PHOLECT BHOWSER Data Data Data Experiments Experiments Results Scripts data.m layers.m blayers.m options.m options.m options.m options.m	Result De Experi Experi Start 1 Total 1 Status Leger Numb filters	tails iment Name iment Description Time Number of Trials s of all Trials ad per of trials after ap	5_2019_13_52_51_78	iment] 19, 1:50:49 PM Jeued(D)	ning(0) 📕 Co	mplete(16) E Stopped	Status: Fluming Queued Stopped Complet Accuracy
Run - 03 05 2019 13 52 51 782	Trial	Status	NumTraining	LearnRate	Accuracy	Loss	0.0070 0.00
Hmu + no_no_xn1a_10_bx_o1_tex	Irial	1 Complete	Num Iraining 650	Learnsate 0.01	0.0090	0,2910	
		2 Complete	700	0.01	0.0070	0.2910	Loss
		3 Complete	750	0.01	0.0080	0.2100	6 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
		4 Complete	800	0.01	0.0080	0.2550	4-
			0.3				
		5 Complete	650	0.02	0.0090	0.2960	3-
		5 Complete 6 Complete	650	0.02	0.0090	0.2960	3-
		5 Complete 6 Complete 7 Complete	706	0.02	0.0090	0.2960 0.2970 0.2490	2-
		5 Complete 6 Complete 7 Complete 8 Complete	650 700 750 800	0.02	0.0090 0.0070 0.0090 0.0090	0.2960 0.2970 0.2490 0.2140	3- 2- 1-
		5 Complete 6 Complete 7 Complete 8 Complete 9 Complete	65(700 750 800 650	0.02 0.02 0.02 0.02 0.03	0.0090 0.0070 0.0090 0.0090 0.0080	0.2960 0.2970 0.2490 0.2140 0.2920	
		5 Complete 6 Complete 7 Complete 8 Complete 9 Complete 10 Complete	650 700 750 800 650 700	0.02 0.02 0.02 0.02 0.02 0.03 0.03	0.0090 0.0070 0.0096 0.0090 0.0080 0.0080	0.2960 0.2970 0.2490 0.2140 0.2920 0.2920 0.2960	3 - 2 - 1 - 0 - 1 - 1 - 1 - 1 - 1 - 1 - 1 - 2 - 1 - 1 - 2 - 1 - 1 - 2 - 2 - 1 - 2
		5 Complete 6 Complete 7 Complete 8 Complete 9 Complete 10 Complete 11 Complete	654 700 750 800 650 700 750	0.02 0.02 0.02 0.02 0.02 0.03 0.03 0.03	0.0090 0.0070 0.0090 0.0090 0.0080 0.0090 0.0090	0.2960 0.2970 0.2490 0.2140 0.2920 0.2920 0.2960 0.2960	



Interoperate with deep learning frameworks

- Import models from TensorFlow-Keras and Caffe for inference and transfer learning
- Exchange models in ONNX format for working with other deep learning frameworks
- Export trained MATLAB deep learning networks to ONNX model format for sharing models



Import models from other deep learning frameworks



Exchange models using ONNX format



Accelerate model training in the cloud

- Train models on different environments using
 MATLAB Deep Learning Container
- Reduce deep learning training times with cloud instances
 - AWS and NVIDIA GPU Cloud support
- Run deep learning training across multiple processors on multiple servers



Accelerating training in the cloud



Run training across multiple processors on multiple servers



Share trained deep learning models with end users

MATLAB Compiler for sharing MATLAB
 programs without integration programming

- MATLAB Compiler SDK provides implementation and platform flexibility for software developers
- MATLAB Production Server provides the most efficient development path for secure and scalable web and enterprise applications





Deploy trained networks to embedded systems

- Generate optimised CUDA code for performancecritical applications
- Generate C++ code to deploy deep learning networks to Intel and ARM processors
- Deploy trained networks as C++ shared libraries, Microsoft .NET assemblies, Java classes and Python packages



Generate CUDA code for optimised performance on GPUs



Optimise C++ code for embedded processors



Accelerate algorithm deployment by running in parallel

- Speed up your deep learning models by using GPU and multicore CPU processors
- Legacy MATLAB algorithms can run on GPUs with minimal code changes
- Run deep learning models directly on virtual clouds such as Amazon Web Services (AWS) or Microsoft Azure



Deploy MATLAB algorithms and deep learning models on GPUs and multicore CPUs



Run MATLAB algorithms directly on EC2 instances in the Amazon Web Services (AWS)



Why MATLAB & MathWorks for Deep Learning?

- ✓ MATLAB provides integrated and complete workflow
- Pre-trained models and interactive tools allow fast implementation from concept to code
- Trained models can be shared to end users easily online or using executable apps
- Automatic code generation enables deployment on embedded production hardware
- Support offered by engineering support, comprehensive documentation, demos and application examples





Online examples of deep learning in medical imaging

- Deep Learning Toolbox
- <u>3D Image Segmentation of Brain Tumors Using Deep Learning</u>
- Deep Learning for Medical Imaging: Malaria Detection
- Medical Image Segmentation Using SegNet
- <u>3-D Deep Learning : Lung Tumor Segmentation</u>
- <u>3-D Deep Learning : 3-D Volume Labeling Assist Tool</u>







For more information visit:

mathworks.com/medical

mpliance with FDA/CE gulations and Standards else and amaktion of method device systems is	a harden			Renew Owner.	🛔 The officers 🔪 Constraints
mpliance with FDA/CE gulations and Standards eting and strandards and standards and strandards	Barriell .				
eling and simulation of medical device systems is access both complete and the second statements of the second statements of the second statement by		Contraction of the second seco		Biomedical Data Analysis and Algorithm Development	60
gn, development, and tasking The FDA has Seen stighting free use of missiening and annuation technogues	Verture			Whether you're analyzing barnesbaal dataxets ur developing adkanood algorithms for diagnostic an therapeutic medical devices. MATLAB provides yr feadbilty and power to work with complex data on engineering resignts.	or with the gradient of dense
 lost faie years to better understand how it can assess mpact of engineering decisions on device performance safety without relying purely on animal end thursan of blais. 	a Oteche Ang an Oteche Alla Teeling Alla Teeling Alla Teeling Alla Teeling	Kyloot Cude		As an engineer or researcher working with borresy you care	draithfa,
Using dynamic system modeling and simulation during the medical device development process can help reduce regulatory toutien and speed up the submission tradines. Algorithm Verification and Text	uter and Tool Valcation in MATLAB	Medical Devices	dente con	House of Excel Automate the excellence and analysis of maps and agraits from fordease	a. wites
usemating the creation of many angleweing rs. MathViens tools used can also be variabled for in FDA/CE-regulated workflows and to meet nonized standards such as IEC 62304.		With MATLAB and Birnathk, you can - Develop and test odvanced algorithms and entire systems before implementation - Simulate and test embedded software alongside mechatoric systems early in the design phase	Lowell ant Madel-Based Design th enables us to implement and hast design reliveneets is minutes and reduce overal development time? and a second development time?	 Develop, test, and verify algorithms including a intelligence (AI) and machine learning models Deploy WATLAB code on processors, GPUs, a for production or prototyping 	rifical n/ FPGAs
m Norm Explore Products		 Prototype designs and create procRoRsoncepts by automatically generating met-fine code Use static analysis to find software bugs and prove correctness of your models and code Automate reporting to prove and accelerate compliance with PDA/CE regulations and industry 	- 844 O-minger Ditor Dale Dege	Learn More	Explore Products
Noal the MethWorks Tool Validation KII - Simuloik Requirements ¹⁶ -MethWorks Research and Colleboration - Simuloik Test ¹⁶ ement Summary - IEC Centeration KI		standards such es IEC 62304		Dutch Epilepsy Clinics Foundation Automates the Detaction and Diagnosis of Epileptic Sectores	Bignat Processing Toutoux ¹⁰⁰ Werenet Toutour ¹⁰⁰
		Using MATLAB and Simulink for Devices	Medical		

Omiting in Endoscopic Surgical Stepler

Prototype Using Model-Besid Design

Battelle Neural Brynase Technology Restores

Movement to a Panalyzed Marc's Arm and

Hand.

Subwen

E

Medviso Repeives FDA 510(k) Approval and

CE Marking for Cardiovensider Anelysis

Evolution of Deep Learning in MATLAB

2016	2017	2018	2019	2020
CNN's	Name Change	Examples	Reinforcement Learning	Deep Learning Data Sets
Pretrained Models Caffe Importer	 Neural Network Toolbox to Deep Learning Toolbox Algorithms LSTM's 	 Signal Processing Audio Text Analytics 	 Algorithms Automatic Differentiation Custom Training Loops Weight Sharing 	Apps Experiment Manager Examples 5G Communications
	 Directed Acyclic Graphs Multi-GPU Training Code Generation GPU Coder 	 Wavelet Scattering Code Generation MATLAB Coder C++ Apps 	Big Image Examples GANs Siamese Network	Over 200+ examples Algorithms Point Cloud Code Generation
	Apps • Image Labeler Interoperability • TensorFlow-Keras Importer	 Deep Network Designer Video Labeler Audio Labeler Interoperability ONNY Support 	 Autoencoders 3-D support Explainable Al Occlusion 	Quantization
			Code Generation MATLAB Coder (ARM) Anns	

Signal Labeler

📣 MathWorks

is a **Leader** in the Gartner Magic Quadrant for 2020 Data Science and Machine Learning Platforms Figure 1. Magic Quadrant for Data Science and Machine Learning Platforms



*Gartner Magic Quadrant for Data Science and Machine Learning Platforms, Peter Krensky, Erick Brethenoux, Jim Hare, Carlie Idoine, Alexander Linden, Svetlana Sicular, 11 February 2020.

This graphic was published by Gartner, Inc. as part of a larger research document and should be evaluated in the context of the entire document. The Gartner document is available upon request from MathWorks. Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, express or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.



Further Learning & Teaching

- Deep Learning Onramp
 - 2 hr online tutorial
- Deep Learning Workshop
 - 3 hr hands on session
 - Contact us to schedule
- Deep Learning Training
 - 16 hr in depth course
 - Online or Instructor Lead
- <u>Teaching Deep Learning with</u>
 <u>MATLAB</u>
 - Curriculum support







MathWorks Engineering Support



Training



Guided Evaluations



Onsite Workshops



Consulting



Technical Support





The Platform

MATLAB, Simulink, and over 100 add-on products for specialized applications



Helping you build an agile workforce today and preparing tomorrow's engineers



From onboarding and implementation to solving advanced engineering challenges