Computational intelligence - first part

conf.dr.ing. Anca Marginean

September 2019

Inteligenta Computationala

Computational intelligence (CI) usually refers to the ability of a computer to learn a specific task from data or experimental observation (wikipedia)

Computational Intelligence (CI) is the theory, design, application and development of biologically and linguistically motivated computational paradigms. Traditionally the three main pillars of CI have been Neural Networks, Fuzzy Systems and Evolutionary Computation. However, in time many nature inspired computing paradigms have evolved. (Computational Intelligence Society)

Computational intelligence is the study of the design of intelligent agents.

Similar to the concept of AI, there is no common accepted definition of CI.

Scurta istorie a Inteligentei Artificiale

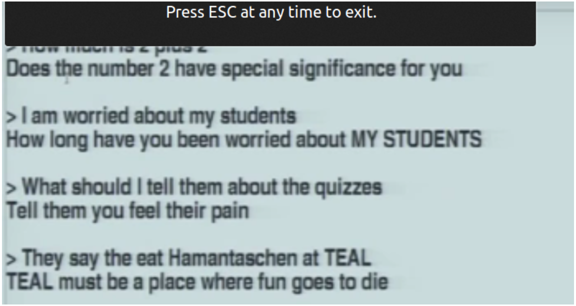
- 1940-1950: Inceputurile
  - 1943: McCulloch si Pitts: circuit boolean ca model al creierului
  - 1950: Turing's article "Computing Machinery and Intelligence"
    Turing machine is a mathematical *model* of computation that defines an abstract machine, which manipulates symbols on a strip of tape according to a table of rules.
    Turing idea: Child programme:"Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulated the child's?

Scurta istorie a Inteligentei Artificiale

- 1940-1950: Inceputurile
  - 1943: McCulloch si Pitts: circuit boolean ca model al creierului
  - 1950: Turing's article "Computing Machinery and Intelligence"
- 1950 - 70: Excitement: Look, Ma, no hands!
  - 1950s: Programe AI: Samuel's checkers program, Newell si Simon's Logic Theorist, Gelernter's Geometry Engine
  - 1956: intalnirea de la Dartmouth: a fost adoptat termenul de "Artificial Intelligence" John McCarthy
  - 1960: Eliza, Analogy, Vision

Scurta istorie a Inteligentei Artificiale

- 1940-1950: Inceputurile
  - 1943: McCulloch si Pitts: circuit boolean ca model al creierului
  - 1950: Turing's article "Computing Machinery and Intelligence"
- 1950 - 70: Excitement: Look, Ma, no hands!
  - 1950s: Programe AI: Samuel's checkers program, Newell si Simon's Logic Theorist, Gelernter's Geometry Engine
  - 1956: intalnirea de la Dartmouth: a fost adoptat termenul de "Artificial Intelligence" John McCarthy
  - 1960: Eliza, Analogy, Vision



1, 2

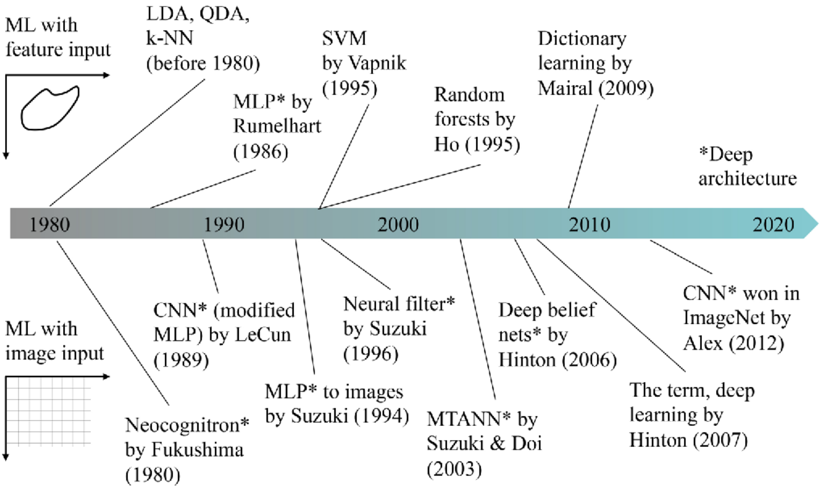
Scurta istorie a Inteligentei Artificiale

- 1940-1950: Inceputurile
  - 1943: McCulloch si Pitts: circuit boolean ca model al creierului
  - 1950: Turing's article "Computing Machinery and Intelligence"
- 1950 - 70: Excitement: Look, Ma, no hands!
  - 1950s: Programe AI: Samuel's checkers program, Newell si Simon's Logic Theorist, Gelernter's Geometry Engine
  - 1956: intalnirea de la Dartmouth: a fost adoptat termenul de "Artificial Intelligence" John McCarthy
  - 1960: Eliza, Analogy, Vision
  - 1965: Robinson's complete algorithm for logical reasoning
  - 1966: "the spirit is willing, but the flesh is weak" - "the vodka is good but the meat is rotten"

Scurta istorie a Inteligentei Artificiale

- 1940-1950: Inceputurile
  - 1943: McCulloch si Pitts: circuit boolean ca model al creierului
  - 1950: Turing's article "Computing Machinery and Intelligence"
- 1950 - 70: Excitement: Look, Ma, no hands!
  - 1950s: Programe AI: Samuel's checkers program, Newell si Simon's Logic Theorist, Gelernter's Geometry Engine
  - 1956: intalnirea de la Dartmouth: a fost adoptat termenul de "Artificial Intelligence" John McCarthy
  - 1960: Eliza, Analogy, Vision
- 1970 - 90: Abordarea bazata pe cunostinte
  - 1969 - 79: Early development of knowledge-based systems
  - 1980 - 88: Expert systems industry booms
  - 1988 - 93: "AI Winter"
- 1990 -: Abordarea statistica
  - Resurgence of probability, focus on uncertainty
  - Deep Blue - 1997 chess Kasparov
  - General increase in technical depth
  - Agents and learning systems... "AI Spring"
- 2000 : Avant si omniprezenta
- 2013 - Deep learning

Hot topic - deep learning history (for medical images)



Organizare propusa

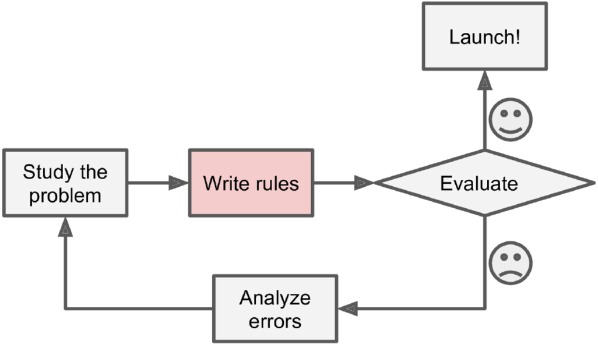
- Prima zi:
  - Machine learning clasic
  - Probleme generale ale ML,
  - Metrici de performanta
  - Diverse exemple in scikit si Matlab
- A doua zi:
  - Deep learning: retele neuronale, retele recurente, aplicatii
  - Exemple in tensorflow si Matlab
  - Exemplu Signal trading - comparatie
- A treia zi:
  - Deep learning in NLP - word embedding, text generation;
  - Vizualizarea retelelor pentru a intelege ce invata - exemplul OCT
  - Reinforcement Learning
  - Genetic Algorithm

Obiective generale ale cursului

- concepte generale ale paradigmelor CI prezentate
- explorare diverse probleme ML
- exemple de aplicare

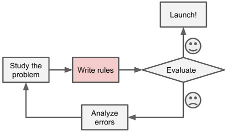
Ce e Machine Learning?

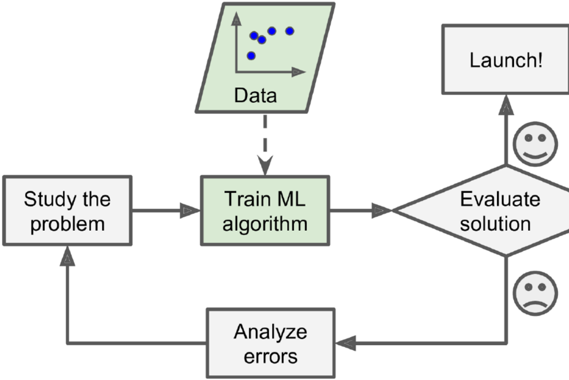
- "Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed" (Arthur Samuel 1959)
  Clasic programming:



Ce e Machine Learning?

- "Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed" (Arthur Samuel 1959)

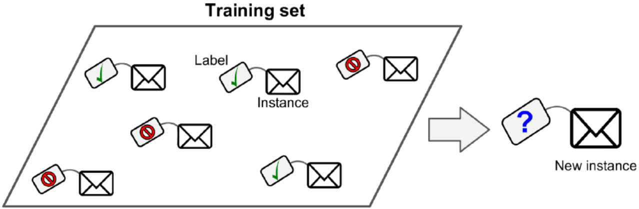




Tipuri de sisteme ML

Criteriu 1: sunt antrenate cu supervizare umana

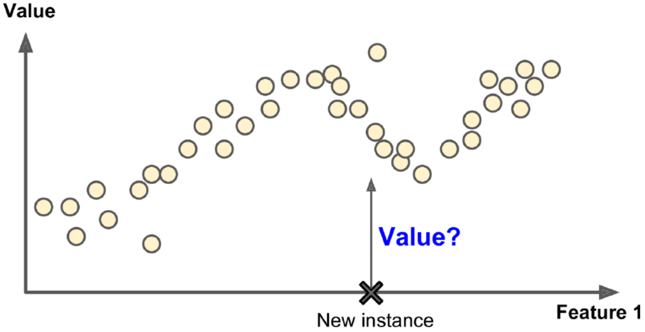
- Supervizate: clasificare,



Tipuri de sisteme ML

Criteriu 1: sunt antrenate cu supervizare umana

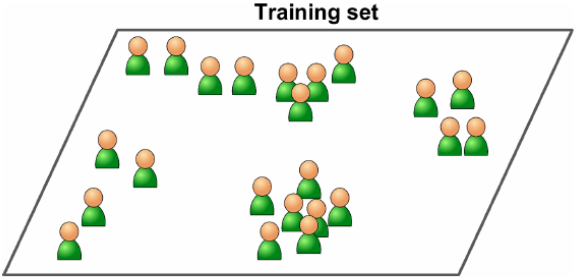
- Supervizate: clasificare, regresie



Tipuri de sisteme ML

Criteriu 1: sunt antrenate cu supervizare umana

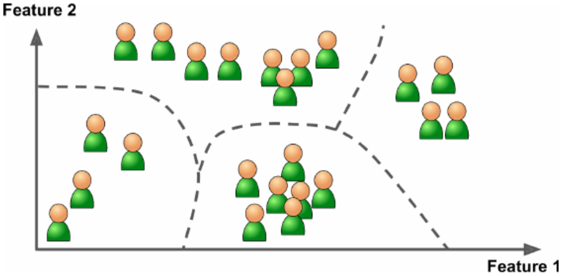
- Supervizate: clasificare, regresie
- Nesupervizate - datele de antrenare sunt ne-etichetate



Tipuri de sisteme ML

Criteriu 1: sunt antrenate cu supervizare umana

- Supervizate: clasificare, regresie
- Nesupervizate - datele de antrenare sunt ne-etichetate

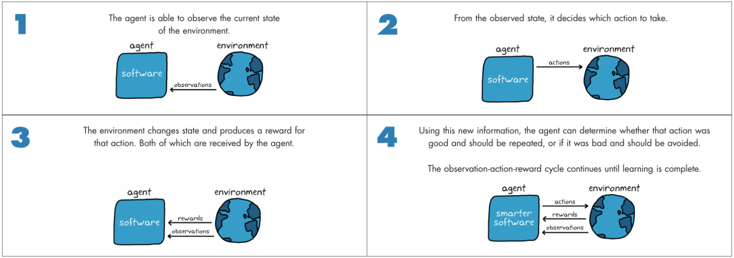


- Semi-supervizate

Tipuri de sisteme ML

Criteriu 1: sunt antrenate cu supervizare umana

- Supervizate: clasificare, regresie
- Nesupervizate - datele de antrenare sunt ne-etichetate
- Semi-supervizate
- Reinforcement Learning (invatare ranforsata)



Algoritmi invatare supervizata

- k-NN - k-nearest neighbors
- Linear Regression
- Logistic Regression
- Decision Trees, Random Forest
- Support Vector Machines (SVM)
- Retele neuronale

Algoritmi invatare nesupervizata

- Clustering: k-Means, Expectation Minimization
- Reducerea dimensionalitatii: Principal Component Analysis (PCA), t-distributed Stochastic Neighbor Embedding (t-SNE)
- Reguli de asociere(association rules): apriori, frequent item sets

Tipuri de sisteme ML

Criteriu 2: Sunt capabile sistemele sa invete din date primite continuu

- Batch learning: sistemele nu invata incremental
  - offline learning - sistemul este antrenatr si pe urma lansat - deci va aplica doar ce a invatat
- Online learning: sistemele se antreneaza incremental
  - mini-batch - impartirea datelor in grupuri mici - solutie si pentru date de dimensiune mare

Tipuri de sisteme ML

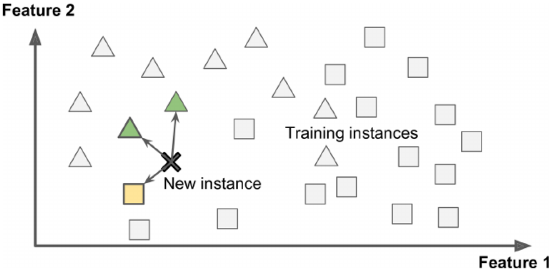
Criteriu 3: Cum generalizeaza sistemele? Dandu-se exemplele de antrenare, e necesar ca sistemul sa fie bun pe acestea, dar trebuie sa fie bun si pe date noi

- Instance-based learning - sistemul invata pe de rost datele si generalizeaza date noi prin diverse masuri de similaritate
- Model-based learning - sistemul invata un model pe care il poate aplica ulterior pentru a prezice (to predict) fara datele din care a invatat

Tipuri de sisteme ML

Criteriu 3: Cum generalizeaza sistemele? Dandu-se exemplele de antrenare, e necesar ca sistemul sa fie bun pe acestea, dar trebuie sa fie bun si pe date noi

- Instance-based learning - sistemul invata pe de rost datele si generalizeaza date noi prin diverse masuri de similaritate

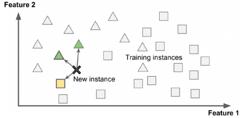


- Model-based learning - sistemul invata un model pe care il poate aplica ulterior pentru a prezice (to predict) fara datele din care a invatat

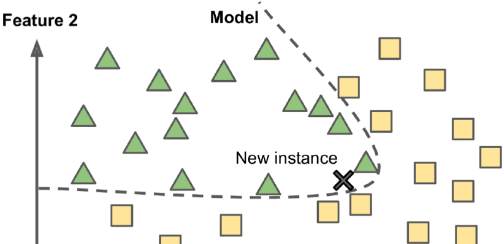
Tipuri de sisteme ML

Criteriu 3: Cum generalizeaza sistemele? Dandu-se exemplele de antrenare, e necesar ca sistemul sa fie bun pe acestea, dar trebuie sa fie bun si pe date noi

- Instance-based learning - sistemul invata pe de rost datele si generalizeaza date noi prin diverse masuri de similaritate



- Model-based learning - sistemul invata un model pe care il poate aplica ulterior pentru a prezice (to predict) fara datele din care a invatat

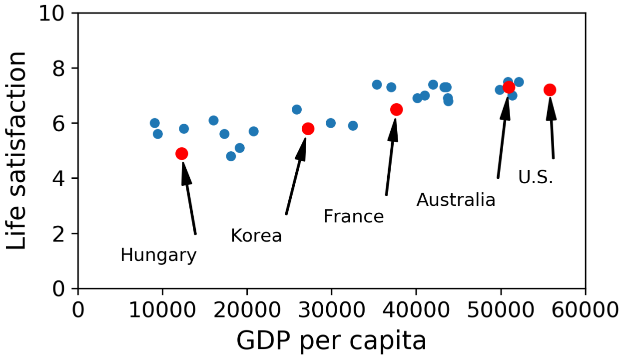


Exemplu de model

Can we say which is the life satisfaction of a country for which we know GDP per capita?

OECD: Better life Index - Life satisfaction

PIB (GDB per capita) de pe site-ul IMF



Time for notebook: 1. GDP - Model based learning

Principalele provocari ale ML

- Date insuficiente
- Date nereprezentative
- Calitate slaba a datelor - (ex. date lipsa)
- Caracteristici (features) irelevante
- Probleme ale invatarii: overfitting, underfitting

No free lunch theorem

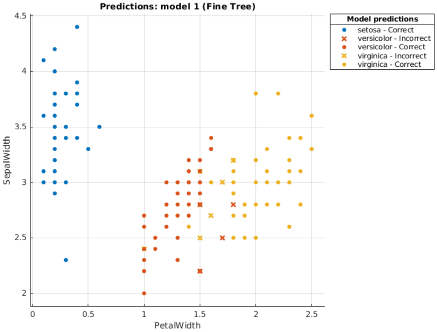
Ce e un model? este o versiune simplificata a observatiilor $\Rightarrow$ asumptii

- (NFL)In absenta oricarei asumptii, nu exista niciun motiv pentru a prefera un model in defavoarea altuia (1996).

Pentru unele date, modelul liniar este bun, pentru altele cel neuronal. Nu exista niciun model care este garantat apriori ca va functiona mai bine.

Exemplu in matlab - Clasificarea florilor de iris

- incarcare date
- vizualizare
- alegere model
- train
- vizualizare rezultate
- schimbare model/train
- schimbare features (caracteristici), alegere model, train
- exportare model
- utilizare model pentru a prezice

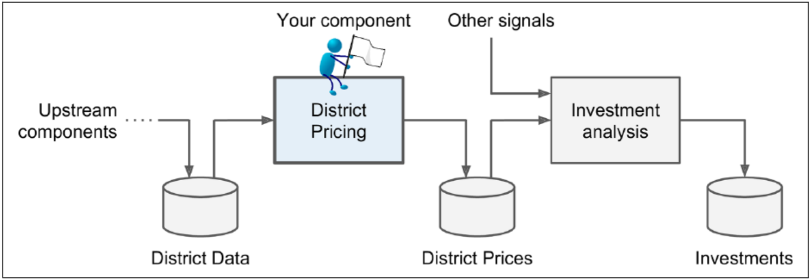


Pauza

End-to-End ML project

1. Look at the big picture :)
2. Obtinerea datelor
3. Descoperirea si vizualizarea datelor
4. Pregatirea datelor pentru algoritmii ML
5. Selectarea modelului si antrenarea lui
6. Fine-tuning al modelului

Look at the big picture

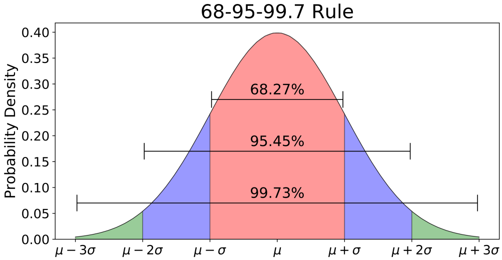


Alegerea unei masuri a performantei

Tipic pentru regresii: Root Mean Square Error (RMSE)

$$RMSE(X, h) = \sqrt{\frac{1}{m} \sum (h(x^{(i)}) - y^{(i)})^2}$$

- masoara deviatia standards a erorii in predictie



- 68% din predictii vor fi la $distanta = 1$ *deviatia standard* de valoarea reala, 95% la $distanta = 2 *$ *deviatia standard*, 99.7% la $distanta = 3 *$ *deviatia standard*

Alegerea unei masuri a performantei - MAE

Alternativa la

$$RMSE(X, h) = \sqrt{\frac{1}{m} \sum (h(x^{(i)}) - y^{(i)})^2}$$

Mean absolute error (MAE) - mai buna in cazul existentei valorilor outlier

$$MAE(X, h) = \frac{1}{m} \sum |h(x^{(i)} - y^{(i)})|$$

Time for Notebook

2. End-to-end Process Californian houses

   - Obiective: Pasii principali in cadrul unui proces care implica ML

Matlab: aplicarea invatarii pe aceleasi date in Matlab (cu/fara noile caracteristici, cu cross-validation) *original_housing.csv*, *processed_californian_houses.csv*

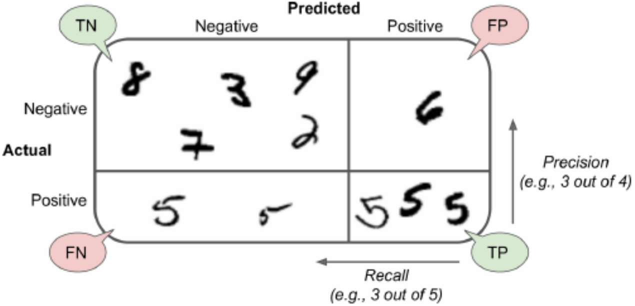
Clasificare - Classification

- Clasificare = predictia unei clase
- Exemplu = recunoasterea unei cifre scrise de mana MNIST

- Obiective:
  - Metrici de performanta pentru evaluare
    Observatie: De obicei, evaluarea e mai greu de facut decat la regresie
  - Seturi de date nebalansate

Tehnici folosite in evaluare

- Acuratete: de ce nu e suficienta acuratetea?
- Cross-validation
- Matricea de confuzie (confusion matrix)
  Pentru recunoasterea lui 5:



- Precision/recall

$$Precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

Precision vs Recall

- F-measure - media armonica intre precizie si recall

$$F_1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} = 2 \times \frac{precision \times recall}{precision + recall}$$

- care sunt cazurile extreme pt precision/recall?
- Exemplu 1: sistem care prezice video-uri safe pentru copii: precizie mare? sau recall mare?
- Exemplu 2: sistem care prezice daca cineva e hot de pe camera de supraveghere

Precision vs Recall

- F-measure - media armonica intre precizie si recall

$$F_1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} = 2 \times \frac{precision \times recall}{precision + recall}$$

- care sunt cazurile extreme pt precision/recall?
- Exemplu 1: sistem care prezice video-uri safe pentru copii: precizie mare? sau recall mare?
  Sistem care respinge multe video-uri chiar daca sunt safe, dar ce pastreaza sigur e safe $\Rightarrow$ precizie mare si recall mic
- Exemplu 2: sistem care prezice daca cineva e hot de pe camera de supraveghere

Precision vs Recall

- F-measure - media armonica intre precizie si recall

$$F_1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} = 2 \times \frac{precision \times recall}{precision + recall}$$

- care sunt cazurile extreme pt precision/recall?
- Exemplu 1: sistem care prezice video-uri safe pentru copii: precizie mare? sau recall mare?
- Exemplu 2: sistem care prezice daca cineva e hot de pe camera de supraveghere
  Sistem care identifica toate cazurile de furt, chiar daca incadreaza si situatii de nefurt: adica 30% precis, dar cu 99% recall de ex.

  De obicei, cresterea preciziei duce la scaderea recall-ului si invers $\Rightarrow$ compromis intre cele doua

Curba precizie vs recall

If someone says "let's reach 99%" precision", you should ask "at what recall?"

Time for start the notebook and come back for formulas
3. Performance Measures - MNIST

ROC curve si AUC (area under the curve

AUC-ROC curve - metrica de performanta pentru pb de clasificare la diverse praguri True positive rate = recall (sensitivity)

$$TPR/Recall/Sensitivity = \frac{TP}{TP + FN}$$

TNR = specificity

$$Specificity = \frac{TN}{TN + FP}$$

False positive rate - rata instantelor negative clasificate incorect ca positive TNR = specificity

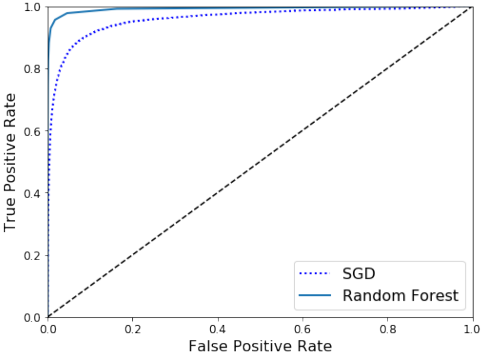
$$FPR = 1 - Specificity = \frac{FP}{TN + FP}$$

ROC curve - sensitivity vs 1 - specificity

AUC = 0.7 - sanse 70% ca modelul sa distinga intre clasa pozitiva si cea negativa

Compararea algoritmilor cu AUC

Un clasificator perfect ar avea ROC AUC = 1 Un clasificator random pur are ROC AUC = 0.5



return to notebook

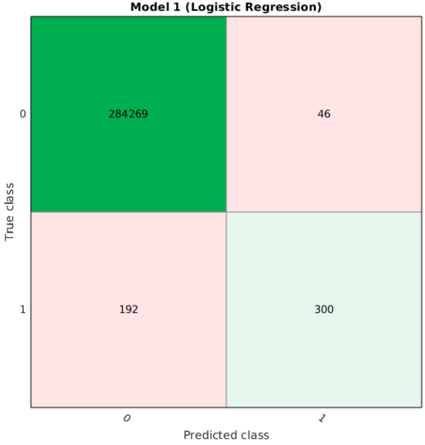
Fraud detection - matlab

Set de date: Wordline si Université Libre de Bruxelles

- Detect a 492 fraudulent transactions from 284,807 transactions in total
- Set de date de pe Kaggle
- Deep Learning Version - tensorflow
- readtable('examples/creditcard.csv')

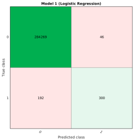
Acuratete? - Logistic regression

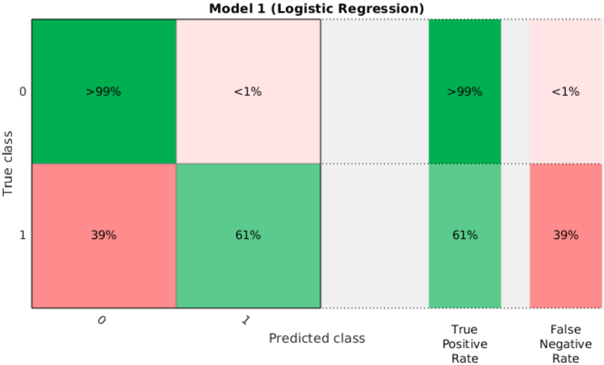
acuratete=99% (52sec GPU Quadro Pro 6000)



Acuratete? - Logistic regression

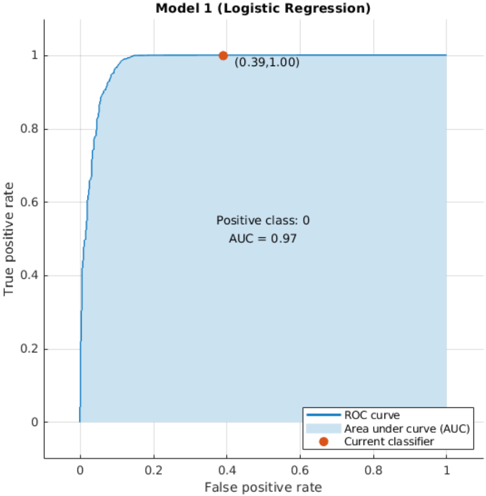
acuratete=99% (52sec GPU Quadro Pro 6000)





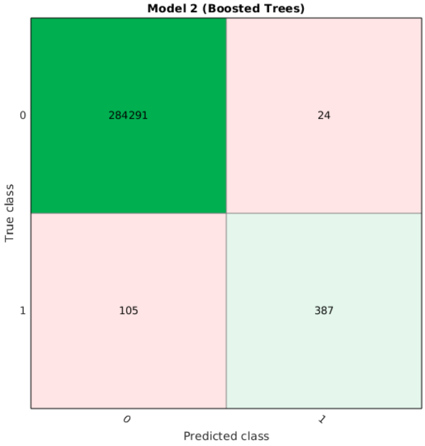
Acuratete? - Logistic regression

acuratete=99% (52sec GPU Quadro Pro 6000)



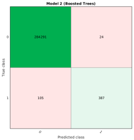
Acuratete? Boosted Decision Tree

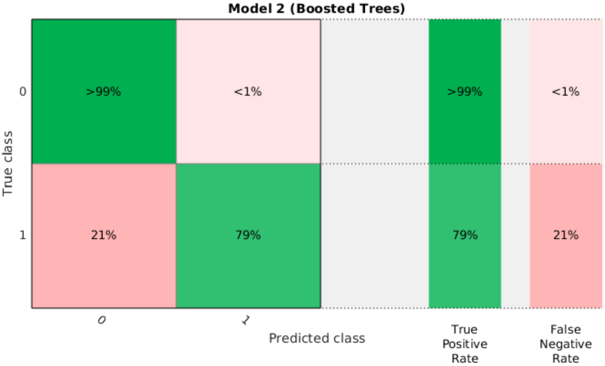
acuratete=100% (700sec GPU Quadro Pro 6000)



Acuratete? Boosted Decision Tree

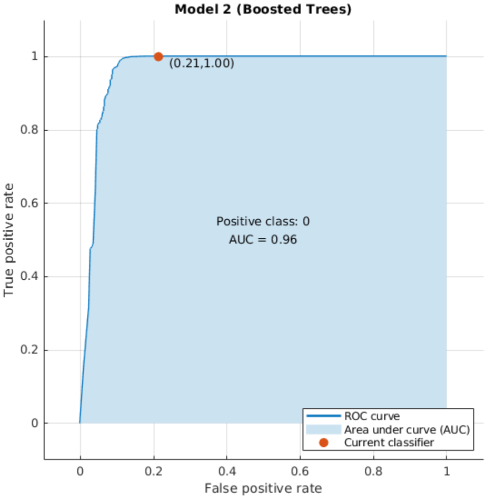
acuratete=100% (700sec GPU Quadro Pro 6000)





Acuratete? Boosted Decision Tree

acuratete=100% (700sec GPU Quadro Pro 6000)



Matlab: play with the saved model

- Load the saved model
- Load the data
- Get prediction probabilties
- Set different threshold and compute TPR, FPR
- see *different_threshold.m*

Exemplul va fi reluat in Tensorflow cu diverse variante de rezolvare a caracterului nebalansat al datelor

Bias vs variance

Time for notebook: 4. Linear Regression- Bias - Variance

- Cum alegem complexitatea unui model?
- Ce inseamna Regresie Liniara?
- Curba de invatare
- Care sunt sursele de eroare ale modelelor in generalizare?

ML frameworks

- Micorosft Azure
- Matlab toolboxes
- scikit-learn, weka, Apache Spark, R
- Tensorflow, Keras, Pytorch, MXNet, Gluon, DL4j

Credit rating Demo

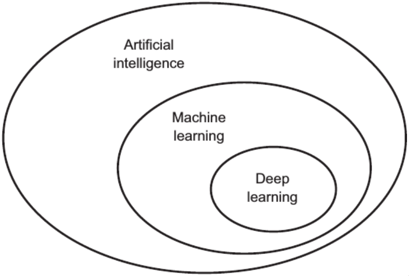
```
openExample('stats/creditratingdemo')
```

Open it as a Live Script
The data: "*CreditRating_Historical.dat*"
/home/nico/install_matlab/toolbox/stats/statsdemos

More matlab

Use Matlab_ex2.pdf

Ce e deep learning?

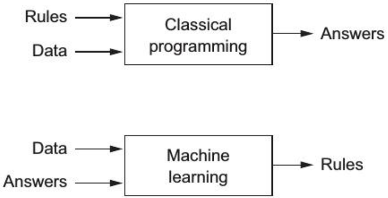


AI (anii 1950)

- automatizarea taskurilor intelectuale realizate in mod normal de oameni

... recomandari (youtube, carti, destinatii turistice), masina autonoma, rationare, identificarea automata a emailurilor spam, a fraudelor financiare

Sistemele de machine learning



- antrenat (trained), nu programat in mod explicit
- **invata**
  - cum sa combine input
  - pentru a produce predictii utile
  - pe data noi
- Ex: automatizarea taskului de descriere a unei poze
  - set de poze insotite de descriere
  - antrenare

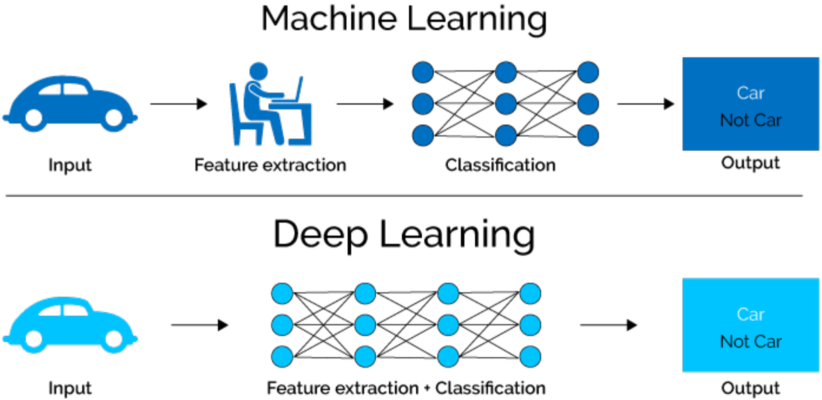
Deep learning (anii 2010)

Invatarea unor reprezentari din date care pune accentul pe

- invatarea unor nivele succesive de reprezentari
- layered representation learning

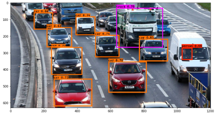
- retele neuronale (neural network)

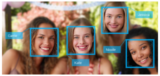
Deep Learning vs Classical Learning

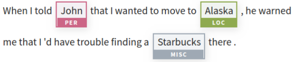


Rezultate deep learning

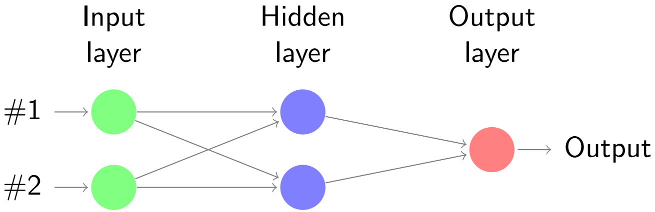
- clasificarea imaginilor (image classification) (near human-level)
- recunoasterea vorbii (speech recognition)
- recunoasterea scrisului de mana
- asistenti digitali Google Now, Amazon Alexa
- traduceri (imbunatatire)
- masina autonoma
- intelegerea textului
- Go (Alpha Go) (2016) (superhuman)



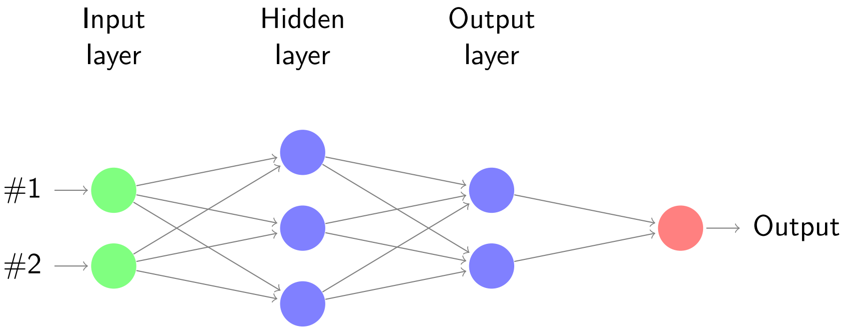




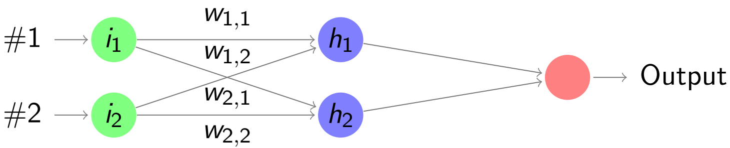
Retele neuronale



Retele neuronale



Retele neuronale



$$a_j = g(in_j)$$

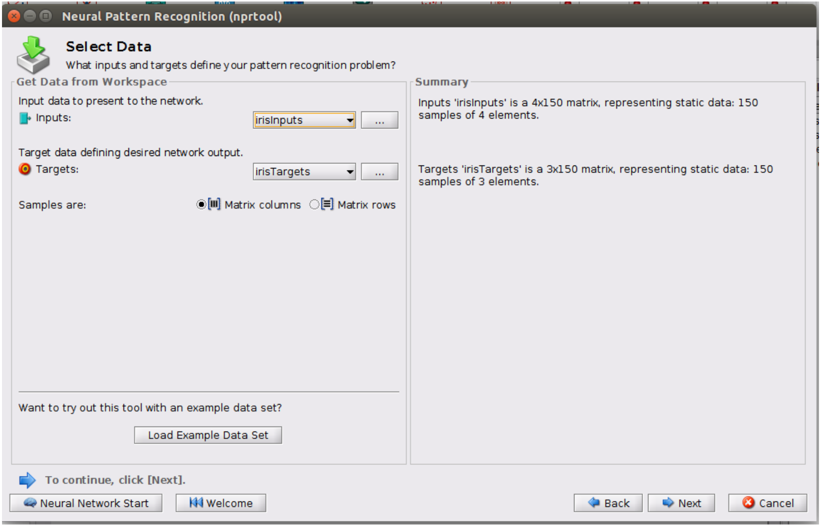
$$a_j = g\left(\sum_i (w_{i,j} * a_i)\right)$$

- a_i outputul neuronului i,
- $a_0 = 1$ bias
- $W_{i,j}$ ponderea legaturii dintre neuronul i (de pe nivelul anterior) si neuronul j (de pe nivelul curent)
- g functie de activare

exemplu: $a_{h_1} = g(w_{1,1} * a_i + w_{2,1} * a_2 + w_{0,1})$

Matlab: Neural Net Pattern Recognition

Part of Deep Learning Toolbox
Limited to a certain structure for the network



Matlab: neural net fit

Predict MPG for cars: *auto_mpg.csv*

1. Import the data with GUI
2. Create X and y

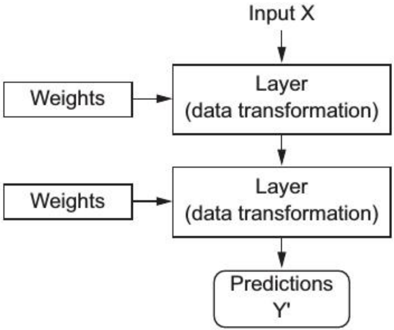
```
X = autompg(:,2:end)
y=autompg(:,1)

x=table2array(x)
y=table2array(y)
```

3. Start NeuralNet Fit

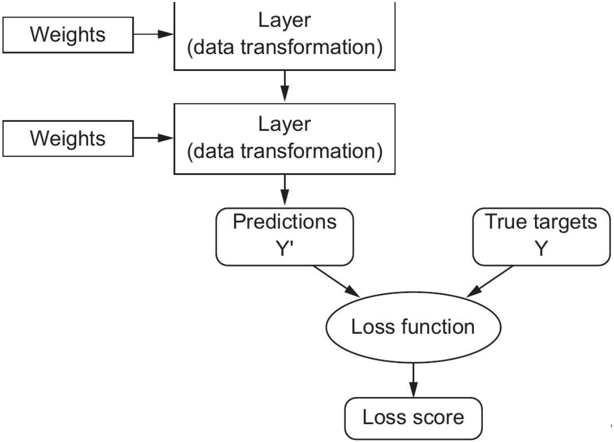
time for notebook

- activation functions and their derivatives
- why neural network can learn non linear functions?
- XOR with Tensorflow

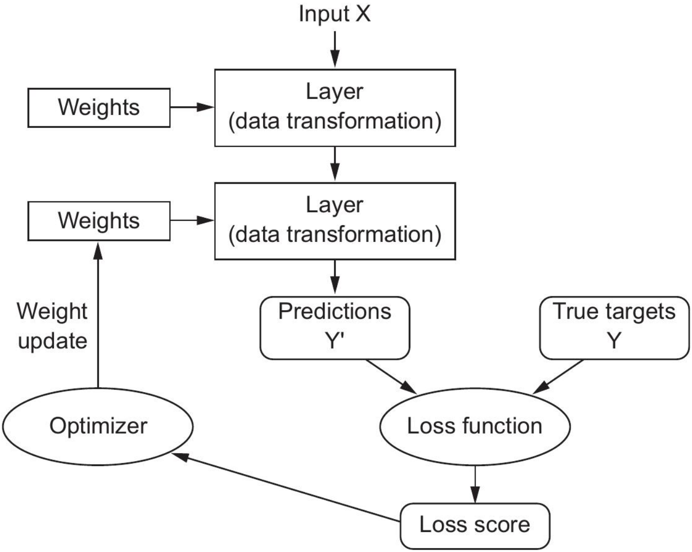
Cum functioneaza deep learning?



Cum functioneaza deep learning?



Cum functioneaza deep learning?



Loss

- Mean square error MSE

$$MSE = \frac{1}{N} \sum_{(x,y) \in D} (y - prediction(x))^2$$

- Log Loss

$$LogLoss = \sum_{(x,y) \in D} -y log(y') - (1 - y) log(1 - y')$$

- $(x, y) \in D$ - setul de date etichetate
- y - labelul exemplului x, unde x e un vector de caracteristici (features). Fiind clasificare, y e fie 0 fie 1
- y' - valoarea prezisa dat fiind x

Invatarea prin Gradient Descent

Regula de actualizare pentru perceptron:

$$w_j = w_j + \alpha * Loss * g'(in) * x_j$$

Back-propagation

- Nivelul de output

$$w_{j,i} = w_{j,i} + \alpha * a_j * \Delta_i$$

, unde $\Delta_i = Loss_i * g'(in_i)$

- Nivel ascuns Propagam eroarea:

$$\Delta_j = g'(in_j) \sum_i w_{j,i} \Delta_i$$

Actualizam ponderile

$$w_{k,j} = w_{k,j} + \alpha * a_k * \Delta_j$$

Neural Network Playground

https://playground.tensorflow.org/

Time for some examples in Tensorflow - MNIST with 10 classes

```
import tensorflow as tf
mnist = tf.keras.datasets.mnist

(x_train, y_train),(x_test, y_test) = mnist.load_data()
x_train, x_test = x_train / 255.0, x_test / 255.0

model = tf.keras.models.Sequential([
  tf.keras.layers.Flatten(input_shape=(28, 28)),
  tf.keras.layers.Dense(512, activation=tf.nn.relu),
  tf.keras.layers.Dropout(0.2),
  tf.keras.layers.Dense(10, activation=tf.nn.softmax)
])
model.compile(optimizer='adam',
              loss='sparse_categorical_crossentropy',
              metrics=['accuracy'])

model.fit(x_train, y_train, epochs=5)
model.evaluate(x_test, y_test)
```

Run it in COLAB: observe the output layer, change the dim of layers, observe the accuracy

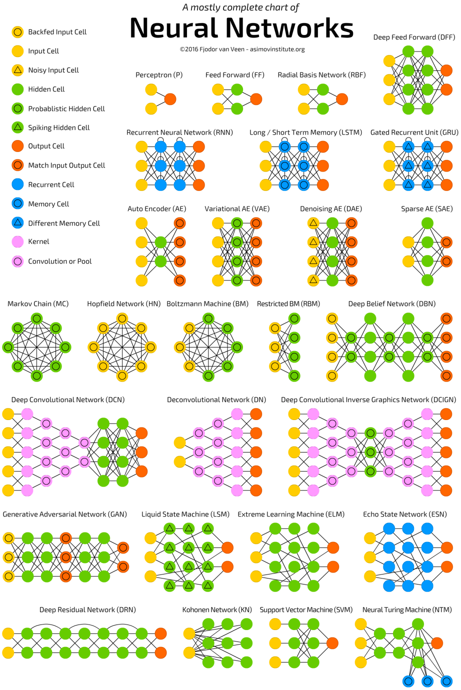
Examples in Tensorflow

Retele neuronale adanci Feed Forward

- More detailed Fashion MNIST - run it in **Colab**
- Basic text classification - with details about Text classification with Word embedding - vineri
- Iris example Good for Optimizers comparison

Main take aways for Deep FeedForward Networks

- Se pot aplica atat pentru regresie cat si pentru clasificare
- Au o structura bazata de layere
- Conteaza optimizatorul folosit


A mostly complete chart of
Neural Networks
©2016 Fjodor van Veen - asimovinstitute.org
Backfed Input Cell
Input Cell
Noisy Input Cell
Hidden Cell
Probablistic Hidden Cell
Spiking Hidden Cell
Output Cell
Match Input Output Cell
Recurrent Cell
Memory Cell
Different Memory Cell
Kernel
Convolution or Pool
Perceptron (P)
Feed Forward (FF)
Radial Basis Network (RBF)
Deep Feed Forward (DFF)
Recurrent Neural Network (RNN)
Long / Short Term Memory (LSTM)
Gated Recurrent Unit (GRU)
Auto Encoder (AE)
Variational AE (VAE)
Denoising AE (DAE)
Sparse AE (SAE)
Markov Chain (MC)
Hopfield Network (HN)
Boltzmann Machine (BM)
Restricted BM (RBM)
Deep Belief Network (DBN)
Deep Convolutional Network (DCN)
Deconvolutional Network (DN)
Deep Convolutional Inverse Graphics Network (DCIGN)
Generative Adversarial Network (GAN)
Liquid State Machine (LSM)
Extreme Learning Machine (ELM)
Echo State Network (ESN)
Deep Residual Network (DRN)
Kohonen Network (KN)
Support Vector Machine (SVM)
Neural Turing Machine (NTM)

MatLab DeepNetwork Toolbox

1. Create Simple NN for MNIST - with CNN, but focus on layers, optimizers

```
openExample('nnet/
    TrainABasicConvolutionalNeuralNetworkForClassificationExam
    ')}
```

2. RNN
   - Sequence-to-sequence regression: Time series with RNN - chickenpox prediction

   ```
   openExample('nnet/
       TimeSeriesForecastingUsingDeepLearningExample')
   ```

   - Sequence classification - Vowels classification, small example on syntetic data

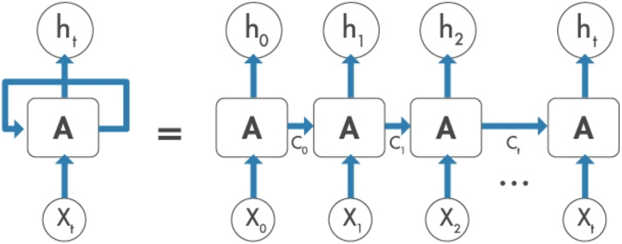
   ```
   openExample('nnet/
       ClassifySequenceDataUsingLSTMNetworksExample')
   or
   run lstm.m
   ```

   - Sequence-to-Sequence Classification - human activities

   ```
   openExample('nnet/
       SequencetoSequenceRegressionUsingDeepLearningExample
       ')
   ```

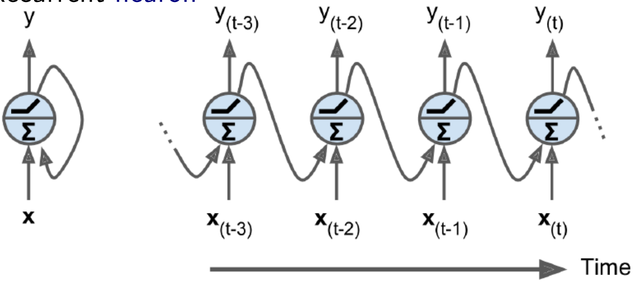
Tipuri de layere

- Dense Layer
- Dropout Layer
- Embedding Layer
- RNN layer: SimpleRNN, GRU, LSTM

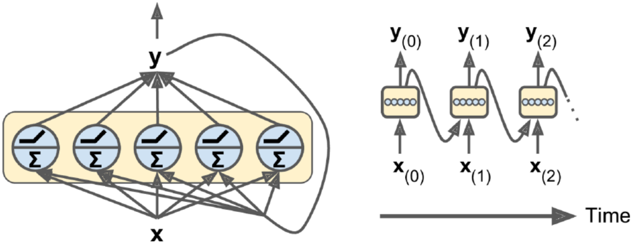


RNN details

- Recurrent neuron



- Recurrent layer



Day 3

- Previous days:
  - What is Machine learning? supervised/unsupervised learning/reinforcement
  - How can we build a ML model? get the data, preprocess the data, search for a good model, predict
  - How can we evaluate a model?
  - What is Deep Learning? Feed forward network, recurrent network, loss, sequence classification/regression

Day 3

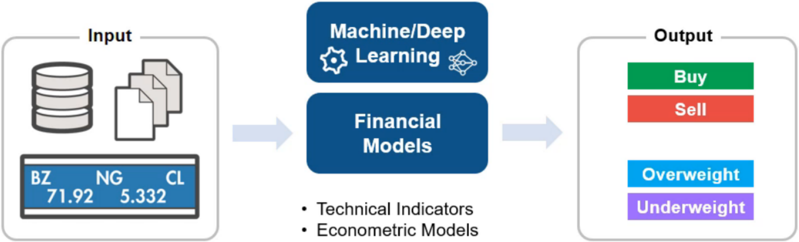
- Previous days:
  - What is Machine learning? supervised/unsupervised learning/reinforcement
  - How can we build a ML model? get the data, preprocess the data, search for a good model, predict
  - How can we evaluate a model?
  - What is Deep Learning? Feed forward network, recurrent network, loss, sequence classification/regression
- Today
  1. Trading Signal Classification
  2. More on Deep Learning: sequence2sequence regression, imbalanced data
  3. How to deal with text? bag of words, word embedding
  4. What's out there in terms of ready to use ML applications?
  5. Brief intro to Genetic Algorithm for optimizatio

Trading Signals Classification

Demo in matlab: *TradingSignals/Demo0_Overview.mlx* Webinar

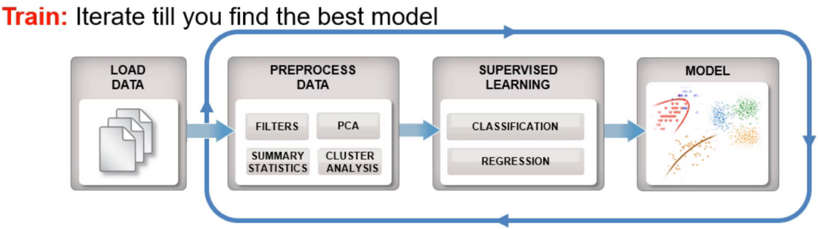
- How to classify trading signals?
- Three approaches: rule based, machine learning, deep learning

How to classify trading signals



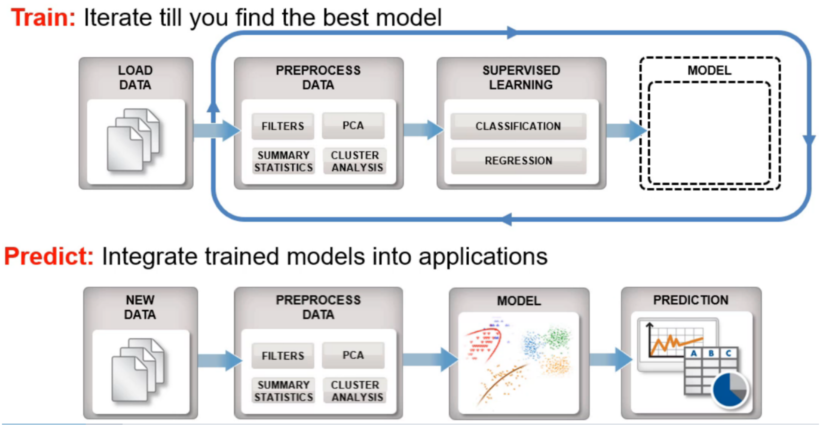
- Technical Indicators
- Econometric Models
- Rule-based Models

How can e model be created with Machine Learning?



The images are Matlab property

How can e model be created with Machine Learning?



The images are Matlab property

Required Toolboxes

- Datafeed toolbox - Retrieve Current and Historical Data Using Bloomberg
  - FRED = Federal Reserve Economic Data
  - $S\&P500$ index for 10 years
- Financial Toolbox
- Statistics and Machine Learning
- Deep Learning toolbox

Preprocess the data

- Data cleaning
- Feature engineering/Factor creation
  - Date time factors
  - Price related indicators

How we decide on the response?

- Use next day returns to generate

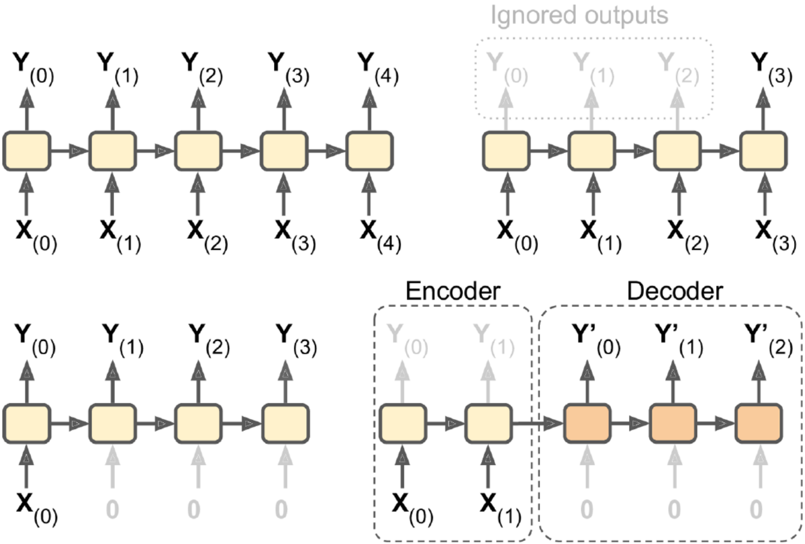
$$"Buy(1)" \, if \;\; returns > 0 and \; "Sell \; (0)" \;\; if \;\; returns \leq 0$$

.

Neural Network designer

Check the structure of the Demo3 network

Working with sequences - which approach was used in trading signals?



Example of sequence2sequence regression

There are more engines equiped with different sensors. Based on these values, predict how many cycles will the engine keep working before failing?

```
openExample('nnet/
    SequencetoSequenceRegressionUsingDeepLearningExample')
```

For some Help on understanding the example, see *Matlab_ex3.mat* file.

Representation of X / y for sequences

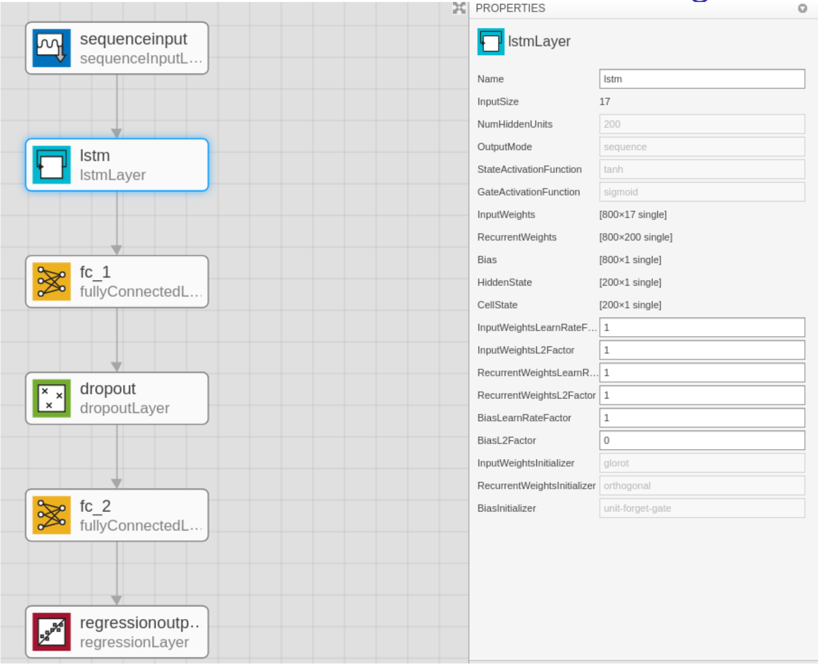
Predictors:

	time 0	time 1	...	time m
feature 1	$value_1^0$	$value_1^1$	...	$value_1^m$
feature 2	$value_2^0$	$value_2^1$	...	$value_2^m$
...				
feature n	$value_n^0$	$value_n^1$	...	$value_n^m$

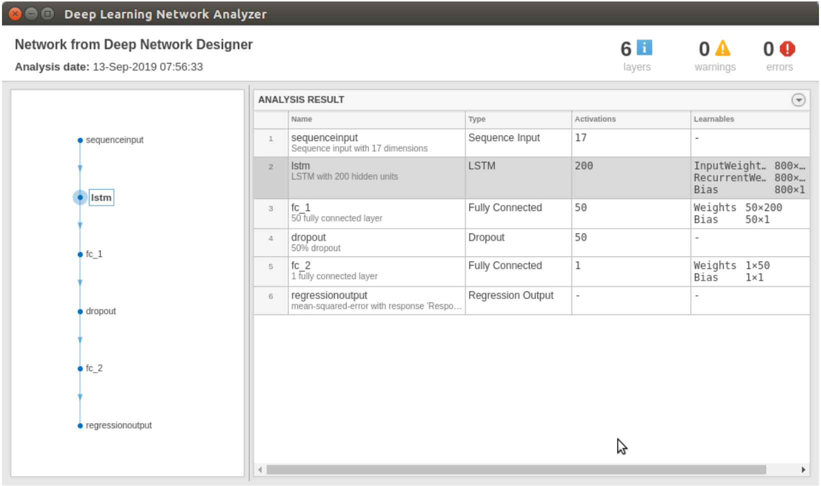
Response:

	time 0	time 1	...	time m
response 1	$value_1^0$	$value_1^1$	...	$value_1^m$
response 2	$value_2^0$	$value_2^1$	...	$value_2^m$
...				
response n	$value_n^0$	$value_n^1$	...	$value_n^m$

Structure of the network - Neural Network Designer



Structure of the network - analyse it

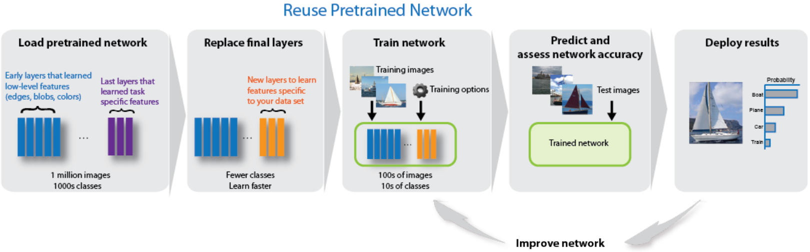


Transfer learning

1. Import pretrained network

```
net = googlenet
deepNetworkDesigner
```

2. Change the network last layers
3. Export network for training
4. Train network



Credit card Fraud detection - reloaded

Solutions for imbalanced data on credit card fraud detection

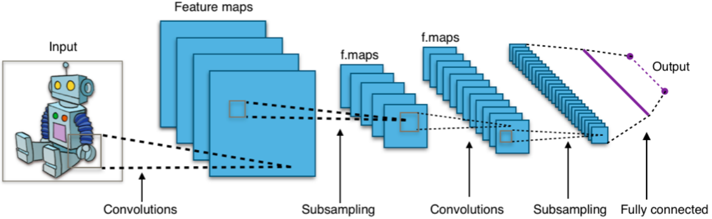
- Use the correct Metric for Evaluation
- Use class weights
- Oversampling the minority class

Classification of sequences with Deep Learning

Quick draw https://quickdraw.withgoogle.com/

Deep learning for image processing

More details on Friday



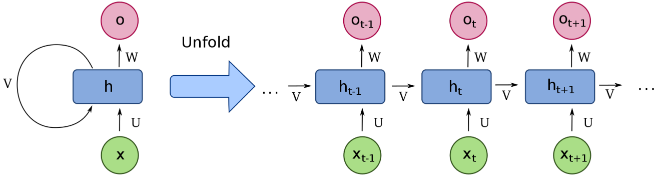
- Convolutional neural network- CNN
  - Image classification
  - Object detection
  - Image segmentation

https://aidemos.microsoft.com/face-recognition

```
./darknet detect cfg/yolov3.cfg yolov3.weights data/dog
    .jpg
```

Text processing

- Word embeddings
- Recurrent neural network - RNN

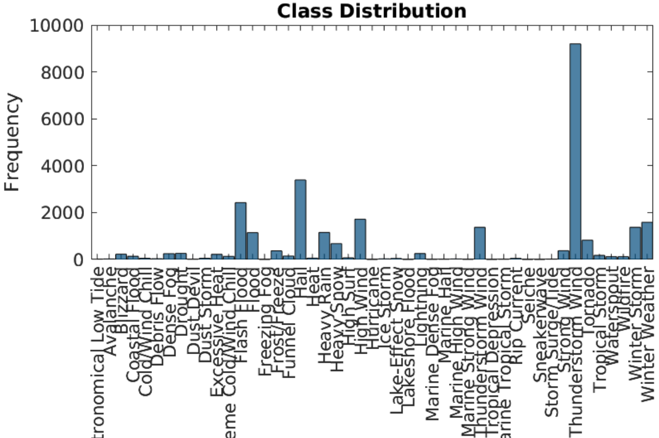


  - Allen nlp (https://demo.allennlp.org/reading-comprehension/NzQwMTEy)
  - Text translation, Text generation https://openai.com/blog/better-language-models/

```
./darknet rnn generate cfg/rnn.cfg shakespeare.weights
```

Text Analytics Toolbox

1. Bag of Words representation: text classification
   - bag-of-words model (also known as a term-frequency counter) - number of times that words appear in each document of a collection.
   - weather reports: *install_matlab/examples/textanalytics/weatherReports.csv*



Basic Text processing

```
documents = ["Oradea is a very nice city.", "You should
    really visit it!"]
documents = tokenizedDocument(documents);
documents = addPartOfSpeechDetails(documents);
details = tokenDetails(documents);
details.PartOfSpeech
documents = normalizeWords(documents,'Style','lemma');

% Erase punctuation.
documents = erasePunctuation(documents);

% Remove a list of stop words.
documents = removeStopWords(documents);

bag = bagOfWords(documents)

bag.Counts
```

Bag Of Words

"Oradea is a very nice city."

"You should really visit it!"

Doc	Oradea	nice	city	really	visit
1	1	1	1	0	0
2	0	0	0	1	1

Text Analytics Toolbox

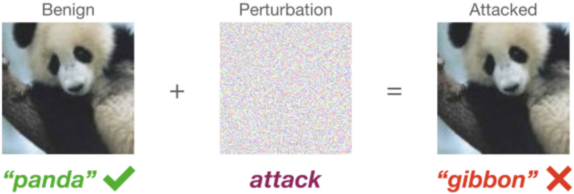
- 2. Word embeddings
- Requires also Text Analytics Toolbox Model for fastText English 16 Billion Token Word Embedding

emb = fastTextWordEmbedding
italy = word2vec(emb,"Italy"); rome = word2vec(emb,"Rome");
paris = word2vec(emb,"Paris");
word = vec2word(emb,italy - rome + paris)
??? word
Text classification with Word Embedding

Utilizare AI

- Microsoft AI Demos
  - Language understanding
  - Text analytics
  - Computer vision
  - Face and emotion recognition
- Microsoft Azure Cognitive services :
  - Demo, LUIS Home
  - Vision
- IBM Watson
  - Discovery
- Allen Institute applications
- Aristo news
- Aristo demo

Don't believe the short-term hype

- History tends to repeat:
  - 1960s: symbolic AI "Within a generation... the problem of creating 'artificial intelligence' will substantially be solved"
  - AI winter
  - 1980s: expert systems
  - AI winter
- 

Don't believe the short-term hype, but do believe in the long-term vision.

Example of news generated with a modern language model OpenAI

- It was given the sequence:
  A train carriage containing controlled nuclear materials was stolen in Cincinnati today. Its whereabouts are unknown.
- And the system automatically generated
  The incident occurred on the downtown train line, which runs from Covington and Ashland stations.
  In an email to Ohio news outlets, the U.S. Department of Energy said it is working with the Federal Railroad Administration to find the thief.
  "The theft of this nuclear material will have significant negative consequences on public and environmental health, our workforce and the economy of our nation," said Tom Hicks, the U.S. Energy Secretary, in a statement. "Our top priority is to secure the theft and ensure it doesn't happen again."
  The stolen material was taken from the University of Cincinnati's Research Triangle Park nuclear research site, according to a news release from Department officials.
  The Nuclear Regulatory Commission did not immediately release any information.
  According to the release, the U.S. Department of Energy's Office of Nuclear Material Safety and Security is leading that team's investigation.
  "The safety of people, the environment and the nation's nuclear stockpile is our highest priority," Hicks said. "We will get to the bottom of this and make no excuses.

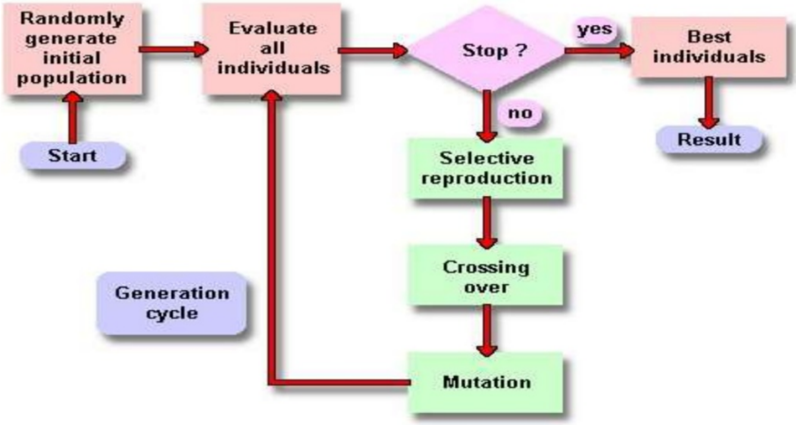
Recommended materials for Deep Learning

- Official documentation of Matlab Deep Learning toolbox
- Machine learning Crash Course (Google)
- Andrew NG - Coursera
- Tensorflow
- Francois Chollet - Deep Learning with Python-Manning Publications (2017)

Network visualization

Class Activation Maps OCT example

Genetic Algorithm for optimization



Find the minimum of a function with GA

```
openExample('globaloptim/gaoptionsdemo')
```

GA box2d Car

Box2d Car

Genetic Algorithm

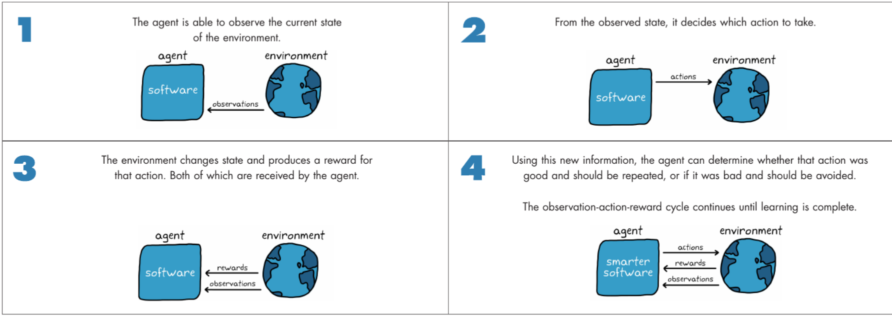
Knapsack problem: Given a set of n items numbered from 1 up to n, each with a weight w_i and a value v_i, along with a maximum weight capacity W...:

- find which items must be included in order to maximize the sum of the value, while keeping the weight under the maximum weight

$$fitness = \begin{cases} 0 & if \sum_{i \ selected} w_i > W \\ \sum_{i \ selected} v_i & if \sum_{i \ selected} w_i \leq W \end{cases}$$

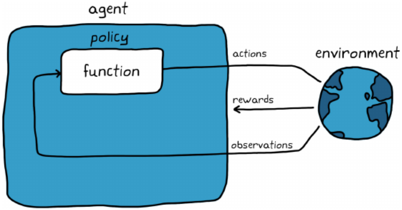
Matlab demo

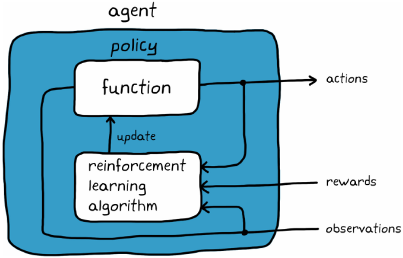
Reinforcement Learning



Policy function

Given an observation, the policy decides what action to take





Main terms in reinforcement

- Action(A): possible moves the agent can take
- State(S): situations returned by the environment
- Reward(R): immediate feedback from the environment to evaluate ????
- Policy(π): strategy that the agent employs to determine next action based on the current state
- Value(V): the expected long-term return with discount. $V\pi(s)$ - expected long-term return of the current state under the policy π
- Q-value: long-term return of the current state s and action a

Policy and value function representation

Policy - mapping that selects an action to take based on observations from the environment

- Depending on the type of RL agent: actor and critic function approximators -
- The actor - represents the policy that selects the best action to take.
- The critic - represents the value function that estimates the long-term reward for the current policy.
- Depending on the application and selected agent: policy and value functions can be defined using
  - deep neural networks,
  - linear basis functions,
  - look-up tables.

That's all for today

Thank you and please use *anca.marginean@cs.utcluj.ro* for any questions and suggestions.