

Referințe utile (I guess) pentru ML, ceva mai mult teoretic

Curs ML

Profesor: Radu Ionescu

[Cursurile](#) pe Drive (nu ajuta asa mult, dar am zis totusi sa le pun)

[Modele de examen rezolvate](#)

The basics

Pentru noțiuni de algebră liniară, probabilități, statistică sau referințe/formule la machine learning și neural networks: cartea [Deep Learning Book](#) de la MIT Press, disponibilă gratuit online.

Pentru videouri pe scurt și bine explicate despre orice legat de probabilități/statistică: [StatQuest with Josh Starmer](#)

MACHINE LEARNING

- [How Machines Learn](#) de la CGP Grey

Clasificatorul Naive Bayes

- [Naive Bayes, Clearly Explained](#) de la StatQuest

Măsurarea Performanței

Printre altele:

- [Machine Learning Fundamentals: Cross Validation](#) de la StatQuest
- [Classification: Accuracy | Machine Learning Crash Course](#) de la Google

- [Classification: Precision and Recall | Machine Learning Crash Course](#) de la Google
- Matricea de confuzie
- [Receiver Operating Characteristic \(ROC\) și Area Under Curve \(AUC\)](#)

Normalizare

- Normalizare L1/L2: [Gentle Introduction to Vector Norms in Machine Learning](#)
- Normalizare min-max:
[https://en.wikipedia.org/wiki/Feature_scaling#Rescaling_\(min-max_normalization\)](https://en.wikipedia.org/wiki/Feature_scaling#Rescaling_(min-max_normalization))
- Normalizare standard / standardizare:
[https://en.wikipedia.org/wiki/Feature_scaling#Standardization_\(Z-score_Normalization\)](https://en.wikipedia.org/wiki/Feature_scaling#Standardization_(Z-score_Normalization))

Metoda Celor Mai Aproapăți Vecini (KNN)

- [StatQuest: K-nearest neighbors, Clearly Explained](#)
- <https://cs231n.github.io/classification/#nearest-neighbor-classifier>

Blestemul Dimensionalității (Curse of Dimensionality)

[Curse of Dimensionality Definition](#)

Regresie Ridge (Ridge Regression)

O variantă a regresiei liniare la care se adaugă regularizare L2.

Descrisă foarte bine în documentația sklearn:

https://scikit-learn.org/stable/modules/linear_model.html#ridge-regression-and-classification

Mașini cu Vectori Suport (SVM)

- [Support Vector Machines, Clearly Explained!!!](#)
- Partea despre SVM din CS231n: <https://cs231n.github.io/linear-classify/>

Metode kernel (Kernel Methods) – sau bleac

- [The Kernel Trick](#), parte a cursului Introduction to Computer Vision de la Udacity
- Sunt explicate și în video-ul de la SVM-uri.

Funcții de pierdere (Loss Functions)

- [Loss Functions Explained](#) de Siraj Raval
- [Visual Information Theory](#) care oferă o prezentare grafică a cross-entropy

Coborârea prin gradient (Gradient Descent) – bleac

- [Gradient descent, how neural networks learn | Deep learning, chapter 2](#) de la 3Blue1Brown

Rețele Neuronale (Neural Networks) – mega bleac

- Articolul lui Karpathy [Hacker's guide to Neural Networks](#)
- Primul capitol din [Intro to Deep Learning with PyTorch](#)
- [But what is a Neural Network? | Deep learning, chapter 1](#) de la 3Blue1Brown
- Perspectiva geometrică a rețelor neuronale: [Neural Networks, Manifolds, and Topology](#)

Antrenarea rețelelor neuronale

- De ce trebuie să inițializezi random ponderile dintr-o rețea: [Initializing neural networks – deeplearning.ai](#)
- [Backpropagation](#)

Rețele Neuronale Convoluționale (Convolutional Neural Networks) – dauna morala

- [Conv Nets: A Modular Perspective](#) de pe blogul lui Christopher Olah
- Pentru a înțelege stride, padding, kernel size, etc: [Intuitively Understanding Convolutions for Deep Learning](#)

- [An intuitive guide to Convolutional Neural Networks](#)
- [CS231n](#) de la Stanford