On Data Science and Deep Learning: A student's curriculum point of view

Dimitrios Katsaros
University of Thessaly, Volos, Greece
dkatsar@uth.gr
September 14th, 2024

***Abstract*—This short note intends to provide answers – according to the author's angle of view – to what Data Science is, and moreover about what technical capabilities and knowledge a Deep Learning engineer should possess. This note was prepared in the context of the "Neuro-Fuzzy Computing" course in ECE@UTH.GR.**

I. What is data science

We can start with some obvious negations, arguing that data science is neither statistics, nor databases, and nor computational science; it is not even the union of all of them [1]. The future of data science is still unclear, since innovation in technologies may change the nature of the job and the future of data scientists [2]. So for the moment, rather than providing a generic and thus vague definition of what Data Science is [3], [4], we describe the *current technical* constituent pillars[1] of Data Science, which are the following (see Figure 1 and Figure 2): a) Distributed and Parallel Computing, b) Data Management, and c) "Learning".

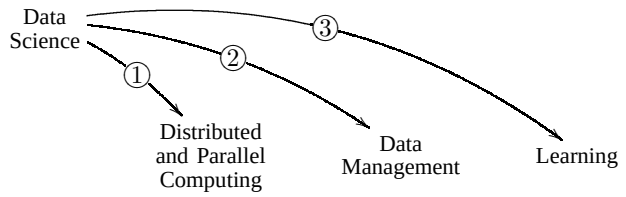

Fig. 1. The three pillars of data science.

□ "Distributed/Parallel Computing". One is expected to have solid knowledge or to be familiar with:

- ⇒ Cluster programming: popular approaches include MapReduce-like models, such as Hadoop, Spark, etc. In the parallel world, CUDA is a popular choice.
- ⇒ Elements of (the theory and practice of) distributed computing
  - ◇ Distributed algorithms and data structures: Indexing in distributed environments (Consistent Hashing, Bloom Filters, Distributed B^+-trees), key-value stores, distributed graph-theoretic algorithms (SP, MST, MDS, MIS).
  - ◇ The CAP theorem and its repercussions, connecting Consistency, Availability, and Partition tolerance of a distributed service, and associated notions of consistency, e.g., eventual consistency.
  - ◇ Leader election protocols.
  - ◇ Distributed consensus and protocols: Paxos, Raft.
  - ◇ The DCS theorem and its repercussions, connecting Decentralization, Consensus, Scale in blockchain design, and relevant protocols such as PoW, PoS, PoB, PoET, Algorand, etc.
  - ◇ Basics of wireless ad hoc networking (because of their use in distributed federated learning): backbones, clustering, transmission/compression,

By now, the astute reader will have recognized that part of the above comprise basic knowledge of *cloud computing*!

□ "Data Management". One is expected to have solid knowledge or to be familiar with:

- → Algorithms and their associated data structures.
  - ▷ The book "Introduction to Algorithms", by MIT Press is a perfect source of knowledge
- → Data types and data management systems
  - ▷ Relational model and SQL, NoSQL systems.
  - ▷ Storage and indexing for (single-/multi-)dimensional data, and for graph data.
  - ▷ Transaction processing.
  - ▷ Complex network data and network science.

□ "Learning". One is expected to have solid knowledge or to be familiar with:

- ↦ (Uni- and multi-)variate calculus, (multi-)linear algebra.
- ↦ (Uni- and multi-)variate probability and statistics.
  - – Sampling (stratified, weighted, reservoir, etc).
- ↦ Applied mathematical optimization
  - – Dynamic programming.
  - – Linear/integer programming.
  - – Un/constrained optimization
    - ∗ First order optimizers: Steepest descent and variants, AdaGrad, Adam, ADOPT, and variants, etc.
    - ∗ Second order optimizers: Conjugate gradient, Newton, Quasi-Newton (BFGS), AdaHessian, etc.
    - ∗ Karush-Kuhn-Tucker, and Frank-Wolfe methods.

[1] Security is embedded within the three pillars.

↦ Data mining
  - Association rules: Apriori and variants (on trees, sequences, etc), graph mining.
  - Clustering: k-means, DBSCAN and other significant density-based algorithms.
  - Classification: decision trees, nearest-neighbor, Bayesian, ensemble methods.
  - Outlier detection methods: statistical, density-based, clustering-based methods.

↦ Machine learning
  - Dimensionality reduction: PCA, EM.
  - Feature engineering and regularization.
  - Regression (linear, logistic).
  - Classification: Bayesian methods, decision trees, probabilistic methods, random forests, ensemble methods (e.g., GBM), nearest-neighbor methods.
  - Kernel methods: SVMs.
  - Clustering: k-means, EM.
  - Graphical models: HMMs.
  - Reinforcement Learning.

↦ Deep learning
  - Deep Multi-Layer Perceptrons.
  - Deep Convolutional Neural Networks.
  - Recurrent Neural Networks: GRU, LSTM and variants/hybrids.
  - Transformers.
  - Autoencoders.
  - Generative Neural Networks: LLMs.
  - Graph Neural Networks.
  - Deep Reinforcement Learning.
  - Spiking Neural Networks.

No-one can ever become an expert in all these fields; instead s/he will excel, say, in a couple of them. However, having *working knowledge* of as many as possible of the rest of the areas is necessary.

II. What is the difference between a machine learning and a deep learning engineer?

A Machine Learning engineer is responsible for designing, building, and deploying ML models that can learn and improve over time. They work with large datasets, develop algorithms, and use statistical and mathematical techniques to train models that can make predictions or decisions based on the input data. Machine Learning Engineers are also responsible for optimizing and improving the performance of these models, ensuring that they are scalable, efficient, and accurate (see Figure 3).

Machine Learning engineer responsibilities:

- Collect and preprocess large datasets.
- Develop and implement ML algorithms and models.
- Evaluate model performance and optimize for accuracy and efficiency.
- Deploy and maintain ML models in production environments.

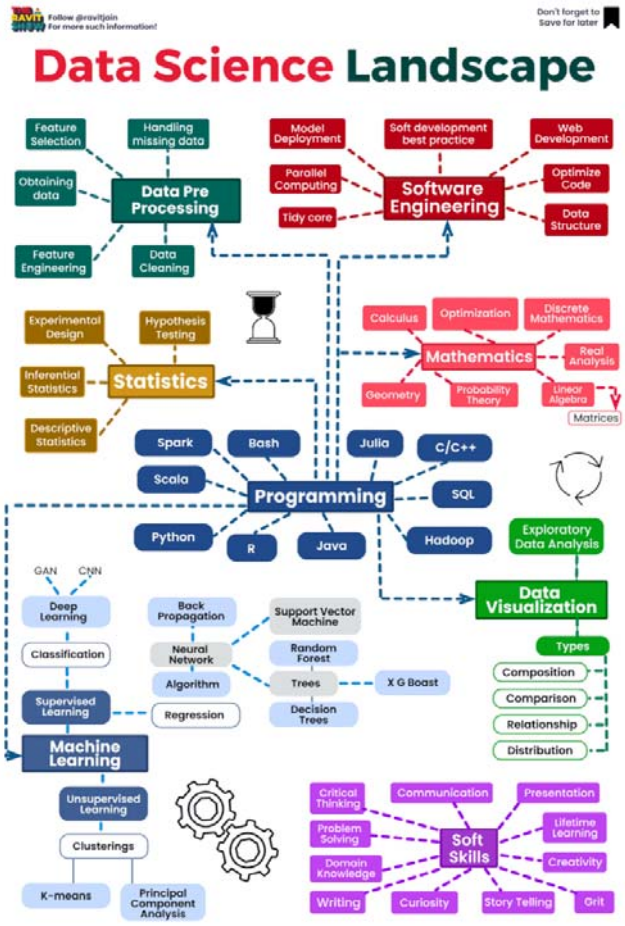

Fig. 2. The Data Science landscape (by Ravit Jain).

- Collaborate with data scientists, software engineers, and other stakeholders to develop solutions that meet business requirements.
- Stay up-to-date with the latest ML techniques, tools, and technologies.

Although there exist varying opinions about the technical skills a ML engineer should posses [5], we think that the following are appropriate:

- Proficiency in programming languages such as Python, R, C++ and Java.
- Strong understanding of statistics and probability theory.
- Experience with machine learning frameworks such as Scikit-learn, TensorFlow, PyTorch, MLflow.
- Knowledge of data preprocessing techniques such as data cleaning, feature scaling, and feature engineering.
- Familiarity with cloud computing platforms such as AWS, Azure, and Google Cloud.
- Excellent communication and collaboration skills.

A Deep Learning engineer is a specialized type of Machine Learning engineer who focuses on developing and deploying DL models. A deep learning engineer's duty is to be an expert

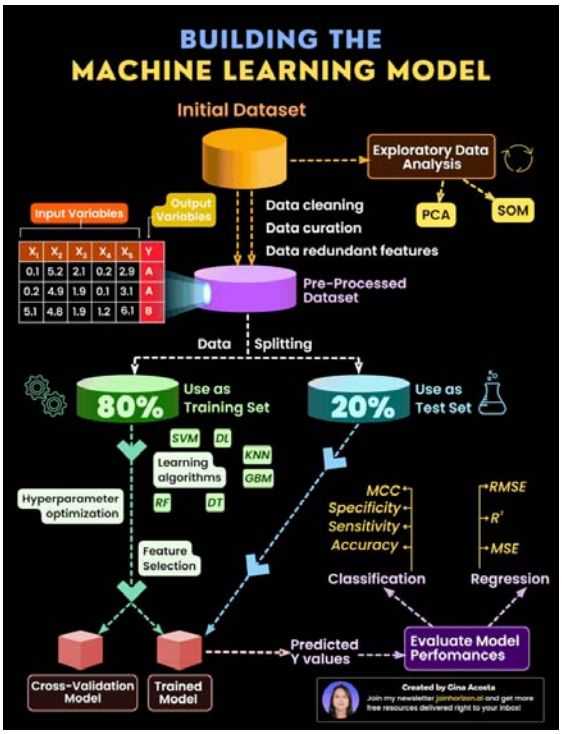

Fig. 3. Building the machine learning model (by Gina Acosta).

in the design and implementation of learning algorithms based on deep and complicated neural network topologies. Because the techniques utilized are more sophisticated theoretically, this is more technical work than that of a “traditional" machine learning engineer.

Deep Learning engineer responsibilities include:

- Design and implement neural network architectures for DL models.
- Train and optimize DL models using frameworks such as TensorFlow, Keras, and PyTorch.
- Fine-tune DL models for specific tasks such as image classification, object detection, NLP, etc.
- Develop custom neural network for specific needs.
- Deploy and scale DL models in production environments.
- Collaborate with data scientists, software engineers, and other stakeholders to develop DL solutions that meet business requirements.
- Stay up-to-date with the latest DL techniques, tools, and technologies.

Altough there exist varying opinions about the technical skills a DL engineer should posses [5], we think that the following are appropriate:

- Proficiency in languages such as Python, C++, CUDA.
- Strong understanding of linear algebra, calculus, and probability theory.
- Experience with deep learning frameworks such as TensorFlow, Keras, PyTorch, Deeplearning4j.
- Knowledge of natural language processing techniques.
- Familiarity with GPU programming and distributed processing.
- Excellent problem-solving and analytical skills.

As far as what programming languages are good to know, one can always consult the TIOBE index[2].

Deep Learning textbooks (according to the author’s preference) that your bookcase should have, include the following:

- Dive into Deep Learning [6].
- Neural Network Design [7].
- Deep Learning [8] (also, **in greek** by Kleidarithmos)
- Designing Machine Learning Systems [9].
- Understanding Deep Learning [10].

One can stay up-to-date with latest Deep Learning developments by reading "The State of AI Report"[3] and reading articles published in leading journals and conferences of the field:

(JNL) Machine Intelligence (Springer/Nature)
(JNL) IEEE Transactions on Neural Networks and Learning Systems
(JNL) IEEE Transactions on Artificial Intelligence
(JNL) Neural Networks (Elsevier)
(JNL) Neurocomputing (Elsevier)
(JNL) Neural Computing and Applications (Springer)
(JNL) Applied Intelligence (Springer)
(JNL) Neural Computation (MIT Press)
(JNL) ACM Transactions on Probabilistic Machine Learning
(JNL) ACM Transactions on Intelligent Systems and Technology
(JNL) Neural Processing Letters (Springer)
(CNF) Neural Information Processing Systems (NeurIPS)
(CNF) International Joint Conference on Neural Networks (IJCNN)
(CNF) International Conference on Learning Representations (ICLR)
(CNF) International Conference on Artificial Neural Networks (ICANN)

Deep learning articles appear also in IEEE TPAMI, JMLR, PMLR, ICML, CVPR, ECML/PKDD, ECAI, AAAI, ACM KDD, IEEE ICDM, SIAM SDM, etc. Many ground-breaking papers appear in topic-specific deep learning periodicals, e.g., IEEE Transactions on Machine Learning in Communications and Networking.

References

[1] C. Ley and S. P. A. Bordas. What makes Data Science different? A discussion involving Statistics2.0 and computational sciences. *International Journal of Data Science and Analytics*, 6:167–175, 2018.
[2] N. Ahmad and A. Hamid. Will Data Science outrun the data scientist? *IEEE Computer magazine*, 56(2):121–128, 2023.
[3] L. Cao. Data Science: A comprehensive overview. *ACM Computing Surveys*, 50(3), 2017.
[4] T. Ozsu. Data Science – a systematic treatment. *Communications of the ACM*, 66(7):106–116, 2023.
[5] A. Schwab-McCoy, C. M. Baker, and R. E. Gasper. Data Science in 2020: Computing, curricula, and challenges for the next 10 years. *Journal of Statistics and Data Science Education*, 29:S40–S50, 2021.
[6] A. Zhang, Z .C. Lipton, M. Li, and A. J. Smola. *Dive into Deep Learning*. Cambridge University Press, 2023.
[7] M. T. Hagan, H. B. Demuth, M. H. Beale, and O. De Jesus. *Neural Network Design*. Martin Hagan, 2nd edition, 2014.
[8] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. The MIT Press, 2016. Translated into greek by Ekdoseis KLEIDARITHMOS.
[9] C. Huyen. *Designing Machine Learning Systems*. O’Reilly Media, 2022.
[10] S. J. D. Prince. *Understanding Deep Learning*. The MIT Press, 2023.

[2]https://www.tiobe.com/tiobe-index/
[3]https://www.stateof.ai/