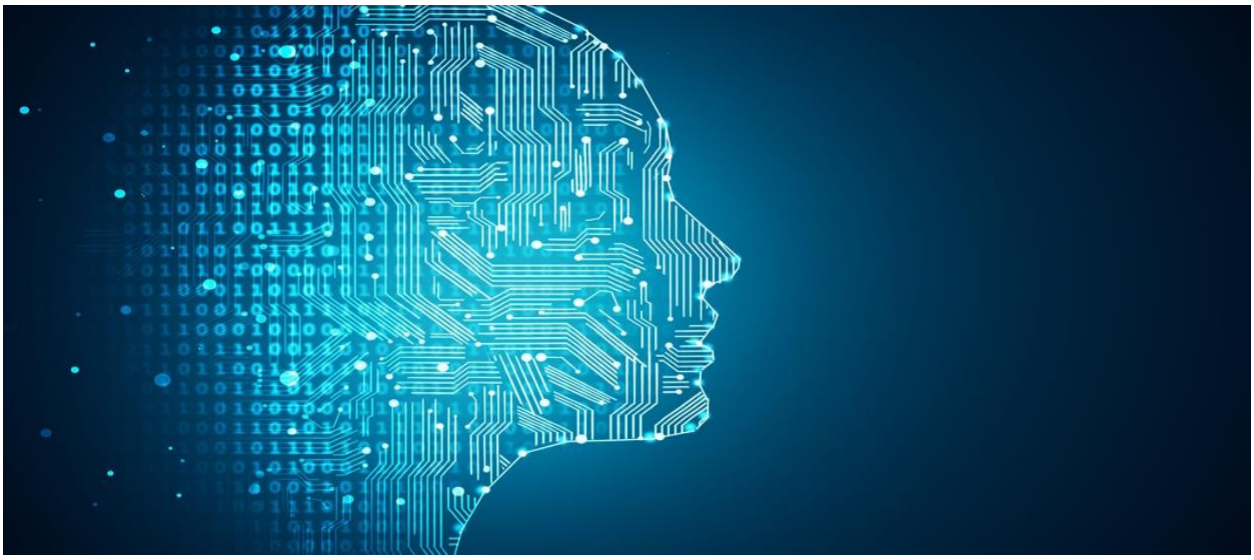

How to Become a Data Scientist?

A Guide

Author: Zaki Alawami

Date: October, 3030

Version: 3.0

Table of Contents

1. What is Data Science?

Data science is the discipline of gaining meaningful insights from data and creating data-driven solutions. Organizations accumulate and store vast amount of data, but this data, although valuable, has not always been recognized as a business asset in and of itself. Data, holistically, has hidden trends and patterns that if discovered can yield valuable insights which can lead to major optimizations in all facets of the business. Data science is ultimately the art of postulating questions or hypothesis that can be validated by thoroughly examining the data. Additionally, historical data in a given domain can be used to create predictive models of future events and behavior in that domain. This has many applications, such as improving the decision-making process, operational efficiency, value creation, organizational integration, safety, and cost optimization using the already existing yet untapped potential of the data assets.

Data science can also be considered as an multidisciplinary field that applies knowledge from various decision-making process, efficiency, value creation, horizontal integration, safety, and cost optimization using the already existing yet untapped potential of our data assets.
disciplines such as mathematics, statistics, computer programming, and domain-specific expertise. It has largely been enabled by the availability of vast amounts of data, the ubiquity of processing power, and innovations in data modeling algorithms.

2. How to Use this Guide?

This guide contains a description of how to become a Data Scientist. The guide covers the main competencies required and how they relate to Data science. This document can be used by organizations to establish a program to create data scientists or by individuals interested in becoming data scientists on their own. The program should include working on various real-world or simulated projects in order to maximize learning of the skills required to successfully become a Data Scientist.

The program required competencies are categorized in Three (3) levels: Basic, Intermediate, and Advanced (Table 1). Each competency level can be completed in about 6 months but it is flexible to accommodate the specific candidate available time, education, experience background, and therefore the missing qualifications needed. For example, if the candidate already has a computer science or computer engineering Bachelor Degree, then the candidate does not need to cover some of the Basic competencies such as those mentioned in 3.1 and 3.2.

The candidate should acquire the relevant certifications in each of the listed competencies. Candidates that complete an MS in Data Science should be considered certified for the covered competencies in the MS degree curriculum.

Level	Beginner	Intermediate	Advanced
Competencies	Computing Foundations	Linear Algebra	DevOps Practices
	Programming & Tools Foundations	Calculus	Reinforcement Learning
	Data Foundations	Big Data Foundations	Deep Learning II
	Database & Data Warehouse Foundations	Machine Learning	Parallel Computer Architectures and Programming
	Statistical Analysis	Deep Learning I	Applied Mathematics
	Probability Theory	Data Engineering	
	Data Analytics/Exploratory Data Analytics	Cloud computing	
	Data Visualization		
	Project Management		
Duration	6-12 Months	6-12 Months	6-12 Months

Table 1. Data Science Competencies and Levels

The candidate or his/her mentor, if available, should perform gap analysis to assess the candidate's current competencies and accordingly devise a target competency set that the candidate needs to develop proficiency in through either as a Master Degree program, real-world assignments, industry and academic certifications, or any combination of these developmental programs.

The candidate should thoroughly review this guide and each of the competencies in order to understand what is required to fully master each competency and how it relates to the field of Data Science. The candidate should use the synopsis of each competency to expand and investigate each of the referenced topics in order to fully understand their impact and contribution to this specialty. Candidates should investigate the latest research in each of these areas and ultimately attempt to apply the gained insight toward creating solutions that can optimize and transform the many facets of their business or areas of interest.

3. Data Science – Soft skills & Qualities

This section lists various soft skills and personal qualities that is preferred or need to be acquired by the candidate.

Candidate Soft skills Requirements
Business acumen to solve business problems and creating solutions
Passionate about data, knowledge, and analysis of data (Data intuition)
Demonstrates critical thinking skills

Avid researcher
Aptitude to learn
Possesses excellent analytical and communication skills
Demonstrates creative and innovative thinking
Comfortable working with a wide range of stakeholders and functional teams

Table 2. Candidate Soft Skills and Qualities Requirements

4. Beginner Competencies Estimated Duration (6 Months)

3.1 Computing Foundations

Keywords: *Computer architecture, Processor, Memory, Storage, Network, OS*

Candidate must be familiar with Computer Architecture, Hardware Processors, Memory, Storage, Network protocols, Operating Systems such as Linux and Windows, OS commands, and Shell scripting. Candidate must be able to use computers effectively for their daily work and can develop small scripts to automate various tasks.

3.2 Programming & Tools Foundations

Keywords: *Programming, Languages, Storage, Open Source Software, Notebooks*

Candidate must become familiar with the fundamentals of programming, in case candidate does not have a degree in a computer related field, including programming languages types (e.g. Object Oriented, functional etc.), data structures, programming concepts, design, practices, testing & debugging, and maintaining software programs.

Candidate need to become familiar and proficient in using a variety of Open Source programming languages that are frequently used by Data scientists, such as Python and all of its important libraries and packages such as Numpy, Pandas, Matplotlib, Scipy, and Scikit-Learn, etc.; R and its important packages such as ggplot2, dplyr, reshape2, etc.; Java, C/C++, and Object-Oriented Programming languages in general. Candidate should be able to utilize some of the Open Source Notebooks and IDEs such as Jupyter, IPython, RStudio, VisualStudio, and Spreadsheets such as Microsoft Excel or Google Sheets. Candidate should also be familiar and proficient in using SQL for querying and manipulating structured data repositories.

Candidate should also become familiar and proficient in using and programming in various specific proprietary packages and platforms for Data Analytics such as SAS, MATLAB.

Candidate must be well capable of developing small applications in at least one well-established programming language such as Java, Python, or C/C++. Candidate should

also be able to use Github as a software development and management platform, and one of the largest repositories of Open Source Software.

3.3 Data Foundations

Keywords: *Data wrangling, Data sources*

Candidate should become very familiar with the basics of data management such as Data Curation, Cleaning, Scraping, Transformation, and Data Wrangling. Data is at the center stage of data-driven methods and technologies. Most often data is imperfect, not clean, or missing some fields and need to be curated, cleaned-up, re-formatted or transformed in order to become fit for analytical purposes. Data Wrangling, sometimes also referred to as Data Munging, is a process of transforming data from one raw format to another to make it more pliable for a specific analytical task. Data scientists do usually spend the majority of their time in finding the right data, and performing data related activities such as data cleaning and data organization.

The candidate should also become familiar with the main data sources in his/her organizations (structured, semi-structured, and unstructured), the various types of data types and business processes that consume or generate this data, and the various rules and regulations applied to various data repositories. The candidate should also become familiar with the issues related to data quality, data completeness and correctness of the various data repositories and the various efforts and initiatives to improve the quality of data in these repositories.

3.4 Database & Data Warehouse Foundations

Keywords: *SQL, NoSQL, OLTP, OLAP, Data Warehouse, Oracle, RDBMS*

The candidate needs to become familiar with Relational Database Management Systems (RDBMS) design and concepts such as structured data, table schemas, tables and the relationships between tables, data attributes stored as table columns, data instances stored as table rows, primary and foreign keys, database views and links, and how to create subroutines or models as Stored Procedures. The candidate should also be familiar with the main advantages of relational databases such as data consistency, integrity, security, atomicity, isolation, durability, availability, and performance.
Candidate need to become proficient in using SQL (Structured Query Language) to query and retrieve information from this database. The candidate should be proficient in using the SQL language to construct various queries using Select commands, perform Join to combine data from multiple sources, perform Aggregations and Group By statements, be able to create tables and perform various Insert and Update operations, and create complex subqueries to retrieve the required data.

The candidate should also become familiar with Non-relational or NoSQL databases (Not Only SQL) as alternatives to relational databases and their main differences such as dynamic or undefined schemas for unstructured or document-based data and using key-

value or wide-columnar formats, and areas of applications such as the ability to store and manage large volumes of data that do not have similar structure, and the need for high scalability to handle Big Data.

The candidate should become familiar with the various and relatively newer database design concepts and features, and how they can be applied in various data analytics scenarios or for higher performance and scalability implementations. Concepts and features such as In-memory execution, In-database execution, Column-oriented or Columnar databases, Graph databases, MPP (Massively Parallel Processing) databases, and parallelized ML algorithm implementations.

The candidate should become familiar with Data Warehouses (sometimes referred to as OLAP) and the main difference between them and Operational or Transactional Databases (OLTP), and understand the motives to create an Analytical Database (Warehouse or OLAP) that is separate from the main Operational Database. The candidate needs to become proficient in using the Analytical database for acquiring and analyzing data using all the analytical tools implemented in their environment.

3.5 Statistical Analysis

Keywords: *Statistical tests, Descriptive statistics, Inferential Statistics*

Candidate must develop a solid understanding of various Statistics concepts and techniques and how to apply them, as they are essential for performing initial and exploratory data analysis. Candidate should become familiar with the various statistical tests, data distributions, and when to apply these techniques. Candidate should become very familiar with Descriptive and Inferential statistics. Descriptive statistics are various quantitative measures that describe the various properties of a sample essentially summarizing the sample using properties such as Mean, Median, Mode, Standard Deviation, and Variance. Inferential Statistics is used to infer properties of the larger population by examining just the available data sample, techniques such as Hypothesis testing, A/B testing, P-values, Confidence intervals, significance testing, Z-tests, t-tests, Chi-squared tests, linear and logistic regression analysis. Advanced topics in statistics can include Statistical Modeling, Bayesian statistics, modeling Time series data, etc.

3.6 Probability Theory

Keywords: *Monte Carlo simulations, Central Limit Theorem, Probability distributions*

Probability theory is the mathematical foundation for Statistics and statistical inference in particular. Candidate needs to become familiar with the fundamental concepts in this field such as random variables, dependent and independent events, Monte Carlo simulations, Markov chains, expected values, standard errors, and the Central Limit Theorem. Candidates also need to become familiar with Bayesian methods, continuous and discrete probability distributions, the various probability distributions such as the Normal (Gaussian), Uniform, Bernoulli, Binomial, Exponential, Log-normal distributions and which is the most suitable distribution to apply for different problems (i.e. input data).

3.7 Data Analytics/Exploratory Data Analytics

Keywords: *Exploratory data Analysis (EDA), Data mining, Descriptive*

Candidate must be able to apply Programming, Data Foundations, and Statistics knowledge to explore, clean and analyze the data in order to uncover hidden trends and patterns, gain insights, and derive meaningful interpretations from and of the data. Apply data visualization techniques to expose structures and trends in the data that can lead to eye-opening insights and meaningful information about the data or problem at hand. The candidate should develop a deep interest and passion for data, cultivate their own intuition for data, ask the right questions that are important to answer a specific problem or business scenario, and what methods to apply to obtain answers for the identified questions. In essence, the candidate should transform his/her mindset to apply data-driven methods and techniques to solve and answer business problems and questions. The candidate should practice Exploratory Data Analysis (EDA) and share their findings and visualizations for as many real or business datasets as possible.

Data Analytics can be classified in different categories that can be complementary and used in different stages of solving the same or different problems. **Descriptive** Analytics provides insights into what has happened in the past. **Diagnostic** Analytics provides insights on why it happened. **Predictive** Analytics utilize statistical modeling or deep learning to forecast what can happen in the future based on the past events. **Prescriptive** Analytics goes the extra step of recommending outcomes or actions to take when an event occurs. Exploratory Data Analysis is more often associated with the first two types of data analytics (i.e. Descriptive and Diagnostic), while the last two types are usually referred to as advanced data analytics techniques.

3.8 Data Visualization

Keywords: *Data representation, communication, Plot types*

The candidate must become very familiar with the various tools, techniques and the numerous chart and graph types that should be used depending on the data size, type, and distribution. The candidate need to become proficient with the various data visualization programming tools such as Matplotlib, Seaborn (Python), ggplot (R), and D3.js (JavaScript) and how to use them to present the data or produce the desired results. The candidate should also become familiar with the various commercial visualization applications and products such as Spotfire, Tableau, Microsoft PowerBI, Oracle Data Visualization, etc., if available.

The candidate must also become familiar with the various chart and plot types such as line and bar charts, histograms, scatter plots, bubble charts, and heat maps and when to use them effectively depending on the data shown and the main purpose of the chart such as comparison, relationship, or understanding the data distribution. The candidate

should also become familiar with visual design principles, color theory, and visual encodings to generate simple yet powerful charts that communicate the desired message clearly and succinctly The candidate should practice communicating data analysis findings effectively using data visualization charts and dashboards to stakeholders in the most appropriate way for them to make data-driven decisions Techniques such as Storytelling should be practiced and used as appropriate for the audience and the data being highlighted.

3.9 Project Management

Keywords: *Leadership, Communication, Planning, Critical thinking, Negotiation, SME*

The candidate should possess and demonstrate Project management skills such as Team leadership, Communication, Planning skills, Critical thinking, Change & Scope management, Time management, Task management, Negotiation skills, and Risk management. The candidate should have sufficient knowledge regarding the Subject matter in focus but it is not necessary that the candidate is the Subject Matter Expert (SME).

4. Intermediate Competencies Estimated Duration (6 Months)

4.1 Linear Algebra

Keywords: *Matrix and Vector operations, Optimization methods*

Candidate need to become familiar with linear algebra equations and solving for the missing variables in these equations. Become familiar with Matrix and Vector operations and how they are applied in Machine and Deep Learning to speed up the algorithm logic and calculations. Candidate needs to become familiar with implementing Linear algebra in at least one of the data science programming languages such as Python, R, Matlab, or SAS if available.

4.2 Calculus

Keywords: *Multivariate, derivatives, partial derivatives, integrals, differential equations*

Candidate needs to develop an understanding of multivariate calculus and how it differs from single-variable calculus especially in areas of limits and continuity. Understand partial derivatives and how it applies to solving high dimensional data science problems. Understand Jacobian Matrices that are used to represent and calculate function derivatives and how this is used to create linear transformations of the complex multi-dimensional data. Develop a deep understanding of vector calculus and how it is used to represent and calculate Gradients. Gradient descent is a fundamental mathematical optimization concept that is used heavily in many Machine and Deep Learning algorithms

to calculate the cost function and how to minimize it to solve various machine learning problems ranging from simple Linear regression to complex Deep Learning algorithms. Understand the difference between various types of Gradient descent such as Stochastic, and Batch and when to apply these different types to solve different problems. Become familiar with integral calculus that is used widely in Machine and Deep Learning such as calculating the probability or Expectation of random variables drawn from various probability distributions and how this is used in Reference Learning, Deep-Q-Learning and Generative networks.

4.3 Big Data Foundations

Keywords: *Big data, Hadoop Ecosystem, Open Source Software*

Candidate must become very familiar with the Big Data concept, motivation, software tools, and any available tools in their environment to manage Big Data workloads.

Digital technologies of the Internet, Mobiles, smart gadgets, devices, and sensors have reached critical penetration levels within societies in general and enterprises in specific. The omnipresent adoption of social media networks has caused a massive amount of data that is being constantly generated and consumed and that has acted as the trigger to the Big Data era that has been embraced by all mainstream enterprises. The general characteristics of "Big Data" are the high Volume, Variety, and the high Velocity rate at which data is being generated. One of the major benefits of Big Data is the possibility of combining multiple types of data from multiple sources (structured, semi-structured, and unstructured) to create new insights or innovative solutions that were not possible or imaginable previously when data were only considered individually or within specific sub-domains or silos.

A Hadoop Ecosystem is a platform or framework used to store, manage, transform, and perform analysis on Big Data, which is mostly unstructured data. It contains many components and mostly Open Source Software tools such as HDFS, MapReduce, YARN, Spark, PIG, HIVE, Kafka, HBase, Flume, Sqoop, Ooozie, etc. The storage repository of the Hadoop Ecosystem is usually referred to as a Data lake as opposed to a Database or Data Warehouse. Hadoop Ecosystems are usually complemented with NoSQL databases that are more suitable for storing unstructured data that does not naturally have a fixed tabular or relational format.

As stated, the candidate must become very familiar with this Big Data concept and environment, and most importantly must define a business problem that contains Big Data elements (such as variety of structured and unstructured data) and must develop a solution for this problem using the Big Data environment and software tools.

4.4 Machine Learning

Keywords: *Statistics, Predictive analytics, Supervised, Unsupervised learning*

Candidate must become well versed in Machine Learning as it is considered the bread-and-butter competency within the Data scientists toolbox. Machine learning is the cornerstone of Artificial Intelligence and is generally characterized by using statistics based algorithms that can learn from available data without relying on rule-based programming. Successful ML algorithms, once properly trained using training data sets, can generalize beyond the training data to correctly interpret or predict results when presented with new or never--seen before data. This is generally referred to as Predictive Analytics.

Candidate should be very proficient in the two main ML algorithm learning categories: Supervised and Unsupervised Learning, and the various ML algorithms in each category, such as Linear and Logistic Regression, Naive Bayes, Decision Trees, Random Forests, Support Vector Machines, Ensemble methods (Supervised), and Clustering (K-means, Hierarchical), Principal Component Analysis, Singular Value Decomposition (Unsupervised), and Independent Component Analysis.

Candidate must know the advantages and disadvantages of all the common ML algorithms, and needs to develop intuition as to when to apply these algorithms in different problem domains depending on data size, distribution, linearity, and the desired classifier output category. This competency must be well mastered and must be demonstrated by solving multiple real business problems in order to develop the required skills and proficiency in this area.

The candidate should also become familiar with the latest research in various areas of ML and how these technologies can be deployed to solve various business problems. For example, Adversarial Machine Learning is a relatively new focus area under Machine Learning that is used within the computer security field to provide better cybersecurity solutions and protection against various vulnerabilities that have dynamic and changing patterns.

4.5 **Deep Learning I**

Keywords: *Neurons, Perceptrons, Artificial Neural Networks, Convolutional*

Candidate must become well versed in Deep learning (DL) as a subfield of Machine learning (ML) that has distinct design background and capabilities than traditional machine learning and that allows data scientists to solve more complex problems. DL is inspired by the perceived structure and functionality of the human brain. The neurons of the brain are simulated using a network of multiple and connected layers containing perceptrons, which are mathematical functions mimicking the neurons. These layered structures are referred to as Artificial Neural Networks (ANN).

The candidate must understand the fundamental difference between DL and ML and when to apply DL as opposed to ML. DL is currently the hottest area of AI, and is extensively and successfully applied to solve Image & Pattern recognition, Speech recognition, Natural Language Processing (NLP), and Autonomous robotics. DL requires

and performs better when very large training datasets are available, and it can generally produce higher accuracy as more data becomes available, yet it is computationally expensive and usually requires specialized hardware resources such as Graphics Processing Units (GPU), especially during the model training phase.

The candidate must become familiar with the various types of Neural Networks and when to apply them for different scenarios, such as Multilayer Perceptrons (MLP), Convolutional Neural Networks (CNN), Transfer Learning (TL), Generative Adversarial Networks (GAN), Recurrent Neural Networks (RNN), Long Short-term Memory (LSTM), etc.
The candidate should also become familiar with the various DL Frameworks and Middleware such as Tensorflow, Keras, Pytorch Caffe2, MXNet, CNTK, and the hardware specific libraries such as Nvidia CUDA, and CuDNN, etc.

4.6 Data Engineering

Keywords: Data modeling, *Data preparations, Data transformation*

Although Data Engineering can be viewed as a separate discipline or specialty, data scientists must have a sufficient level of understanding of the concepts, techniques, and tools employed in Data Engineering as they may need to perform some of these tasks as part of their data science workflow, or in order to communicate their requirements to a Data engineer to curate and avail the required data in the required format in order to successfully complete the required project.

There are various skills that need to be mastered by a Data Engineer. These include designing, building, maintaining, and optimizing databases and data warehouses, data modeling and designing schemas, data query optimization, data Ingestion, integration & transformation, data filtering & aggregation, creating data pipelines and ETL (Extract Transform, Load) jobs for various analytical use cases, maintaining data security and governance requirements, and finally deploying and maintaining developed predictive models and solutions in production. One of the main skills of a Data Engineer that need to be mastered by Data scientists as well is the transformation of raw data into a form that can be easily queried to generate insights, or consumed as historical training data to generate predictions for a defined business case.

There are different data repositories and environments that Data engineers need to master such as Relational Databases, Analytical warehouses, Big Data environments that are sometimes referred to as a Hadoop Ecosystem, or environments that specifically handle Real-time data for streaming data analytics. Although the overarching goal is to generate value out of this data via data-driven techniques, these environments have their own architectures, products, and tools that might require specific skills. An example is the need to be proficient in SQL when dealing with relational databases versus the need to be proficient in Python, R, and NoSQL databases when dealing with unstructured data that reside in a Hadoop environment. Although, recently traditional relational databases started to provide support for languages such as Python and R but their main underlying interface is still predominantly SQL.

This Data engineering competency is closely related to competencies 3.3 and 3.4 in the Basic competencies section but the candidate needs to expand these competencies by fully understanding the skillset needed and used by Data engineers.

4.7 Cloud computing

Keywords: *On-demand computing, Elastic, IaaS, PaaS, SaaS, Private, Hybrid, Public*

Cloud Computing, as defined by the National Institute of Standards and Technology (NIST) is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

The main characteristics of cloud computing are on-demand, self-service, rapid elasticity, pay-per-use, resource pooling, and multi-device support. Cloud computing's main attraction is enabling the consumption of computing resources or an application as a utility just like electricity when compared to building and maintaining computing infrastructures on-premise, which provides cost savings due to the economies of scale and the reduced data center and energy expenditures.

The three main cloud service models are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).

IaaS model provides the highest degree of control to the consumer by obtaining access to raw compute, storage, and network resources on which the consumer needs to load and manage their own Operating Systems and applications.
PaaS model provides a platform on which consumers can build and run their own applications. This model is often used for application development and deployment, such as Data Analytics and Data science workloads.
SaaS model provides a complete software stack of application and platform that is completely supported by the cloud provider. Examples are Google Drive and Microsoft Office 365.

There are three cloud deployment models namely: **Private**, **Hybrid**, and **Public** cloud deployments. Private clouds are usually single-tenant but the infrastructure can be on or off-premise and either self-managed or managed by a third party. Public clouds are owned by a cloud service provider and are multiple-tenant with a pay-as-you-go OPEX model. Hybrid clouds is a combination of the other two types, and usually act as a bridge between private and public cloud resources.

The candidate needs to become familiar with this cloud operating model, especially for Data Analytics, Data Science and AI workloads as they are often run on the cloud, and that are now more commonly referred to as Machine Learning as a Service (MLaaS).

Candidate needs to become familiar and practice running Data science workloads using the top cloud providers in this area that provide a complete platform or APIs for specific managed services. Examples of cloud platform and service providers include Amazon Web Services (e.g. Amazon Machine Learning, SageMaker, DeepLens, Lex, Polly, Rekognition, Transcribe, etc.), Microsoft Azure (e.g. Azure Machine Learning, Machine Learning Studio, Azure Databricks, Azure Cognitive Services, Cognitive Search, etc.), Google Cloud Platform (Google Prediction API, ML Engine, Datalab, Dialogflow, Cloud Vision API, Cloud Natural Language API, Cloud Speech API, etc.), IBM Cloud (e.g. Watson Studio, Watson Knowledge Catalog, Watson Discovery, Watson Visual Recognition, Watson Machine Learning, Deep Learning, Data Refinery, etc.), and Oracle Cloud (e.g. Oracle Autonomous Transaction Processing, Autonomous Data Warehouse).
The candidate should be fully aware of the pros and cons of running workloads in the cloud vs. on-premise and the various security and jurisdiction concerns of running in the cloud.

5. Advanced Competencies — Estimated Duration (6 Months)

5.1 DevOps Practices

Keywords: *Continuous Integration, Continuous Delivery, Microservices, Containers, Orchestration, Agile, Infrastructure as Code, Software Defined Infrastructure*

DevOps is essentially a method to eliminate the traditional siloed approach between software Development and Operations teams and align them in order to deliver software and services using agile methodology to the customers. DevOps is a collection of practices and technologies including Continuous Integration/Continuous Delivery (CI/CD), Microservices, Infrastructure as Code (IaC), Containerization, Repository Managers, Configuration Management, and Monitoring.

Candidates need to become fully aware and proficient in using the DevOps practices and technologies, as they may adopt it in order to develop, deploy, and maintain their developed solutions such as predictive models or AI applications to their customers in an agile fashion.

One of the main advantages of DevOps is the ability to provide rapid feature delivery to the customers more frequently, with higher reliability, better quality, and on-demand elasticity. DevOps also adopts the Microservices application architecture, where applications are decomposed into major components or capabilities and utilized by the customers as APIs (Application Programming Interface) via HTTP. The rapid update frequency of DevOps requires the automation of the entire development and operations pipeline into what is called Continuous Integration and Continuous Delivery practices and tools. Infrastructure resources also need to be provisioned on-demand as part of the

DevOps process in order to properly deliver the software capabilities and customer expansion requirements using software defined Infrastructure or what is called Infrastructure as Code.

Container technology is also widely used as part of the DevOps practice and represents the next level in infrastructure virtualization enabling software applications to be pre-packaged with all the required runtime dependencies such as files, and libraries for efficient and safe deployment to the customers. Containers are lightweight as they do not contain the operating system and as such multiple containers can be spun on either bare metal or virtualized servers. Containers allow applications and hardware resources to be provisioned and dismantled almost instantly and automatically with elastic scalability and application isolation. Orchestration may also be needed for the management and automation of containers. There are various container implementations such as Docker, and Singularity, while Kubernetes or Red Hat OpenShift are common orchestration tools.

As mentioned, DevOps integrates, streamlines, and automates the entire software development and infrastructure operations and provisioning processes using software tools. These DevOps tools provide a full technology stack that covers the entire development and operations lifecycle. The candidate needs to become aware and well capable of using these tools that are deployed as part of their organizations DevOps environment in order to develop and maintain AI solutions for their customers. Some examples of these tools are CI/CD tools (Jenkins, Travis CI, Git, Microsoft Azure DevOps Server, Nexus or JFrog responsory and artifact managers), Infrastructure as Code (Iac) and configuration management tools (Ansible, Chef, Puppet). DevOps also incorporates tools for Monitoring and logging in order to be provide the DevOps teams with a comprehensive visibility and transparency of the entire process in order to address any issues across the entire development and operations lifecycle accordingly (monitoring tools include Splunk, Grafana, Kibana)

5.2 Reinforcement Learning

Keywords: *Intelligent agents, Action-Reward, Deep Q-networks*

Candidate need to become familiar with Reinforcement Learning (RL) theory programming techniques, and possible areas of application. RL is a relatively recent subfield of ML based on the concept of Action and Reward based learning, where a software agent learns how to interact with any given environment and eventually achieves maximum performance by adjusting its behavior according to the feedback it receives from the environment. RL can be applied to create intelligent agents that can master various tasks without being specifically programmed for the task or environment specifics such as playing all kinds of games, creating intelligent robotics that can navigate their environments and complete a given task, self-driving cars, Finance and trading, etc.

The candidate needs to become familiar with the various RL techniques used to define the environment and how the agent interacts with it, such as Markov Decision Processes

(MDP), Dynamic Programming, Monte Carlo Methods, Temporal-Difference Methods, Q-Learning, Value-based, Policy-based, and Actor-Critic Methods, etc.

Reinforcement Learning combined with Deep Learning neural networks is referred to as Deep Reinforcement Learning. Deep RL can be combined with Q-Learning (referred to as Deep Q-Network - DQN), or Policy gradients (such as Deep Deterministic Policy Gradient - DDPG) to solve complex problems either in real or continuous space. This will be expanded on in the next section, Deep Learning II (5.3).

5.3 Deep Learning II

Keywords: *Convolutional Neural Networks, Deep Q-networks, Frameworks*

In the Deep Learning I competency, the candidate hopefully learned the fundamentals of DL and practiced by creating and training basic Neural Network architectures that solve basic image or pattern recognition prediction problems. The candidate should also have become familiar with the various DL frameworks and the hardware that is required for Neural Networks (NN) training and inference.

In this advanced level of this competency, the candidate is required to further and deepen their knowledge about NNs in various ways:
Neural networks, and especially CNNs, require careful design and optimization of the network architecture in terms of the number of hidden layers and their sizes, filter sizes, pooling layers, dropout layers, batch normalization, and the activation functions used in the network. This becomes evident when examining the various CNN architectures that were designed to compete in the ImageNet challenge such as AlexNet, Inception or GoogleNet, VGGNet (VGG-16, VGG-19), ResNet, etc. These Neural networks can consist of a high number or deep layers and millions of parameters (e.g. ResNet has 152 layers and 60 million parameters) but can surpass human level accuracy in image recognition. The candidate should become familiar with the various designs of these networks and the general principles of the design concepts and their tradeoffs between complexity and accuracy. The candidate should learn to design their own deep neural networks that are capable of solving relevant business problems if they exist, and should also be able to utilize the aforementioned pre-trained NNs to solve related business problems using the Transfer Learning method, assuming there are similar general features between the initial training set used by these models with the targets. The candidate should also practice working with these complex NNs and tuning them to generate better results for different tasks.

The candidate should also investigate utilizing NNs to solve real-world or business problems other than image or speech recognition, such as solving business problems involving time-series (real-time) data, spatial data, sequential data, Natural Language Processing related problems, and Recommendation systems.

Candidate should experiment with Deep Q-Networks (DQN), or Deep Reinforcement Learning in general (introduced in section 5.2). DQN combines DL with RL to

approximate the action-value function and has been demonstrated by Google DeepMind to be able to learn to play various video games better than humans. DQN is still in early stages, can be tricky to implement, and may not yet generalize to a wider set of problems but it holds the promise of creating AI (Artificial Intelligence) solutions that can learn to solve complex problems without being explicitly programmed for these problems. It is probably the closest technique on the path to AGI (Artificial General Intelligence) and the candidate should become familiar with it. If the candidate can define a problem, where DQN might provide a solution then candidate should pursue it as the outcome can be of substantial benefit. This topic can also be pursued for further research by graduate degree candidates if they are enrolled in a graduate degree.

Candidate should also become acquainted with Generative Adversarial Networks (GAN) and their possible areas of business applications. GANs are constructed using two Neural Networks that are adversarial or are contesting with each other. The generative network produces synthetic data while the discriminative network tries to determine if the generated data is fake or real. This contest allows each network to optimize their NN such that the system eventually generates synthetic data that is indistinguishable from the real data. GANs holds promise for applications in various domains such as Geological modeling or Simulation. Candidates are encouraged to explore innovative business applications and attempt creating possible solutions using this technology. This area can also be pursued for research by graduate degree candidates.

Additionally, candidates need to become proficient with using multiple DL frameworks, and should be familiar with the advantages of using one over the other depending on the business task and objectives such as language used, performance, scalability, deployment, simplicity, Type of NN required, and granularity control of the network layers. This is a deeper dive into the DL frameworks beyond the basic awareness and familiarity with a specific framework that was required in section 4.5.

5.4 Parallel Computer Architecture & Programming

Keywords: Computer architecture, *Parallelism, SIMD, MIMD, Hardware accelerators*

Candidates need to understand parallel computer architecture and programming and how to leverage these capabilities in order to improve the performance of model training and inference and scale these models especially when consuming large datasets or large-scale deployment is required.

The candidate should become familiar with the general parallel processing theory and architectures and the distinction between SIMD (Single Instruction, Multiple Data) and MIMD Multiple Instruction, Multiple Data) architectures. Currently most computer processors such as CPUs and GPUs are classified as SIMD, which describes their capability to perform the same operation on multiple data points at the same time. Vector processing is a variant of SIMD where specialized hardware components process a one-dimensional array of input numerical data using pipelining to achieve parallelism, and is used to be a separate processor category implemented via specialized processors but it

is now mostly implemented within the CPU using a software library of vector instruction sets (such as Intel MMX). In MIMD, the processors act independently by performing different operations on different data points simultaneously. Examples of MIMD are Intel Xeon Phi and the Connection Machine (e.g. CM5) by Thinking Machine. There are also shared-memory and distributed memory architectures where either all processors share a single memory pool (UMA - Uniform Memory Access), or the latter where each processor has its own local memory (NUMA - Non-Uniform Memory Access).

Parallel computers can implement parallelism at various hardware and software levels. Parallelism is implemented in hardware using a combination of multi-processors, multi-cores, many-cores, clusters, and grids of connected clusters. There are also Massive Parallel Processing architectures that distribute tasks using a large number of independent specialized processing units and interconnect networks, and this concept is adopted and deployed in certain Database appliances and in some Big Data technologies.

Hardware accelerators are used to speed up the training and sometimes the inference time for Deep Learning neural networks. Graphical Processor Units (GPU) are usually used for this purpose. There are other accelerators such as Field-Programmable Gate Arrays (FPGA), Application-Specific Integrated Circuits (ASIC) that can provide higher training efficiency in certain cases. Deep Learning can also utilize the multi-core and multi-threads of CPUs, but it would run much slower on CPUs due to the lower number of cores or effectively the degree of parallelism that is available in the CPU compared to the other accelerators. GPUs such as NVIDIA's Tesla V100 GPU, and ASICs such as Intel Nervana NNP (Neural Network Processor) and Google TPU (Tensor Processing Unit) deserve special attention and focus as possible hardware processors for accelerating and scaling Deep Learning training workloads. Neuromorphic and Quantum Computing represent new paradigms that can be used for designing Artificial intelligence specialized applications in the future.

Candidates should also become familiar with the various Interconnection Networks that are deployed in various systems either for high throughput and low latency communication between processors and nodes, or to transfer data from storage systems. Examples of Interconnect networks are Ethernet, InfiniBand, and OmniPath in addition to other proprietary interconnect networks.

The candidate should also become familiar with the processing system memory hierarchy, which include processor registers, cache, main memory (RAM), storage, and the newer class of Non-volatile memory or storage that is based on Flash or other technologies such as Resistive, Phase-change, stacked, or High Bandwidth Memory (HBM), and how they can be used to accelerate the performance of big data workloads.

Candidate needs to also be fluent with the various software frameworks, libraries, and parallel compilers that are used to parallelize and scale their code either on specific accelerators or on general CPUs. For example, CuDNN (NVIDIA CUDA Deep Neural Network) is provided by NVIDIA to create GPU-accelerated library of primitives for executing Deep Neural Networks on their GPUs. There are some Python modules that provide concurrency capability such as Threading, and Multiprocessing. There are also various libraries that provide parallel execution of Python code on CPUs (multi-processor,

multi-core) such as Parallel Python (PP), PyMP, VecPy, etc. There are Python libraries that provide concurrency using the NVIDIA GPUs such as PyCUDA. There are also development efforts to eventually converge AI and HPC workloads on various hardware processors and accelerators using abstraction software APIs such as Intel Nervana Graph. Solutions for High Performance AI and scaling Deep Learning (training) workloads on multi-GPUs, especially across multi-nodes are available but continue to be matured by the vendors. As an example, NVIDIA has created NCCL (NVIDIA Collective Communication Library) for multi-GPU scalability, which uses MPI-like directives. There are also efforts to scale DL frameworks using MPI (Message Passing Interface), which is a mature large-scale parallelization library used successfully within the HPC community, such as Caffe-MPI. Candidate should become familiar and up to date with the latest tools and methods that are available to provide scalable and parallel DL solutions to efficiently address the scalability challenges as required by the environment he/she focuses on.

5.5 Applied Mathematics

Keywords: *Mathematical optimizations, Gradient descent*

Become very familiar with the mathematical notations and formulas for various Machine Learning algorithms such as Linear Regression (in terms of the Hypothesis and Cost functions), and the Gradient Descent formulas. Mathematical optimization is one of the main enablers for performing machine learning on large amounts of data. Most of the time off-the shelf optimization algorithms are available in many of the open source libraries and packages such as Gradient descent and that can be applied directly to solve most problems in Machine Learning. However, in certain cases where the data is very large and the data patterns are complex, designing a mathematically optimized algorithm or formula can have a major improvement on the efficiency and scalability of the final AI solution. Becoming well versed in the various optimization techniques and methods and their formulas is desirable for advanced and expert data scientists. Although, creating novel mathematical optimization formulas may fall in the realm of advanced and research mathematicians (usually PhD theses), but keeping pace with research in this area and being able to adopt these novel optimization algorithms in solving Machine Learning problems more efficiently is the mark of advanced and top-notch data scientists.

6. Program Additional Resources

6.1 Data Science Courses and Certifications

Refer to the "Data Science Complementary Guide" Excel sheet; section "Courses & Certifications"

This is just a recommended list of Data Science courses and certifications, but is by no means final or conclusive. Data science is a very dynamic area, and this list should be reviewed and updated annually as feedback is received from the participants, and as new courses and certifications are either modified or new ones become available.

MOOCs are found to be an excellent learning tool for a variety of topics including Data Science. MOOCs create and continuously improve their contents, the courses are developed by some of the best industry leaders in the topics covered, and the hands-on or capstone projects are well-selected to make sure students are capable of applying and practicing what they have learned using real cases and real data. The MOOCs that focus on project-based learning are usually the best learning sources. I encourage the candidates that make use of this guide to continuously send feedback on the most useful courses so it can be ever-greened.

Not all these courses and certifications are required in each level. Some of them overlap, while others are vendor-specific. Directors or mentors, if available, should work closely with the candidates to determine which of the courses/certifications should be pursued by the candidate to complement his/her existing knowledge and skills, and the platforms and tools that should focused on depending on their perspective environments.

The courses/certifications categorization for each level is not strict. Some courses/certifications can be taken at different levels as some of them cover competencies that overlap different levels. Mentors or candidates have the flexibility to schedule some of these courses/certifications at different levels than what is suggested in the list.

6.2 Master of Science in Data Science Programs

Refer to the "Data Science Complementary Guide" Excel sheet; section "Master of Science Degree"

This section provides a list of suggested Universities for acquiring a Master of Science in Data Science. Again, this is just a recommended list of university programs, but is not is conclusive. These recommendations should be updated as feedback is received from the candidates.

6.3 Data Science Recommended Conferences

Refer to the "Data Science Complementary Guide" Excel sheet; section "Conferences"

This is a list of recommended Data science related conferences, but is not conclusive. List should be updated with the candidates' feedback as they attend and evaluate related conferences.

6.4 Societies and Research

Refer to the "Data Science Complementary Guide" Excel sheet; section "Societies & Research". These are just recommendations, and mentors/candidates can research the for the best society or research portal that is more focused on their special area of interest.

6.5 Open Data Resources

Refer to the "Data Science Complementary Guide" Excel sheet; section "Open Data Resources". These are just recommendations, and mentors/candidates can search for other data sources and portals that might provide better data to meet their specific area of interest.

Candidate can search for open data sources through the relatively new Google Dataset Search Portal (https://datasetsearch.research.google.com)

6.6 Data Science Recommended Readings

Refer to the "Data Science Complementary Guide" Excel sheet; section "Recommended Readings"

This contains a list of recommended articles, tutorials, and books. It should be updated regularly, as this is a dynamic field.

Acronyms and Glossary

AGI	Artificial General Intelligence
AI	Artificial Intelligence
ANN	Artificial Neural Network
ASIC	Application-Specific Integrated Circuits
BA	Bachelor of Arts
BS	Bachelor of Science
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CuDNN	CUDA Deep Neural Network (by NVIDIA)
DDPG	Deep Deterministic Policy Gradient
DL	Deep Learning
DQN	Deep Q-Network
EDA	Exploratory Data Analysis
ETL	Extract Transform, Load
FPGA	Field-Programmable Gate Arrays
GAN	Generative Adversarial Network
GPU	Graphics Processing Units
HBM	High Bandwidth Memory
HPC	High Performance Computing
IaaS	Infrastructure as a Service
LSTM	Long Short-term Memory
MDP	Markov Decision Processes
MIMD	Multiple Instruction, Multiple Data
MLaaS	Machine Learning as a Service
ML	Machine Learning
MLP	Multilayer Perceptron
MOOC	Massive Open Online Courses
MPI	Message Passing Interface
MPP	Massively Parallel Processing
MA	Master of Arts
MS	Master of Science
NCCL	Collective Communication Library (by NVIDIA)
NIST	National Institute of Standards and Technology
NLP	Natural Language Processing
NN	Neural Network
NNP	Neural Network Processor
NoSQL	Not only SQL
NUMA	Non-Uniform Memory Access
OLAP	Online Analytical Processing
OLTP	Online Transaction Processing
PaaS	Platform as a Service
PhD	Doctor of Philosophy

RAM	Random Access Memory
RDBMS	Relational Database Management Systems
RL	Reinforcement Learning
RNN	Recurrent Neural Network
SME	Subject Matter Expert
SaaS	Software as a Service
SIMD	Single Instruction, Multiple Data
SQL	Structured Query Language
TL	Transfer Learning
TPU	Tensor Processing Unit
UMA	Uniform Memory Access