Computing Competencies for Undergraduate Data Science Curricula

Draft 2

December 2019

ACM Data Science Task Force

We welcome your feedback. Please submit comments at
https://forms.gle/Xp8oiPywCo4h3tB79
by March 31, 2020.
Please also watch for a Call-For-Exemplars (Data Science courses and majors).

A letter from the Co-chairs of the ACM Education Board

Dear Readers

In 2009, Turing award winner Jim Gray spoke of data science as a fourth paradigm of science (empirical, theoretical, computational and data-driven) arising from and capitalizing on the huge amount of data that is now available for investigation. The confluence of the availability of data and increasing sophisticated tools, processes, and algorithms for analyzing and drawing knowledge and insight from data has impacted every area of scientific engagement. It has also opened up exciting new opportunities for interdisciplinary work across the many fields including (but certainly not limited to) computer science, mathematics, statistics, and information science from which it draws foundational knowledge.

For computer science, the emergence of data science offers both tremendous opportunity and something of a conundrum, as once again the emergence of a new and closely related computing practice or field raises inevitable questions about whether and how it fits into current post-secondary computer science curricula.

This document represents an effort by the ACM Education Board through the work of the Data Science Task Force to answer this question. It is an effort to put our own data science house in order. This document is not, however, an effort to claim ownership or even primacy in data science. To do so would be to negate the powerful interdisciplinarity that data science makes possible.

It is our hope that this document will represent a productive step in a conversation that engages all relevant fields and disciplines. Toward this end, the ACM Education Board wishes to express our willingness and excitement about participating in future, more expansive and inclusive conversations regarding the promise and practice of data science.

Chris Stephenson
Jane Prey
Co-chairs, ACM Education Board
ACM

ACM Data Science Task Force

Andrea Danyluk, Co-chair, Williams College, USA

Paul Leidig, Co-chair, Grand Valley State University, USA

Scott Buck, Intel Corporation, USA

Lillian Cassel, Villanova University, USA

Maureen Doyle, Northern Kentucky University, USA

Keegan Hines, Capital One, USA

Tin Kam Ho, IBM, USA

Andrew McGettrick, University of Strathclyde, UK

Suzanne McIntosh, New York University, USA

Jian Pei, Simon Fraser University, Canada

Weining Qian, East China Normal University, China

Karl Schmitt, Valparaiso University, USA

Christian Servin, El Paso Community College, USA

Hongzhi Wang, Harbin Institute of Technology, China

CONTENTS

Chapter 1: Introduction

1.1 Charter

At the August 2017 ACM Education Council meeting, a task force was formed to explore a process to add to the *broad, interdisciplinary conversation* on data science, with an articulation of the role of *computing discipline-specific contributions* to this emerging field. Specifically, the task force would seek to define what the computing/computational contributions are to this new field, and provide guidance on computing-specific competencies in data science for departments offering such programs of study at the undergraduate level.

There are many stakeholders in the discussion of data science – these include colleges and universities that (hope to) offer data science programs, employers who hope to hire a workforce with knowledge and experience in data science, as well as individuals and professional societies representing the fields of computing, statistics, machine learning, computational biology, computational social sciences, digital humanities, and others. There is a shared desire to form a broad interdisciplinary definition of data science and to develop curriculum guidance for degree programs in data science.

This volume builds upon the important work of other groups who have published guidelines for data science education. There is a need to acknowledge the definition and description of the individual contributions to this interdisciplinary field. For instance, those interested in the business context for these concepts generally use the term “analytics”; in some cases, the abbreviation DSA appears, meaning Data Science and Analytics.

This volume is the second draft articulation of computing-focused competencies for data science. It recognizes the inherent interdisciplinarity of data science and situates computing-specific competencies within the broader interdisciplinary space.

1.2 Motivating the study of Data Science

Those who study Data Science have to develop a mind set with a strong focus on data – the collection of data and, through analysing it appropriately, using this to bring about beneficial insights and changes. For instance:

- Obtaining data about the quality of air in a city can result in removing dangerous pollution or sending warning messages to those who suffer from asthma.

- Collecting data about traffic in real time can result in steps being taken to avoid traffic congestion.

- Collecting patient data can lead to new insights for disease diagnosis and treatment.

- Recording data about speech in a certain area can assist with speech recognition.

The possibilities are endless, and the contributions that Data Science can make to transforming businesses, transforming society and basically shaping the future for the better are huge. The possibilities also carry with them potentially negative consequences.

Students of Data Science need to be imbued with the 'joy of data,' seeing data as the 'currency or fuel of our time'. They also need to be imbued with a strong sense of professional and ethical responsibility. Data Science courses ought to reflect such sentiments; likewise the education of data scientists.

The topic of careers is of course important from a marketing perspective. Suffice it to say that the current demand is considerable and growing daily.

1.3 Committee work and processes

The Data Science Task Force was initiated at a meeting of the ACM Education Council in August 2017. The Co-chairs were appointed at the meeting and were charged with developing a charter for the work, as well as assembling a task force with global representation.

The Co-chairs drafted a proposal to create the Task Force, which was approved by the ACM Education Board in January 2018. The initial Task Force – approximately half of the members of the current committee – convened for a full-day meeting in February 2018.

In preparation for a second face-to-face meeting in July 2018, the Task Force designed two surveys to gather input from academia and industry on the computing competencies most central to Data Science. The results of the survey are presented in this report, with details provided in Appendix B. During this time, the Co-chairs invited additional members to join the committee and began to develop a global advisory group.

At the July 2018 meeting, the ACM Task Force developed the set of computing-focused Knowledge Areas for Data Science that appeared in the first public draft of this project (available at http://dstf.acm.org/DSReportInitialFull.pdf). With the release of the first draft report, the ACM Data Science Task Force called for discussion and feedback from all data science constituencies. They presented the report and gathered comments at conferences and meetings, including Educational Advances in Artificial Intelligence (EAAI-9), held at AAAI in January 2019; the SIGCSE Symposium in February 2019; the Conference Board of the Mathematical Sciences in May 2019; and the Joint Statistical Meetings in July 2019.

Since the release of the first draft, the ACM Data Science Task Force has held two additional face-to-face meetings at SIGCSE 2019 and in August 2019. Based on feedback from the community, they (with the help and feedback of subcommittees) revised the list of Knowledge Areas as well as specified competencies for those in significantly greater detail than previously.

With the release of this second draft report, the ACM Data Science Task Force is calling for discussion and feedback from all data science constituencies. The Task Force will be presenting the report and gathering comments at conferences and meetings such as the SIGCSE Symposium in March 2020. The Task Force will also be putting out a **Call For Examplar (CFE) Courses and Major Curricula**. Finally, the Task Force also welcomes feedback by email to the Co-Chairs:

- Andrea Danyluk (andrea@cs.williams.edu)
- Paul Leidig (leidigp@gvsu.edu)

For the feedback link, CFE, and updates on this project, go to http://dstf.acm.org.

1.4 Overview of this Report

This report updates and builds on Draft 1 of this project. Many sections from the initial draft have been modified or replaced. Others sections in Draft 2 are entirely new.

Significant Modifications include the following:

- The Competency Framework is the foundation on which these curricular recommendations have been developed. This draft re-describes the competency framework as it is actually implemented here.
- The list of Knowledge Areas has been significantly modified, and the competencies for each are specified in much greater detail.

Additions include:

- Tier 1 (T1), Tier 2 (T2), and Elective (E) attributions for competencies.
- A section explaining how the reader should interpret the above attributions, especially when designing a Data Science undergraduate curriculum.
- Chapters on broadening participation, characteristics of data science graduates, and challenges for colleges and universities developing Data Science majors.

1.5 Acknowledgments

This draft report has benefited from the hard work, insights, feedback, and support of many individuals and organizations. We thank all of them for their contributions: Matt Bishop (UC Davis), Yu Cai (Michigan Technological University), Mine Cetinkaya-Rundel (Duke University), Li Chen (Intel Corporation), Jessen Havill (Denison University), John Impagliazzo (Hofstra University), Ha-Kyung (Hidy) Kong (Seattle University), Andres Mendez-Vazquez (Cinvestav Guadalajara, Mexico), Marion Neumann (Washington University in St. Louis), Alan Peterfreund (SageFox Consulting Group), Rajendra Raj (RIT), Carol Romanowski (RIT), Deepak Tosh (University of Texas at El Paso), Yi-Chieh (Jessica) Wu (Harvey Mudd College), the ACM Education Advisory Committee, the ACM Education Board, the Conference Board of the Mathematical Sciences, as well as participants at EAAI, AAAI, SIGCSE, and the Joint Statistical Meetings in 2019.

Chapter 2:
Current View of Data Science and Prior Work

2.1 Interdisciplinarity in Data Science

Data Science is an inherently interdisciplinary field. The rise of Data Science is directly connected to the rise of large data sets across nearly every topic domain. The sciences, social sciences, business, humanities, and engineering all are seeing opportunities for discovery and decision-making expanded by unprecedented amounts of raw or structured data. The data is too large to allow effective human analysis without the automation of processes. Data Science is the field that brings together domain data, computer science, and the statistical tools for interrogating the data and extracting useful information.

Data Science requires effective integration of a domain to provide data and a context for its exploration, statistics, and computer science. Domain experts understand their data and perhaps know what they can expect to learn from the data. They want tools and techniques to get the job done. They need to know enough about the tools and techniques to be confident that the results will be reliable.

Statisticians bring expertise in "data analysis, data collection, modeling, and inference." [Park City 2017, p 7]. Statistical models "describe, predict, and explain processes" [Park City 2017, p 8]. The PCMI Data Science Guidelines provide a full description of the statistics and mathematical foundations needed for data science.

Computer scientists bring methods for dependable storage, for protecting privacy, and the integrity of the data. They bring expertise in applying high performance computing and networked systems for efficient computation. Algorithms for machine learning and deep learning techniques allow results that go beyond direct analysis of the existing data and offer opportunities for discovery that may not be anticipated by the data owners. Computer Science offers tools for the analysis of data of all types, whether numeric, text, image, sound, or complex combinations of basic data types.

Each component of the Data Science environment: the **domain** that provides the data; **statistics** for analysis, modeling, and inference; and **computer science** for data access, management, protection, as well as effective processing in modern computer architectures, is essential. However, a random collection of the three elements does not constitute a meaningful Data Science program. Data Science is interdisciplinary and requires the effective integration of the three components to produce meaningful results.

True interdisciplinary work is challenging. If each component remains independent, the relationships remain blurred and the opportunities for cross-fertilization are reduced. Some see interdisciplinary efforts as a reduction in one or more of the component areas. When dealing with a truly interdisciplinary subject, the goal must be to see the new whole that is composed of contributions from each part. It is not possible to include everything from every part, but

that is not the point. The point is to define something new that takes important parts from each contributor.

Early programs in Data Science will often work with a group of existing courses from the participating disciplines. That is practical and easy to bring a new program into existence. The difficulty in that scenario is to make the essential connections so that all the parts work together to support discovery and decision making in the domain. Cross references between courses, projects that call upon topics learned in other courses, and a comprehensive project to bring all the pieces together are essential to turn a mixed set of courses into a cohesive, interdisciplinary program.

2.2 Prior work on defining data science curricula

As an inherently interdisciplinary area, data science generates interest within many fields. Accordingly, there have been a number of Data Science curriculum efforts, each reflecting the perspective of the organization that created it.

This project looks at data science from the perspective of the computing disciplines, but recognizes that other views contribute to the full picture. The following examples are especially important, and have informed the committee's work.

The EDISON Data Science Framework (2018)

EDISON is a project started in September 2015 "with the purpose of accelerating the creation of the Data Science profession." The core EDISON consortium consists of seven partners across Europe. Since 2015, the group has worked to create the EDISON Data Science Framework. This collection of documents includes a general introduction, as well as four detailed components, including:

- Data Science Competences Framework
- Data Science Body of Knowledge
- Data Science Model Curriculum
- Data Science Professional Framework

This comprehensive set of curricular volumes parallels the intended structure of our work. EDISON was in earlier stages as this project began; at present, it is clear that there are significant overlaps, and future versions of our work with reconcile our model with the EDISON curriculum, with the intention of creating a complementary volume, rather than a replicated or competing volume.

The National Academies of Science, Engineering, and Medicine Report on Data Science for Undergraduates (2018)

As the press release announcing the publication of the National Academies report states, "Data science draws on skills and concepts from a wide array of disciplines that may not always overlap, making it a truly interdisciplinary field. Students in many fields need to learn about data collection, storage, integration, analysis, inference, communication, and ethics." The report highlights the demand for data scientists and calls for a broad education for students across programs of study. Identifying many data science roles, including those related to hardware and software platforms, data storage and access, statistical modelling and machine learning, and business analytics, among others, the report does not presume that every data scientist will be expert in all areas, but rather that programs will develop to allow graduates to fulfil specific roles.

The intent of the National Academies report was to highlight the importance, breadth, and depth of data science, and to provide high-level guidance for data science programs. It is not a detailed curricular volume in the sense of the EDISON project or this ACM Data Science effort.

The Park City Report (2017)

The Park City Math Institute 2016 Summer Undergraduate Faculty Program convened with the purpose of articulating guidelines for undergraduate programs in data science. The three-week workshop brought together 25 faculty from computer science, statistics and mathematics. The base assertion of the report and proposed curriculum is that data is the core: "The recursive data cycle of obtaining, wrangling, curating, managing and processing data, exploring data, defining questions, performing analyses, and communicating the results lies at the core of the data science experience."

The resulting list of key competencies shows the interdisciplinary nature of data science, with an understandable focus on the mathematics and statistics:

- Computational and statistical thinking
- Mathematical foundations
- Model building and assessment
- Algorithms and software foundation
- Data curation
- Knowledge transference – communication and responsibility

The role of computer science appears in the description of computational thinking: "Data science graduates should be proficient in many of the foundational software skills and the associated algorithmic, computational problem solving of the discipline of computer science." However, further description relates these skills to understanding the programming and algorithms behind "professional statistical analysis software tools."

The Park City report deserves further description. It includes an outline of the Data Science Major:

1. Introduction to data science
    a. Introduction to Data Science I
    b. Introduction to Data Science II
2. Mathematical foundations
    a. Mathematics for Data Science I
    b. Mathematics for Data Science II
3. Computational thinking
    a. Algorithms and Software Foundations
    b. Data Curation—Databases and Data Management
4. Statistical thinking
    a. Introduction to Statistical Models
    b. Statistical and Machine Learning
5. Course in an outside discipline

The report also includes a description of each of the courses. For the purposes of this report, it is noted that programming is introduced in Introduction to Data Science I and II, and appears again

as a part of Algorithms and Software Foundations. The course in Data Curation includes traditional databases as well as newer approaches to data storage and interaction. The course in Statistical and Machine Learning "blends the algorithmic perspective of machine learning in computer science and the predictive perspective of statistical thinking."

Although there certainly are additional aspects of computer science that are relevant to the preparation of a student of data science, there is clearly an effort to combine the mathematical and computer science contributions to produce a blended program. This ACM Data Science report builds on the Park City work with a heavy orientation toward computer science. The position of the Task Force is that any Data Science program will have to reflect competencies in mathematics, statistics, and computer science, possibly with different emphases. This is consistent with the view of the National Academies report. Graduates of programs following the Park City guidelines will have valuable strengths and graduates of programs following these ACM guidelines will have different, but equally valuable strengths.

The Business Higher Education Framework (BHEF) Data Science and Analytics (DSA) Competency Map (2016)

The work provides a four-level competency map. The base, or Tier 1, level describes personal effectiveness competencies. These are not considered competencies learned in school, but rather part of an individual's personal development. Examples include integrity, initiative, dependability, adaptability, professionalism, teamwork, interpersonal communication, and respect.

Tier 2 describes academic competencies to be acquired in higher education. These are most relevant to this report and include the following:

- Deriving value from data
- Data literacy
- Data Governance and Ethics
- Technology
- Programming and Data Management
- Analytic Planning
- Analytics
- Communication

Tier 3 presents workplace competencies: planning and organizing, problem solving, decision-making, business fundamentals, customer focus, and working with tools and technology.

Tier 4 is for Industry-Wide Technical Competencies. These are not specified, but represent skills that are common across sectors of a larger industry context.

Though Tier 2 includes a competency in "Programming and Data Management," the description mentions only "Write data analysis code using modern statistical software (e.g., R, Python, and SAS)." This set of competencies does not address a need for developing new software or systems in support of data science, but relies on available tools.

Business Analytics Curriculum for Undergraduate Majors (2015)

This report was produced in 2015 by the Institute for Operations Research and the Management Sciences (INFORMS). Reflecting the focus of programs in Business, this INFORMS curriculum assumes basic computer literacy as a starting point. It suggests revising some of the standard courses in statistics to meet newer needs. The resulting course list includes: Data Management, Descriptive Analytics, Data Visualization, Predictive Analytics, Prescriptive Analytics, Data Mining, and Analytics Practicum. It also includes electives.

Like the guidelines from the Business Higher Education Framework, the focus is on doing something with data, primarily to serve business needs. There is no mention of programming. The data management course includes SQL, but has no prerequisites. The emphasis in the data mining course is on framing a business problem. Data mining techniques are compared, and large datasets are to be used. The tools to be used for that purpose are not specified.

Initial workshops related to this ACM Data Science Curriculum effort (2015)

In October 2015, the National Science Foundation sponsored a workshop with representatives of many perspectives on data science. Some attendees represented established programs, others represented societies with an interest in data science. The final report, "Strengthening Data Science Education Through Collaboration," describes the discussions and reflects the diversity of opinions. Although opinions varied, there were some areas of agreement. Those form the basis of the list of Knowledge Areas in this current ACM report.

Summary

The review of existing curricular efforts suggests that it would be important to capture in a single volume the contributions that computing makes to data science. Through developments such as the Internet of Things, sophisticated sensors, face recognition and voice recognition, automation, etc., computing opens up many avenues for data collection. It can play a vital role as a custodian of information with great attention being paid to maintenance but crucially also to security and confidentiality matters. Then the analysis of large amounts of information and utilization of that for the purposes of machine learning or augmented intelligence in its various roles can bring significant benefit.

2.3 Survey of academic and industry representatives

In order to gain an understanding of the current data science landscape, the ACM Data Science Task Force conducted a survey of ACM members, representing academic institutions and industry organizations. Through outreach to ACM members, the Task Force was also able to reach computing professionals outside of ACM membership. In all cases, the Task Force sought global participation. There were 672 responses to the academic survey and 297 responses to the industry survey.

Academic Survey

The academic survey asked academics whether their institution had any sort of data science program at the undergraduate level, asked what type of program was offered, in what department(s) it was housed, and what computing areas were required, elective, or not present in the program. It also allowed respondents to add to the list of computing areas specified in the survey. Finally, the survey asked participants whether their data science program had a "data science in context" requirement – i.e., a requirement that students apply data science to another area.

Nearly half of respondents from academic institutions (47%) reported they did not offer an undergraduate data science program. However, over half of those who reported offering some type of program offered a full bachelor's degree in data science.

Nearly all of the programs offering a bachelor's degree in data science required courses in programming skills and statistics. In addition, the majority of programs also required data management principles, probability, data structures and algorithms, data visualisation, data mining, and machine learning. Other courses included topics such as ethics, calculus, discrete mathematics and linear algebra. We note that a majority of programs also required a "data science in context" course.

Administratively, the largest percentage of programs were housed in a computer science department; however, almost as many were in an "other" category. This result might be somewhat skewed, given that the survey was fielded primarily with ACM members.

Additionally, over half of these programs reported graduating 10 students or less annually.

We expect that the number of Data Science programs will increase, as will the number of students choosing to study it. This, then, is an ideal time to articulate computing-based competencies for those programs.

Industry Survey

The industry survey roughly mirrored the academic survey; however, the primary question was whether a company looked for job applicants with data science experience and what computing experience they required or preferred those applicants to have.

In the survey of industry representatives, nearly half (48%) responded that they look for candidates specifically with data science or analytics degrees or educational backgrounds. We found it particularly interesting that the majority of employers reported these employees work as individual contributors on data science tasks.

Industry respondents reported requiring experience or skills in similar areas to those required by college or university Data Science programs. One slight difference is that employers reported requiring more computing skills than statistical or mathematical skills.

Other Observations

The ACM Task Force was somewhat surprised by certain survey results. For instance, industry respondents did not report data security and privacy as a required competency area for job applicants. We note that this may reflect employers' understanding of what Data Science (and Computer Science) programs are requiring of their majors. That is, it might reflect the reality of the applicant pool, rather than a "wish list" of competencies.

Similarly, we note that academic institutions reported what they currently require, rather than what they would require in an ideal world. This might, in some cases, reflect the availability of courses and faculty at an institution, rather than a "gold standard" for Data Science programs.

A more detailed summary of survey results is presented in Appendix B.

References

[ACM 2103] *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science (ACM/IEEE 2013):* *https://www.acm.org/education/CS2013-final-report.pdf*

[ASA 2014] *Curriculum Guidelines for Undergraduate Programs in Statistical Science (ASA 2014b): http://www.amstat. org/education/pdfs/guidelines2014-11-15.pdf>*

[BHEF 2106] *Data Science and Analytics (DSA) Competency Map, Business The Business Higher Education Framework (BHEF) version 1.0 produced in* November 2016

[Cassel 2011] Interdisciplinary computing is the answer: now, what was the question?, by Lillian N. Cassel, ACM Inroads 2, 1 (Feb 2011), 4-6. DOI=http://dx.doi.org/10.1145/1929887.1929888

[CasselTopi 2015] *Strengthening Data Science through Collaboration,* by Lillian Cassel and Heikki Topi, Technical Report and report of 2015 NSF Workshop. http://www.computingportal.org/sites/default/files/Data%20Science%20Education%20Workshop%20Report%20.0_0.pdf

[CUPM 2015] *Curriculum Guide to Majors in the Mathematical Sciences (MAA 2015).* See http://www.maa.org/ sites/default/files/pdf/CUPM/pdf/CUPMguide_print.pdf

[EDISON] *The Edison Data Science Competence Framework* http://edison-project.eu/edison/edison-data-science-framework-edsf.

[Edison 2015] Data science professional uncovered: How the Edison project will contribute to a widely accepted profile for data scientists, by Manieri, A.; Brewer, S.; Riestra, R.; Demchenko, Y.; Hemmje, M.; Wiktorski, T.; Ferrari, T.; and Frey, J. published in IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom), 588–593. National Academies of Sciences, Engineering, and Medicine. 2018.

[INF 2015] *Business Analytics Curriculum for Undergraduate Majors*, Coleen R. Wilder, Ceyhun O. Ozgur (2015) published in INFORMS Transactions on Education 15(2):180-187. https://doi.org/10.1287/ited.2014.0134

[NatAc 2018] *Data Science for Undergraduates: Opportunities and Options*, published by the National Academies of Sciences, Engineering, and Medicine, 2018. Washington, DC: The National Academies Press. https://doi.org/10.17226/25104

[Park City 2017] *Curricular Guidelines for Undergraduate Programs in Data Science* by DeVeaux, R.; Agarwal, M.; Averett, M.; Baumer, B.; Bray, A.; Bressoud, T.; Bryant, L.; Cheng, L.; Francis, A.; Gould, R.; Kim, A.; Kretchmar, M.; Lu, Q.; Moskol, A.; Nolan, D.; Pelayo, R.; Raleigh, S.; Sethi, R.; Sondjaja, M.; Tiruviluamala, N.; Uhlig, P.; Washington, T.; Wesley, C.; White, D.; and Ye, P. 2017. Annual Review of Statistics and Its Application 4:15–30.

Chapter 3:
Introduction to the Body of Knowledge

3.1 Knowledge Areas

Following the work of previous ACM curricular volumes (see [1], for instance), this report is organized around Knowledge Areas (KAs). The KAs identified for the first draft of this report (released in January 2019) were based on survey input (see Section 2.3) as well as prior work, with special attention being given to the results of the workshop reported in [2].

Feedback on the first draft prompted the Task Force to revise the list of KAs. The core computing discipline-specific Knowledge Areas for Data Science described in this (second) draft are (in alphabetical order):

- Analysis and Presentation (AP)
- Artificial Intelligence (AI)
- Big Data Systems (BDS)
- Computing and Computer Fundamentals (CCF)
- Data Acquisition, Management, and Governance (DG)
- Data Mining (DM)
- Data Privacy, Security, Integrity, and Analysis for Security (DP)
- Machine Learning (ML)
- Professionalism (PR)
- Programming, Data Structures, and Algorithms (PDA)
- Software Development and Maintenance (SDM)

For a full curriculum the above need to be augmented with competencies in calculus, discrete structures, probability theory, elementary statistics, advanced topics in statistics, and linear algebra, among others. A complete curriculum would also include at least one domain context for application of Data Science concepts and methods.

3.2 The Competency Framework

The Competency Framework provides a framework for the description of the various Knowledge Areas.

In deciding on a competency framework, the Data Science Task Force looked to other ACM curricular volumes for guidance. In particular the 2013 ACM Computer Science volume ([1]) and Chapter 4 on Competencies in the IT 2017 ([3]) volume received significant attention. The notion of Knowledge Areas was present in both as was the notion of competence. The IT 2017 volume had introduced the notion of *disposition*, a concept that had its origins in [4], as well as the concept of *context* to highlight distinctive situations; both of these featured in the thinking of the Task Force.

In the final analysis, it was decided to adopt a competency framework that best suited the Task Force who had been charged with providing curricular guidance in the new area of Data Science. There was a wish to have some latitude in approach so as to be able to convey their thinking and permit an appropriate focus on the discipline and its implementation.

The Framework

In broad outline, the framework takes the form of a set of Knowledge Areas, each Knowledge Area representing a significant and coherent body of material to be included in Data Science degree programs. Knowledge areas provide a vocabulary of terms that the community will use and focus on in discussion about Data Science. They capture concepts that are significant and central to the discipline. The definition of these should be as self-contained as possible to reflect their central role.

Each Knowledge Area description follows a template. (See Figure 3-1.) To amplify:

- Following the name of the Knowledge Area, there will be a relatively brief paragraph describing the area and its relevance in Data Science
- A section addressing the scope of the Knowledge Area as it applies to Data Science
- A section capturing very high-level competencies, skills as well as dispositions
- A list of sub-domains, and finally
- The definitions of the listed sub domains

The Knowledge Areas themselves are further refined into a set of sub-domains, each of these being defined by associated competencies. See Figure 3-2 at the end of this chapter for a list of Knowledge Areas and the accompanying sub-domains.

Sub-domains

In describing and defining the sub-domains, the initial text is intended to draw attention to the relevance of the topic within Data Science, bearing in mind that the overall report is to capture the study of Data Science at the undergraduate level. This is followed by sections addressing knowledge, skills, dispositions and context.

The knowledge section takes the form of a bulleted list of major topics or topic areas within the sub-domain.

The skills section is a bulleted list of skills to be acquired during study of the sub-domain. Naturally this tends to reference the knowledge items within that sub-domain.

Skills are expressed in terms of learning outcomes (LOs). Formally, a learning outcome captures an aspect of what a student is expected to accomplish through study of particular topics.

Knowledge Area Name

Text giving a brief description of the knowledge area and its role in Data Science.

Scope	Competencies
• High level description of the scope of this knowledge area, stressing its relevance to Data Science • The description should be in the form of a relatively small number of bullet points	• To be kept at a very high level. More detail is provided with the sub-domains. • Provide bullet items that capture knowledge, skills, and dispositions
Sub-domains	
List sub-domains here	List additional sub-domains here

The various sub-domains are then addressed in order. For each sub-domain, the sub-domain name is followed by an optional brief paragraph describing the sub-domain and then

- A list of knowledge topics
- A list of skills
- An optional list of dispositions, and finally
- An optional context section

Likewise for the other sub-domains.

Figure 3-1 Template for Knowledge area definitions

LOs have their origins in Bloom's taxonomy. This was originally developed in 1956 and has been the subject of much study and development over the years. See, for example, [5] which is a significant development and forms the basis of the approach used here; throughout, the Task Force has placed an emphasis on capturing aspects of the discipline of Data Science rather than always ensuring strict adherence to the framework.

Basically a set of cognitive processes is identified, namely to *remember, understand, apply, analyze, evaluate and create.* These cognitive processes are seen to be of increasing cognitive challenge, though the precise challenge owes much to the context (as defined by the set of topics to which it applies) as well as the background of a student. Students can demonstrate a particular cognitive skill in various ways. Table 3-1 – which owes much to [5] - gives a list of verbs that illustrate how achievement of a particular cognitive process can be demonstrated; within the list there are some illustrations of use. The list of verbs should be regarded as indicative and not exhaustive.

The various learning outcomes then vary in terms of the level of challenge they represent. Moreover, the number of curriculum hours involved in achieving an LO can also vary. For a degree program in higher education, it would be expected that the higher-level cognitive processes feature in some of the LOs.

Additional Features of sub-domains

On the T1, T2, E designations

The three designations T1, T2 and E have been used and are associated with the various knowledge, skills, and dispositions. In brief

- T1 (Tier 1) denotes an item that all Data Science graduates should have mastered
- T2 (Tier 2) denotes an item that most Data Science graduates would be expected to have mastered. Any given Data Science graduate would be expected to have mastered a majority of T2 items.
- E (Elective) signifies an item that, although important, could reasonably be regarded as forming part of an elective

These designations may appear at different levels of granularity

- When placed at the level of a sub-domain, a designation applies to all items within that sub-domain
- Otherwise, the designation applies at the item level

The Data Science Task Force recognizes that it is unreasonable to expect all T2 topics to be accommodated within the one program. This issue is further developed in Chapter 4 of this document, which addresses (at a high level) building a complete major curriculum. These three designations are intended to offer guidance on the possible selection of topics for a newly designed Data Science program, facilitating, for instance, placing an emphasis on the statistical elements or the cognitive elements that support machine learning and/or artificial intelligence.

Cognitive Processes and indicative competency verbs

Remember – retrieve relevant knowledge from memory

Recognize	identify
Recall	retrieve

Understand – construct meaning from instructional messages

Interpret	clarify, paraphrase, represent, translate
Exemplify	illustrate, instantiate
Summarise	generalise
Infer	conclude, extrapolate, interpolate, predict
Compare	contrast, map, match
Explain	

Apply – carry out or use a procedure in a given situation

Execute	carry out (e.g. programs, demonstrations)
Implement	use (e.g. tools, techniques), solve (e.g. equations)

Analyse – break material into its constituent parts and determine how the parts relate to one another and to an overall situation

Differentiate	discriminate, distinguish, focus, select (e.g. from a set of techniques, systems, methods), make informed choices
Organize	find, make coherent, integrate, outline, parse, structure
Attribute	decompose (e.g. hierarchical decomposition)

Evaluate – make judgements based on criteria and standards

Check	coordinate, detect, monitor, test (e.g. programs, data sets for bias, completeness, etc)
Critique	judge, critically review

Create – put elements together to form a coherent or functional whole; reorganise elements into a new pattern or structure

Generate	hypothesize (e.g. attributes of data sets), derive (e.g. equations, formulae), construct models (e.g. of software systems), abstract (e.g. in software), classify (e.g. data)
Plan	design (e.g. programs, systems, questionnaires, research activities, implementation of new initiatives)
Produce	construct (e.g. systems, proofs, technical guidance, technical papers, presentations to senior management), derive (e.g. equations, formulae, information from data sets)

Table 3-1 Cognitive Processes and Associated Competence Verbs

---

On dispositions

The concept of *disposition* arose in [4]. To quote (subject to formatting):

> *The dispositions and pre-dispositions category arose from an attempt to capture the "areas of values, motivations, feelings, stereotypes and attitudes" applicable to computational thinking.*

These included:

- *Confidence in dealing with complexity*
- *Persistence in working with difficult problems*
- *The ability to handle ambiguity*
- *The ability to deal with open-ended problems*
- *Setting aside differences to work with others to achieve a common goal or solution, and*
- *Knowing one's strengths and weaknesses when working with others.*

This concept was further amplified in [3] where it states that

> *Dispositions encompass socio-emotional skills, behaviours, and attitudes that characterise the inclination to carry out tasks and the sensitivity to know when and how to engage in these tasks […]. To distinguish dispositions from knowledge and skills, [..], a disposition "concerns not what abilities people have, but how people are disposed to use these abilities."*

On the context sections

An optional context section is used to draw attention to aspects of sub-domains that may vary depending on the environment, such as the geographical location (where laws and culture may vary).

The Body of Knowledge

The complete definition of the Data Science Body of Knowledge (computing oriented) appears in Appendix A of this volume.

Analysis and Presentation
- Foundational considerations
- Visualization
- User-centered design
- Interaction design
- Interface design and development

Artificial Intelligence
- General
- Knowledge representation and reasoning – logic based
- Knowledge representation and reasoning – probability based
- Planning and search strategies

Big Data Systems
- Problems of scale
- Big data computing architectures
- Parallel computing frameworks
- Distributed data storage
- Parallel programming
- Techniques for Big Data applications
- Cloud computing
- Complexity theory
- Software support for Big Data applications

Computing and Computer Fundamentals
- Basic computer architecture
- Storage systems fundamentals
- Operating system basics
- File systems
- Networks
- The web and web programming
- Compilers and interpreters

Data Acquisition, Management, and Governance
- Data acquisition
- Information extraction
- Working with various types of data
- Data integration
- Data reduction and compression
- Data transformation
- Data cleaning
- Data privacy and security

Data Mining
- Proximity measurement
- Data preparation
- Information extraction
- Cluster analysis
- Classification and regression
- Pattern mining
- Outlier detection
- Time series data
- Mining web data
- Information retrieval

Data Privacy, Security, Integrity, and Analysis for Security
- Data privacy
- Data security
- Data integrity
- Analysis for security

Machine learning
- General
- Supervised learning
- Unsupervised learning
- Mixed methods
- Deep learning

Professionalism
- Continuing professional development
- Communication
- Teamwork
- Economic considerations
- Privacy and confidentiality
- Ethical considerations
- Legal considerations
- Intellectual property
- On automation

Programming, data structures and algorithms
- Algorithmic thinking and problem solving
- Programming
- Data structures
- Algorithms
- Basic complexity analysis
- Numerical computing

Software development and maintenance
- Software design and development
- Software testing

Figure 3-2 The (Computing) Data Science Knowledge Areas (with sub-domains)

References

[1] Computer Science Curricula 2013, Curriculum Guidelines for Undergraduate Degree Programs in Computer Science, produced by the Joint Task Force on Computing Curricula formed by the Association for Computing Machinery and the IEEE Computer Society, published by ACM on 20th December 2013

[2] *Strengthening Data Science through Collaboration,* by Lillian Cassel and Heikki Topi, Technical Report and report of 2015 NSF Workshop. http://www.computingportal.org/sites/default/files/Data%20Science%20Education%20Workshop%20Report%20.0_0.pdf

[3] Information Technology 2017, Final Curriculum Report IT2017, published by ACM on 10th December 2017

[4] Barr, V. and Stephenson, C. Bringing computational thinking to K-12: What is involved and what is the role of computer science education community? ACM Inroads, 2, 1 (May 2011), 48-54.

[5] Lorin W. Anderson and David R. Krathwohl (editors) A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives, published by Addison Wesley Longman, New York and London 2001

Chapter 4:
Building a Program from Curricular Recommendations

Borrowing heavily from the Computer Science 2013 volume:

> *As useful as the Body of Knowledge is, it is important to complement it with a thoughtful understanding of cross-cutting themes in a curriculum, the "big ideas" of [Data Science]. In designing a curriculum, it is also valuable to identify curriculum-wide objectives, for which the Principles and the Characteristics of Graduates chapter of this volume should prove useful.*

These observations are highly relevant in the context of this present report. How are they to be interpreted? Data scientists have to behave ethically, seeing opportunity and benefit in employing the methods and techniques of the discipline to analyze and derive new information from data. The methods and techniques have their origins in the basic disciplines of computer science, mathematics, statistics and machine learning reflecting the interdisciplinary nature of the discipline which, in given situations, may be complemented by application domain knowledge. Ethics, practical skills, recognizing the potential of data, assembling high quality data, and seeking new knowledge from that data to bring benefit must underpin the philosophy and ethos of a program in data science.

In designing a Data Science program these matters need to be captured in a high level description of the expected outcomes of the program, or more formally in a specification phrased in terms of the expected learning outcomes from study of the program. Since Data Science is an emerging discipline, it is to be expected that the curriculum will be revised regularly. Thought needs to be given to the implications of that. But there are other considerations that also need to be taken into account.

Program design considerations

Aspects of the nature of degree programs are often determined by national or by institution wide considerations. Below two particular cases are highlighted. But study programs throughout the world - for instance, in Australia / New Zealand, Canada, China and South America, tend to be a variant on these, though in some countries, e.g. Australia or South Africa, it is often possible to register for a program that provides two qualifications.

Higher Education in the US

A rich range of educational possibilities exists with large engineering schools, large research-oriented schools, as well as liberal arts colleges, community colleges, and many others. Typically, Bachelor's degree programs are 4 years in duration. In many institutions students have the chance to explore options especially in the early years. In the later years they have the

possibility to touch on topics that border on research. As well as these majors, the system also accommodates liberal arts degrees as well as minors, which can be taken in conjunction with study of another discipline.

Possibilities for study abroad may exist and typically occur during the third year. Efforts have to be made to match the curricula in the home and abroad institutions so that students genuinely benefit and are not disadvantaged when they return to their home institution.

Another common feature is to allow students to undertake internships in industry. These may be of 6 months duration but may also last for the full year (and this may extend the duration of the degree program). These need to be carefully planned. The experience of the students should contribute significantly to the student's knowledge and understanding of Data Science and the institution should ensure that proper quality checks are in place as a safeguard. Variants of this do exist, e.g. in the UK in the form of Graduate Apprenticeships, where there is a deeper bond with industry that involves a partnership throughout the entire study program.

The European Scene

The European Higher Education Area (EHEA) is "a group of 48 countries that cooperate to achieve comparable and compatible higher education systems throughout Europe." This is to facilitate student as well as staff mobility between EHEA countries with a view to facilitating employment. The system is underpinned by the Bologna process. This is characterized by three cycles: the first is a Bachelor's degree cycle lasting 3 years; the second is a Masters cycle lasting a further 2 years; and the third is for Doctorate degrees and lasts an additional 3 years. The Bologna system includes a qualifications framework, a system of credits and quality assurance. The European Credit Transfer System (ECTS) stipulates, for instance, that 90 credits is the typical number of credits per year.

Despite the Bologna framework, there are differences in achievement across the various countries brought about by considerations such as the differing outputs from the school systems.

On detailed curricula

In terms of detailed curricula, a wide variety of possibilities exists, even beyond the considerations mentioned above about program structures. For instance, a program can reflect different emphases on computing, statistics, machine learning, mathematics, or a particular discipline of application outside of those.

Data Science is a new discipline, not always well understood by students nor by their parents / advisors. It is desirable that, in the early stages (e.g. the first semester), an initial course - with a title such as *An Introduction to Data Science* – should be provided to enthuse students and even attract additional students to the discipline. A significant premium should be placed on motivation and conveying the philosophy as well as the real achievements of the discipline; the course might be designed so that it is available to a wide cross section of the student community.

It is natural to expect that classes in the early stages of a degree program reflect the basis of

computer science, statistics, mathematics and machine learning. However, certainly in the later phases, and even relatively early in the program, all the classes ought to have a distinctive Data Science flavor to reflect and promote the character of this emerging discipline. In the later phases of the program, possibilities ought to exist allowing students to explore different options and possibilities, perhaps driven by career ambitions or research leanings.

Most material in the compulsory classes should be T1 / T2 material, though the possibility should exist for particular expertise in an institution infusing the curriculum. For optional / elective classes in the later years, the material should be T2 / E. A major project in the final year (that demonstrates a student's ability to solve a significant Data Science problem by pulling together material from several classes) should fall into the E category.

As a final important comment, generally Data Science programs should exist in an environment where there is encouragement as well as incentives and rewards for exceptional performance.

Data Science in context

In addition to developing foundational skills in computing and statistics, data science students should also learn to apply those skills to real applications. It is important for data science education to incorporate real data used in an appropriate context.

Data Science curricula should include courses designed to promote dual coverage combining both data science fundamentals and applications, exploring why people turn to data to explain contextual phenomena. Such courses highlight how valuable context is in data analytics; where data are viewed with narratives, and questions often arise about ethics and bias. It can be beneficial to teach some courses with a disciplinary context so that students appreciate that data science is not an abstract set of approaches. Related application disciplines might include physics, biology, chemistry, the humanities, or other areas.

Exemplar courses and programs

Although there are a great variety of possible course possibilities, the above comments have been generic in nature. The next (final) draft of this document will include exemplar courses and programs that institutions may adopt or adapt for their local settings.

References

Infosheet – Bologna Process Overview, produced by the National Commission for Further and Higher Education, from the project 'Supporting the Bologna Process', Malta, 2014-2016.

The Bologna Process Implementation Report for the Yerevan Ministerial Summit in 2015 is available online at: http://www.ehea.info/news-details.aspx?ArticleId=385

Chapter 5: Broadening Participation

5.1 Overview

Data Science (DS) is a new field with roots in Mathematics, Statistics, and Computer Science (CS) and applications to multiple fields. The fact that it is an emerging discipline provides an opportunity for diversity and inclusion; we should ensure from its inception that there is broad representation of students who are welcome and persist to graduation and beyond. The fields most closely aligned to DS are also fields with relatively unbalanced representation of various groups, including, for example, women and racial minorities, among others. Data Science can learn from these degree programs' successes and failures.

Consider the following examples of disparity in fields frequently associated with Data Science. In the U.S. women represent 57% [1] and underrepresented minorities (URMs)[1] represent 25% [2] of all Bachelor's recipients in the U.S., but only 19.5% and 12.6% of CS graduates, respectively [3]. According to the National Science Board *Science and Engineering Indicators 2018* [4], the gender disparity is true for both 2-year and 4-year degree programs in the U.S. According to EuroStat [4], a website of the European Union, over 1.3 million people were enrolled in the field of Information and Communication Technologies (ICT) in the European Union (EU) in 2016. Females were largely in a minority, accounting for only one in six ICT students (16.7%). The data for gender representation are somewhat better for Australia, although they still have lower than 35% female enrollment in CS. And Australia, like many nations, is grappling with disparities among other groups. Samaras [6] notes that "as social, economic, and political opportunity becomes increasingly wedded to ICT access in the information society, Indigenous digital disadvantage threatens to perpetuate or exacerbate the existing inequalities constraining access.

This report is not the first to identify the emergence of Data Science as an opportunity to reverse the problem of imbalance observed in the fields from which DS has grown. Excellent work and recommendations for ensuring broad participation in Data Science exist. As stated in the National Academies of Sciences, Engineering, and Medicine (NAS) report, *Envisioning the data science discipline: the undergraduate perspective* [7], the Data Science community has an opportunity to build DS curriculum to be welcoming and inclusive from the start. As such, Data Science, as a discipline, should take the best practices for broadening participation from Mathematics, Statistics, and Computer Science as well as successes from countries that have more balanced representation. One of four key recommendations in the NAS report is proactive and intentional collaboration on Data Science curricula and broadening participation between two and four year institutions. Another key recommendation is to require assessment metrics for inclusion in the overall assessment of Data Science programs.

The AAAS report *Levers for Change: An assessment of progress on changing STEM instruction* [8] discusses problems common to both Mathematics and Computer Science including the lack

[1] URMs in CS include all U.S. minorities except Asian Americans who are well represented.

of support for using evidence-based teaching techniques, dearth of information on inclusion of Lesbian, Gay, and Transgendered students, and continued underrepresentation of Hispanics, African Americans and students from lower socio-economic homes. AAAS recommends support for faculty to use evidence-based instruction, providing more role models, acknowledging and confronting implicit bias, and reducing stereotype threat classes.

The South Big Data Hub's *Keeping Data Science Broad: Negotiating the Digital & Data Divide* report [9] includes ten asks for the community, including the following that directly impact curriculum: Foster partnerships between different institutional types; Provide flexible pathways into data science education; Time & space to discuss collaboration; Hiring female faculty, faculty of color, and female faculty of color; and Provide examples of curriculum for 2-year college degrees, certificates or pathways.

The ACM Data Science Task Force is focusing on undergraduate curriculum, however, it is equally important for data science industry professionals and graduate programs to embrace the goal of broadening participation by ensuring fair and inclusive work environments.

5.2 Benefits of Broadening Participation

A major argument for broadening participation in computing has been the shortage of computer scientists, whereby recruiting and retaining women and other underrepresented groups would help fill worker shortages. Similarly, shortages in the data science workforce are anticipated. According to [10], based on a survey of the data analysis department of China's commercial commission, the gap between the need for and supply of basic data analysis talent exceeds 14,000,000. According to a report by the European Commission in 2017 [15] the EU was forecasted to face a data skills gap corresponding to 769,000 unfilled positions by 2020. And [11] reports that the data scientist role will become increasingly important in all industries. Therefore, the argument for filling worker shortages applies here, however the value of ensuring inclusion of women and other underrepresented groups goes beyond having a ready and ample workforce.

As studies have repeatedly demonstrated, diverse teams are smarter; they focus more on facts, process facts more carefully, and are more innovative [12]. The National Council for Women & Information Technology (NCWIT)'s "*What is the Impact of Gender Diversity on Technology Business Performance: Research Summary*" [13] summarizes research studies and identifies key findings of benefits and costs of diverse teams. Diversity is a prudent financial decision. Gender-balanced companies perform better financially, particularly when women occupy a significant proportion of top management positions. Furthermore, gender-balanced teams improve team productivity. Teams that are more diverse adhere to project schedules in various technology companies, share knowledge and have lower project costs than homogeneous teams.

Intentional inclusion and diversity are necessary to reduce societal bias as data science continues to be used for decision making from health care to hiring decisions. News articles are published regularly (e.g., 43 articles in Google News with the terms "bias Artificial Intelligence" on 12/13/2019) either discussing or highlighting bias in machine learning algorithms. Propagation of societal biases should be anathema to Data Science. We do not claim that only

underrepresented individuals can grapple with these issues, but as the data presented above demonstrate, diverse teams with members of many different backgrounds are smarter.

Data Science should be open to all. Data Science jobs, like computing and statistics jobs, pay high salaries – according to [14], the average data scientist earns $121,189 in the U.S. – and the field should be open to all, independent of class, race, gender, sexual orientation, gender identity, ethnicity and other factors that do not influence one's ability to succeed in the field. If not, we are faced with an issue of social equity.

5.3 Recommendations

Data Science programs should report student and faculty demographics as part of assessment.

The NAS report recommends that assessment of diversity be part of all Data Science programs. This is paramount to ensuring Data Science does not repeat inequities that exist in other STEM programs. Data Science programs should monitor and report on enrollment and graduation rates by gender, race, national origin, and socio-economic status.

Assessment reporting should include demographics for faculty as well, and include recruitment activities in support of diverse faculty. The composition of Data Science faculty must be considered as part of holistic approach to broadening participation. The value of a diverse team is argued above, and this also applies to faculty. In addition, diversity and inclusion is improved for student populations when students experience diverse faculty and learn in an inclusive setting. Ponjuan [16], for example, reports on the positive impacts on Latino students when Latino faculty are present. Therefore, Data Science programs should incorporate best practices in hiring to ensure a diverse group of faculty. This requires intentional outreach to underrepresented faculty at research conferences and conferences for underrepresented groups. NCWIT offers multiple resources on how to construct the job advertisement and interview techniques for welcoming new faculty.[2] Once hired, it is even more important to provide a welcoming and supportive work environment for all faculty. NCWIT has a number of references for recruiting and retaining females in technology, developing male allies, and having strong mentoring programs.

Data Science course content should be designed to support a diverse student body. Faculty, trained in inclusive and diverse teaching pedagogy, should implement these methods to support all recruited students.

"Inclusive pedagogy at its core is learner-centered and equity-focused, creating an overarching learning environment in which students feel equally invited and included. Drawing from a large body of research—much of it foundational scholarship on teaching and learning—it is clear that learning outcomes are improved for everyone when teachers attend to student differences

[2] Such as "7 tips for Conducting Inclusive Faculty Searches," available at https://www.ncwit.org/resources/ncwit-tips-7-tips-conducting-inclusive-faculty-searches/ncwit-tips-7-tips-conducting

and take deliberate steps to ensure that all students, across differences […] feel welcomed, valued, challenged, and supported in their academic work." [17] Data Science faculty – indeed, all faculty, should learn about inclusive pedagogy and put such techniques into practice.

Many resources provide guidance on inclusive pedagogy [18]. The National Center for Women & Information Technology (ncwit.org) provides resources for recruiting and retaining women, and The American Association for Advancement of Science *Levers for Change: An Assessment on Changing STEM Instruction* also provides recommendations. Additionally, the ACM Special Interest Group on Computer Science Education (SIGCSE) has published significant work to improve diversity and inclusion, as have publications from Research on Equity & Sustained Participation in Engineering, Computing, & Technology (RESPECT).

Promotion and review criteria should incorporate evaluation of evidence-based inclusive teaching practices.

Providing professional development is not enough to ensure that effective teaching practices are maintained [8]. Therefore, it is important to devise promotion and review criteria that reward faculty for sustained implementation of inclusive teaching.

Data Science programs should include sustained funding for faculty development in teaching.

So that faculty may learn and continue to develop best practices in teaching, Data Science programs should provide the funding necessary for this.

Intentional collaboration between 2-year and 4-year post-secondary colleges and universities.

Community colleges in the U.S. enroll, on average, a diverse student body. This creates an opportunity, then, to recruit diverse students to Data Science. Efforts should be taken to create 2+2 pathways for Community College students who wish to pursue an A.S. followed by a B.S. in Data Science. As noted by Lyon and Denner [19], Community College faculty should be partners on 4-year advisory boards to ensure that students transferring from community college have a smooth, and clear, pathway to a bachelor's degree.

Similar collaboration should be initiated at analogous institutions, where they exist, across the globe.

References

[1] U.S. Department of Education National Center for Education Statistics. 2018. *Postsecondary Institutions and Cost of Attendance in 2017-18; Degrees and Other Awards Conferred, 2016-17; and 12-Month Enrollment, 2016-17*. Technical Report. Washington, DC.

[2] U.S. Department of Education National Center for Education Statistics. 2019. Status and Trends in the Education of Racial and Ethnic Groups. On the Internet at https://nces.ed.gov/programs/raceindicators/indicator_ree.asp (visited August 2019).

[3] Betsy Bizot and Stu Zweben. 2019. Generation CS, Three Years Later. On the Internet at https://cra.org/generation-cs-three-years-later/ (visited August 2019).

[4] National Science Board. 2019. *Science and Engineering Indicators 2018.*

[5] EuroStat. 2018. Girls and women under-represented in ICT. On the Internet at https://ec.europa.eu/eurostat/web/products-eurostat-news/-/EDN-20180425-1 (visited December 2019).

[6] Katrina Samaras. 2005. Indigenous Australians and the Digital Divide, International Journal of Libraries and Information Studies, Vol 55, Issue 2-3 (Jan 2005).

[7] National Academies of Sciences, Engineering, and Medicine. 2018. *Envisioning the data science discipline: the undergraduate perspective: interim report*. National Academies Press.

[8] AAAS, *Levers for Change: An assessment of progress on changing STEM instructio*n (funded by NSF DUE-11414449) https://www.aaas.org/resources/levers-change-assessment-progress-changing-stem-instruction?et_rid=79380746&et_cid=3116297 (visited December 2019).

[9] Renata Rawlings-Goss et al. 2018. *Keeping Data Science Broad: Negotiating the Digital and Data Divide Among Higher-Education Institutions*, South Big Data Innovation Hub. https://drive.google.com/file/d/14l_PGq4AxOP9fhJbKqA2necsJZ-gdiKV/view (visited December 2019).

[10] https://cloud.tencent.com/developer/news/321995 (visited December 2019; with translation by Hongzhi Wang).

[11] https://programminginsider.com/tomorrows-jobs-5-fastest-growing-jobs-in-the-tech-sector/ Programming Insider, Nov 2019. (visited December 2019).

[12] Rock, D., & Grant, H. 2016. Why diverse teams are smarter. *Harvard Business Review*, *4*(4), 2-5.

[13] Barker, L., Mancha, C., Ashcraft, C. 2014. *What is the Impact of Gender Diversity on Technology Business Performance? Research Summary*, NCWIT.

https://www.ncwit.org/sites/default/files/resources/impactgenderdiversitytechbusinessperformance_print.pdf (visited December 2019).

[14] Cynthia Harvey. 2019. Top-Paying U.S. Cities for Data Scientists and Data Analysts. InformationWeek, Nov 15, 2019. http://www.informationweek.com/top-paying-us-cities-for-data-scientists-and-data-analysts/d/d-id/1336248 (visited December 2019).

[15] *Final results of the European Data Market study measuring the size and trends of the EU data economy*. 2017. https://ec.europa.eu/digital-single-market/en/news/final-results-european-data-market-study-measuring-size-and-trends-eu-data-economy
(visited December 2019).

[16] Luis Ponjuan. 2011. Recruiting and Retaining Latino Faculty Members: The Missing Piece to Latino Student Success. National Education Association.
http://www.nea.org/archive/49914.htm (visited December 2019).

[17] Inclusive Pedagogy. Center for New Designs in Learning and Scholarship.
https://cndls.georgetown.edu/inclusive-pedagogy/ (visited December 2019).

[18] Lecia Barker and Joanne Cohoon. 2009. Key Practices for Retaining Undergraduates in Computing. NCWIT.
https://www.ncwit.org/sites/default/files/resources/keypracticesretainingundergraduatescomputing_final.pdf

[19] L. A. Lyon and J. Denner. 2017. Community Colleges: A Resource for Increasing Equity and Inclusion in Computer Science Education, Communications of the ACM, Vol. 60 No. 12, Pages 24-26.

Chapter 6:
Characteristics of Data Science Graduates

Graduates of Data Science (DS) programs should have fundamental competency in all areas described by the Body of Knowledge. However, there are also competencies that graduates of DS programs should have that are not explicitly listed in the Body of Knowledge.

Data Science graduates should have an appreciation and knowledge of foundational work in Computer Science, Mathematics, Statistics, and other applied domain areas contributing to Data Science. Data Science graduates should be aware that while DS is a new discipline, it is founded on powerful mathematical, statistical, and computing foundations and approaches. When faced with a new problem, they can and should draw upon existing techniques when appropriate. Below, we describe the characteristics that we believe should be attained at least at an elementary level by graduates of Data Science programs. These characteristics will enable their success in the field and further professional development. Some of these characteristics and skills also apply to other computing fields. They are included here because the development of these skills and characteristics should be explicitly addressed and encouraged by Data Science programs.

Preparation in basic mathematical, statistical, and computing skills:

Data Science graduates should have acquired some basic education in computing (programming, databases, use of the Internet); be able to program on their own with one or two common languages (Python, R); be aware of some common libraries such as sklearn in Python, R packages, and several method or domain specific libraries; and be able to learn new languages and new libraries when needed. They should be familiar with concepts in applied mathematics, covering multi-variate calculus, linear algebra, optimization, and graph theory, in addition to concepts in probability and basic methods in statistics. They should be sufficiently literate to read practitioner-oriented papers that contain descriptions of methods in basic mathematics and statistics terminology, and high-level descriptions of algorithms and experimental results. They should also be familiar with several public data depositories that may serve as application examples to illustrate methodologies.

Flexibility and joy of learning and working in a fast-paced discipline:

Data Science is a rapidly evolving discipline that sees development of new techniques, tools, and application domains every day. A DS graduate needs to be prepared to pursue continuous learning from colleagues, professional peers, and the community through various on-line sources and conferences, and be able to learn from experience.

Broad interest in application domains, and preferably, with a few specializations:

Most Data Science graduates will work in an application discipline, driving forward exciting change and innovation in a business or other type of institution. Analytic methods learned in

school are often generic and neutral to application domains. A Data Science graduate needs to understand the mission, challenges, and constraints of the application domain so as to guide the focus of the analysis and the selection of methods. This requires broad interests, strong curiosity, fast learning, accurate communication, and empathy of and dedication to the clients' key concerns. These abilities are often amplified when the data science graduate has passion and intrinsic interest in the application domain.

Staying alert of broader societal, non-technical concerns:

The work environment of Data Science graduates operates in and interacts with a broader society through its lines of business. Collecting and using data from certain sources may need to take into account the societal impact. Data Science graduates should stay knowledgeable of current affairs, and stay alert of any sensitivity on privacy, security, and bias related issues, so as to plan and deliver studies with such concerns in mind.

Strong communication skills:

Data science graduates will need to learn about a topic of concern from clients of the hosting institution, and deliver results back to the clients in an understandable way.

Knowledge and goal acquisition: The clients are often experts in other areas or are decision makers, and are not familiar with data science technology. The Data Science graduate needs to be able to identify the relevant analytic issues from the client's concerns through dialogs or documents in the client's preferred terminology. These require clear thinking, fluent communication skills, ability to ask the right questions, in addition to good interpersonal skills for building lasting engagement and collaborations.

Result delivery: Likewise, on presentation of results, the Data Science graduate needs to explain and interpret the numerical conclusions in the client's terminology, and deliver text and graphics ready to be digested by non-technical personnel.

Commitment to professional responsibility:

Graduates need to be aware of relevant social, legal, ethical, and cultural issues when working with various data sets and deriving meaning through data science. In addition, awareness and understanding of the different international cultural and legal implications is critical. As practicing professionals, they must understand their responsibility and the possible consequences of their work. They must understand the limitations of the tools they use and their own personal limitations. A strong ethical dimension in the work of a graduate is important to be alert to bias, both in the tools they use and in data and the sampling of data.

Chapter 7: Challenges for Institutions

Institutions offering an undergraduate degree in Data Science face a number of challenges. Who should host such an offering? An Institute or School of Data Science? A Department of Data Science? Should it be an interdisciplinary program? Or should it be housed as one of many offerings within a more broadly based unit or set of academic units? The particular choice is likely to depend on the philosophy underpinning institutional organization: is it research based, is it based on industrial considerations or is it education based? Inevitably personalities will play a role, reflecting the expertise, perspectives, interests and ambitions of individuals.

Administrative Challenges

The interdisciplinary nature of Data Science can create challenges for the administrative structure of an institution. It is vitally important that there is in place a structure that is fully supportive of the program, giving due attention to the wellbeing of the students and the faculty involved in the delivery of the curriculum. Support needs to be provided to ensure the ongoing health and development of the discipline and its promotion both within the institution and within the wider community.

Marketing of the Program

The marketing of a degree program will be a central concern. To establish a certain momentum, it is important to have cohorts of a reasonable size. Yet Data Science is a relatively new discipline, unfamiliar to many career advisors, parents or guardians who can exercise influence, or even to applicants themselves. Accordingly, information about the program has to provide a sharp and focused description of the discipline, provide a brief outline of what is involved in studying the discipline, explain some of the major accomplishments and relevance of Data Science, and give an indication of career opportunities. Further, it needs to be prominent and of high quality with an identified faculty contact.

One area that merits particular concern is ensuring that the program is attractive to diverse audiences, including, for example, women and ethnic minorities. This necessitates a holistic approach involving broad consideration of all aspects from marketing, to the curriculum, to the ethos and atmosphere associated with the degree program.

One way of attracting interest from students is to offer a very imaginative and highly motivating course to first year students and to have that open to a broad spectrum of students. If students find this sufficiently appealing they may well opt for Data Science as a major.

Faculty Considerations

In some institutions, often in the US, departments are being encouraged by administration (with pressure from trustees, in some cases) to start Data Science programs. Those most frequently pushed in this direction are in Computer Science, Statistics and Mathematics departments who may already be facing staffing challenges and growing numbers of students. Then strong leadership is required to address such challenges.

Central to the delivery of a Data Science program is a collection of faculty members who have expertise in research / scholarship in data science. Of course they should be committed educators who can motivate and properly support students. Faculty should be particularly alert to the needs of the students. Given the newness of Data Science and the breadth of it, students will be taking courses for a Data Science program/major from a wide range of faculty across an institution. This poses a challenge in ensuring that all faculty are on board with the aims, objectives and ethos of the program and reflect this in their approach.

A valuable aspect of study is often lectures given by invited lecturers from industry. Faculty members can assist in organizing relevant talks and meetings. Student societies, prizes for exceptional student work, drawing attention to competitions and opportunities as well as having appropriately imaginative displays are all aspects that faculty can facilitate.

Implementation of the Curriculum

This Data Science volume provides guidance on education at the undergraduate level on the computing aspects of a Data Science program. Its instantiation in any one institution is likely to be governed by many factors such as the wider environment, the number of students in a cohort, and so on. Institutions are encouraged to consider how best to integrate the recommendations of this report into the existing provisions. A flexible approach is encouraged (with institutions 'playing to their strengths' in Computer Science, Statistics and Mathematics); the identification of T1, T2 and E learning outcomes has been undertaken to further encourage and facilitate just that.

Students will inevitably experience applications during their study and the range should be broad. Beyond this, through options / electives students can be given the opportunity to study topics in some detail so that they can be better informed about possible areas of application.

In the teaching of Data Science, the ethical issues should be seen to pervade the whole curriculum. The privacy and confidentiality of information is a vital matter and must be treated with the utmost care and attention. During their education as well as in their professional lives, ethical issues have to be seen to receive attention.

Data Science Minors

A Data Science minor will include a number of core modules but will fall short of a full program. The intention should be to allow students from other disciplines to gain a solid understanding of the principles and potential of Data Science with a view to their application in other fields of endeavour.

An important aspect of the existence of minors is the creation of links with other departments. Such links should lead to greater understanding and hopefully mutually beneficial collaboration.

Resources

In terms of resources to support Data Science, languages (such as Python) are an important consideration. It is desirable that the language implementation should include libraries to support Data Science and so various aspects of visualization, statistics, and machine learning. Moreover, there should be software to support the teaching of relevant mathematics.

But different communities prefer different languages, popular alternatives to Python being Scala or Java; Python alone is often considered ineffective for processing Big Data. Then there should be available multi-server cluster resources such as an Apache Spark or an Apache Hadoop cluster, or other cluster resource that facilitates the storage and processing of Big Data. The cluster might be on the premises (on-prem) of the home institution or in a public, private, or hybrid cloud. A single sever is not sufficient.

Again, for the purposes of dealing with Big Data, tools for topics such as visualization, machine learning, etc. need to run in a distributed fashion. Running on a single server is typically not sufficient.

Most clusters are Linux-based. Accordingly students will need access to Linux command line training to learn how to navigate and run code from the command line. There need to be cluster administrators who maintain the cluster hardware and software and support the user community.

Beyond the basic languages and related tools, students should have access to web services that are relevant to Data Science as well as data sets of various kinds. During their studies the attention of students should be drawn to important collections of data.

Appropriate financial resources may be needed to ensure appropriate access to staff, equipment, resources, laboratories, web services, etc.

Keeping current

Given the rate at which Data Science is evolving as a discipline, students need to be guided in how to keep current in terms of their knowledge and skills. The availability of online material is important in this regard. One possible set of resources are offered as MOOCs; students would benefit from guidance on how to use a wide variety of relevant resources effectively and indeed how to identify those of quality.

Remaining current can also be facilitated by being a member of a network of similarly minded professionals. Again guidance can be provided on how best to proceed on this topic.

Appendix A
The Body of Knowledge: Computing Competencies for Data Science

This appendix contains 11 Knowledge Areas (KAs):

- Analysis and Presentation (AP)
- Artificial Intelligence (AI)
- Big Data Systems (BDS)
- Computing and Computer Fundamentals (CCF)
- Data Acquisition, Management, and Governance (DG)
- Data Mining (DM)
- Data Privacy, Security, Integrity, and Analysis for Security (DP)
- Machine Learning (ML)
- Professionalism (PR)
- Programming, Data Structures, and Algorithms (PDA)
- Software Development and Maintenance (SDM)

The KAs are further divided into sub-domains. Competencies (with "tiers" – i.e., the recommended level of requirement for a Data Science degree) are given for each.

Analysis and Presentation (AP)

The human computer interface provides the means whereby users interact with computer systems. The quality of that interface significantly affects usability in all its forms and encompasses a vast range of technologies: animation, visualisation, simulation, speech, video, recognition (of faces, of hand-writing, etc.) and graphics. For the data scientist, it is important to be aware of the range of options and possibilities, and to be able to deploy these as appropriate. Through the use of graphs and other forms of diagrams, visualisation can be used in providing readily understood summaries but can also greatly assist in guiding such activities as clustering and classification.

<table>
<tr><td>Scope</td><td>Competencies</td></tr>
<tr><td>

- Importance of effectively presenting data, models, and inferences to clients in oral, written, and graphical formats.
- Visualization techniques for exploring data and making inferences, as well as for presenting information to clients.
- Effective visualizations for different types of data, including time-varying data, spatial data, multivariate data, high- dimensional multivariate data, tree- or graph-structured data, discrete / continuous data, and text.
- Knowing the audience: the client or audience for a data science project is not, in general, another data scientist.
- Human-Computer Interface considerations for clients of data science products.

</td><td>

- Gain familiarity with the main strands of knowledge underpinning approaches to Analysis and Presentation
- Provide the range of skills and techniques (including tools) that can be employed in addressing the challenges of Analysis and Presentation and creating efficient and effective interfaces
- Instill a critical demeanor but also confidence and creativity regarding all aspects of the human computer interface
- Select tools appropriate for the size of the data/Big Data to be rendered

</td></tr>
<tr><td colspan="2">Sub-domains</td></tr>
<tr><td>AP-Foundational considerations – T1
AP-Visualization – T1
AP-User-centered design – T2
AP-Interaction design – T2
AP-Interface design and development – E</td><td></td></tr>
</table>

AP-Foundational considerations – T1

Presenting data in a suitable form is a challenging but important endeavor. For the data scientist this is fundamentally enabling them to display data in a form that is attractive to users / audiences and readily and appropriately understandable, but is also potentially of great value in providing insights and characteristics including underlying structure. Fundamentally it influences usability.

Knowledge

- Contexts for addressing the human computer interface: visualization of data, web pages, multimedia material, instructional material, the general computing environment paying attention to navigational considerations
- The role of theories, models, principles, guidelines, standards
- Different measures of effectiveness and attractiveness of an interface
- The use of color and multimedia as well as ergonomics and web services
- Cognitive models that influence interaction
- The concept of augmented reality
- Software support to assist with the perception regarding analysis and presentation
- Accessibility considerations for different groupings of users including those with special needs

Skills

- Justify the adoption of a user centred approach to analysis and presentation
- Describe how considerations of attention, perception, recognition, speech, movement affect the usability of an interface and indicate their role in analysis and presentation
- Explain the desirable impact of differently-abled users and differently aged groups (including children) on interfaces
- Outline ways in which bias may be perceived in interfaces
- Outline the range of software that can be employed in support of analysis and presentation

Dispositions

- Positive recognition of the vital role of an interface in affecting all aspects of usability

AP-Visualization – T1

Different kinds of data benefit from different approaches to their Visualization. Data scientists need to be aware of this and be familiar with the techniques to be employed in any particular situation.

Knowledge

- The role of visualization in Data Science
- Illustrations, including both historical and contemporary examples, of visualization
- Characteristics of effective visualization

- Suitability of different techniques for different data and for different users
- Dashboards and interactive visualisation
- Software to support visualization
- Inference based on visualization
- Preparing for visualization - scaling, the role of color
- Chart types - tables, scatter plots, pie charts, histograms, graphs, data maps including pixel-, glyph-, graph- and map-based representations

Skills

- Identify famous examples of visualization in common use
- Identify the various roles that visualization can play in Data Science
- Given a set of data that has to be used for a particular purpose, identify and implement an effective approach to its visualization
- Describe the role of visualization in classification and categorization and identify approaches that facilitate this

Dispositions

- A heightened appreciation of the role of visualization

AP-User-centred design – T2

The fundamental approaches to the design of interfaces that benefit users are explored. Inevitably testing is involved to provide assurances about a successful outcome.

Knowledge

- The user-centred design process
- Relevant life cycle models and standards
- Interaction design patterns, visual hierarchy, navigational considerations
- Identification and capturing of functionality and requirements
- Quality considerations including completeness and consistency and checking for these
- Prototyping
- Design for resource constrained situations (e.g. mobile devices)
- Maintenance considerations
- Relevant software support

Skills

- Discuss a range of approaches to prototyping, identifying the strengths and weaknesses of the various approaches
- Given a particular application, describe approaches to checking the consistency and completeness of functionality and requirements
- Discuss the role of statistics in evaluating interfaces
- Identify standards, languages and tools that support the design of high quality user interfaces

Dispositions

- Inculcate a positive and creative attitude to, and efficient approaches to, the design of high quality interfaces

AP-Interaction Design – T2

It is desirable to review the range of issues that have to be addressed and the techniques that can be employed. Best practices (at the time of their creation) will have been captured in appropriate user interface standards

Knowledge

- The various possible roles of an interface; issues associated with addressing the main possibilities
- Implications of collaborative activity
- Characteristics of high quality interface design
- Approaches to the evaluation of interfaces including walkthroughs, experiments, heuristics
- Consideration of color, multimedia, speech recognition, animation, touch and gestures
- Data driven applications (with database back end)
- Handling failure, help facilities
- Addressing accessibility considerations
- User interface standards

Skills

- Evaluation of interfaces' effectiveness for a variety of tasks and a variety of purposes and users
- Identify a national and an international user interface standard and the implications of adherence to them
- Explain the possible implications of collaborative activity on interaction design
- Describe and explain the importance of the design parameters that have to be addressed in creating educational material

Dispositions

- Heightened awareness of the possibilities regarding the computer interface
- Recognition that there are both national and international user interface standards

AP-Interface design and development – E

The data scientist has to be able to apply a range of programming techniques to the creation of ever more effective interfaces

Knowledge

- Software architecture patterns
- GUI libraries
- Interaction styles and interaction techniques

- Software support including GUI libraries
- Interface animation techniques
- Role of animation and multimedia in interfaces

Skills

- Explain the importance of software architecture patterns and interface design patterns to interface design
- Explain the problems associated with navigation in interface design, and how to address these
- Create a GUI interface for a given data science application
- Explain the considerations in creating an interface for a resource constrained device

Dispositions

- The creation of a positive and yet critical approach to interfaces

Artificial Intelligence (AI)

Artificial Intelligence (AI) includes the methodologies for modelling and simulating several human abilities that are widely accepted as representing intelligence. Perceiving, representing, learning, planning, and reasoning with knowledge and evidence are key themes.

Concepts and methods developed for building AI systems are useful in Data Science.
For example, knowledge graphs such as semantic ontologies are both used and generated by data scientists. Computer vision algorithms can be used in analysis of image data; speech and natural language processing algorithms can be applied in analysis of speech or text data. Machine learning algorithms are applied extensively to extract patterns from data. Thus a student who is well versed in AI will be able to apply those techniques in a Data Science context.

Conversely, Data Science methods are applied extensively in AI systems. Data Science students should have an understanding of AI systems and the way they work, if they plan to apply their work to AI.

Due to their centrality in Data Science, AI competencies related to images, text, and machine learning are highlighted elsewhere. Working with images and text is in the Data Acquisition, Management and Governance KA; Machine Learning is its own KA but is also referenced extensively in the Data Mining KA. This knowledge area addresses knowledge representation, reasoning, and planning.

Scope	Competencies
● Major subfields of AI ● Representation and reasoning ● Planning and problem solving ● Ethical considerations	● Describe major areas of AI as well as contexts in which AI methods may be applied. ● Represent information in a logic formalism and apply relevant reasoning methods. ● Represent information in a probabilistic formalism and apply relevant reasoning methods. ● Be aware of the wide range of ethical considerations around AI systems, as well as mechanisms to mitigate problems.
Sub-domains	

AI-General – T1, T2 AI-Knowledge Representation and Reasoning (Logic-based models) – T2, E AI-Knowledge Representation and Reasoning (Probability-based models) – T1, T2, E	AI-Planning and Search Strategies – T2, E

AI-General

Given the utility of AI approaches for knowledge representation and inference, a data scientist should be aware of their range and history. A data scientist should develop a good sense of existing work in order to know where to look for possible solutions to the full range of possible problems one might encounter.

Knowledge
T1:

- History of AI
- Reality of AI (what it is, what it does) versus perception
- Major subfields of AI: knowledge representation, logical and probabilistic reasoning, planning, perception, natural language processing, learning, robotics (both physical and virtual)

Skills
T1:

- Describe major branches of AI in order to recognize useful concepts and methods when needed in Data Science

T2:

- Articulate *what* AI systems are and *that* they both collect and use data to implement AI as well as collect and generate data that can be used by data scientists.
- Describe qualitatively *how* robots (physical or virtual), agents, and multi-agent systems collect and use data to embed, deliver, or implement artificial intelligence.
- Describe data collected and produced by AI systems that can be useful for data science applications.

Dispositions
T1:

- Appreciate that AI is not a new field, but rather one with a long and rich history.

T2:

- Know what the major areas of concern are in AI, as well as the types of problems they address, in order to know where to look for approaches when needed, thus avoiding re-discovery of existing methods.

AI-Knowledge Representation and Reasoning (Logic-based Models)

For certain types of problems, methods of formal logic can be appropriate for representing information and performing inference. A data scientist should be aware of such approaches and know how to map them to inference problems.

Knowledge
T2:

- Predicate logic and example uses
- Automated reasoning: forward chaining, backward chaining
- Reasoning integrated into large-scale systems (e.g., Watson)
- Ontologies, knowledge graphs (e.g., protege, ConceptNet, YAGO, UMLS)

Elective:

- Automated reasoning: resolution, theorem proving
- Languages for automated reasoning

Skills
T2:

- Express natural language statements in predicate logic.
- Express predicate logic statements in natural language.
- State example uses and limitations of predicate logic.
- Name example algorithms and/or systems for efficient automated reasoning.
- Describe automated reasoning in a logic-based framework by, for example, forward or backward chaining.
- Give examples of cases where reasoning is integrated into large-scale data-driven systems (e.g., Watson)

Elective:

- Describe a specific method for automated theorem-proving.
- Describe what an ontology is, giving examples of existing technologies, contexts in which they can be used (e.g., question answering), and how they are used (e.g., to aid in disambiguation).
- Describe how ontologies are constructed.
- Implement a medium-sized reasoning problem.

Dispositions
T2:

- Appreciate the benefits and limitations of logic-based representations of knowledge.
- Be aware of the rich history behind formal logic and logic-based algorithms, in order to draw upon them for specific applications.

AI-Knowledge Representation and Reasoning (Probability-based Models)

Probability models lie at the heart of many inference techniques for data science. A data scientist should be aware of a wide range of ways in which information can be modeled in formal probability-based systems.

[Note: The items designated T1 in this knowledge area will likely move to a new KA when a joint task force develops complete curriculum guidelines for Data Science.]

Knowledge

T1:

- Fundamental concepts: random variables, axioms of probability, independence, conditional probability, marginal probability. (x-ref Probability, a fundamental knowledge area for DS, not computing discipline-specific)
- Causal models

T2:

- Bayesian networks
- Markov Decision Processes (MDPs)

Elective:

- Reinforcement Learning
- Probabilistic logic models (e.g., Markov logic networks)

Skills

T1:

- Justify the need for probabilistic reasoning.
- Define fundamental concepts such as random variables, independence, etc.
- State axioms of probability.
- Use the above fundamental concepts and axioms to model a simple system and answer questions.
- Describe what causal models are, and how they may be used.

T2:

- State what a Bayesian network is, giving a small- or medium-sized example.
- Demonstrate contexts in which Bayesian networks can be useful (e.g., diagnostic problems).
- Demonstrate how Bayesian networks can be used to make inferences; understand that exact reasoning is intractable in most cases; state examples of approaches for more efficient reasoning (e.g., Belief Propagation).
- Identify independence relationships implied by a Bayesian network.
- State what a Markov Decision Process is, giving a small or medium sized example.
- Demonstrate contexts in which MDPs can be useful (e.g., optimization or control problems).
- Demonstrate how MDPs can be used to make inferences.

Elective:

- Construct a Bayesian network for a small- or medium-sized problem.
- Apply a learning algorithm to construct a Bayesian network for a small- or medium-sized problem.

- State how the parameters of a MDP can be learned. Give examples of algorithms that can be used to do so.
- Apply a reinforcement learning algorithm to an appropriate problem.
- Give examples of probabilistic logic models, such as Markov logic networks, identifying applications for which they are useful.
- Apply a probabilistic logic model to a small- or medium-sized problem.

Dispositions
T1:
- Appreciate the benefits and limitations of probability-based representations of knowledge and methods for performing inference over them.

AI-Planning and Search Strategies

Beyond representing and reasoning about the world, AI methods allow for planning a step-by-step solution and then carrying it out. A data scientist should be aware of these techniques in order to apply data-driven methods to improve performance or to understand how to gather data from such systems. Note that while several of the methods included here (e.g., breadth- and depth-first search) also appear in the KA on Programming, Data Structures, and Algorithms.

Knowledge
T2:
- State space representation of possible solutions to a problem
- Breadth- and depth-first (i.e., uninformed) search of a state space
- Heuristic (i.e., informed) search of a state space (e.g., A* search)

Elective:
- Stochastic search algorithms (e.g., genetic algorithms, simulated annealing)
- Constraint satisfaction problems and methods

Skills
T2:
- Explain how a solution to a problem can be viewed as a state in a space of possible solutions (e.g., assignments of values to variables).
- For a given problem, model it as search in a multidimensional state space.
- Explain how breadth- and depth-first search can be used to search a space of solutions modeled as a graph.
- Explain how heuristics can be used to (potentially) speed up graph/state space search.

Elective:
- Apply uninformed search to find a solution to a problem modeled as a state space (where the graph representing the space is likely developed as the search is performed, rather than provided as input).
- Design a heuristic for a small problem.
- Apply an informed search approach to a small- or medium-sized problem.
- Apply a stochastic search approach to a small- or medium-sized problem.

- Explain how a stochastic search algorithm addresses issues of exploring a space (e.g., avoiding local minima); explain how a stochastic search algorithm addresses local search in a space of promising solutions.
- Explain how the solution to a problem may involve specific constraints on particular variables as well as their relationships to each other; describe methods for articulating these constraints.
- Implement search algorithms.
- Model a small problem as a constraint satisfaction problem.
- Apply a constraint-satisfaction algorithm to a small- or medium-sized problem.

Dispositions
T2:

- Appreciate that there may be multiple acceptable solutions in a state space, as well as multiple ways to find them. Different solutions or problem-solving approaches should be used depending on external conditions, such as the need for optimality, time constraints, etc.
- Appreciate the relationship between algorithm, heuristics, and optimality of solution to a problem.

Big Data Systems (BDS)

The term 'Big Data' has been coined to describe systems that are truly large; these might include, for instance, files of videos, images, handwriting, etc. that cannot be accommodate on a single server. Such systems introduce problems of scale: how to store vast quantities of data, how to be certain the data is of high quality, how to process that in ways that are efficient and how to derive insights that prove useful. These matters are addressed below under the headings of problems of scale, data storage, high performance computing, and complexity theory. These topics include a range of techniques typically used in addressing the problems of scale. Such systems can be complex and so consideration is given also to software support for Big Data applications.

<table>
<tr><td>Scope</td><td>Competencies</td></tr>
<tr><td>Problems of scale and the implications of Big Data on computation requirements
Theoretical and methodological issues employed in the context of Big Data
Appropriate algorithms to harness the processing power of the cluster
Approaches to simplifying the programming interface used in developing Big Data applications</td><td>Describe the main strands of knowledge needed to address Big Data applications, highlighting areas where collaboration is desirable
Provide familiarity with a range of skills that may be used in the implementation of Big Data applications
Instil confidence in dealing with the problems of Big Data</td></tr>
<tr><td colspan="2">Sub-domains</td></tr>
<tr><td>BDS-Problems of Scale – T1
BDS-Big Data Computing Architectures - E
BDS-Parallel Computing Frameworks - E
BDS-Distributed Data Storage – T2, E
BDS-Parallel Programming – T2
BDS-Techniques for Big Data Applications – T2</td><td>BDS-Cloud Computing – T2
BDS-Complexity Theory - E
BDS-Software Support for Big Data Applications – T2</td></tr>
</table>

BDS-Problems of Scale – T1

The computational problems associated with managing and processing very large amounts of data typically increase as the amount of data increases. Measurement provides insights into the rate of increase and the attendant computational consequences.

Knowledge

- The need for measurement in the context of Big Data, including size, capacity and timing
- The concept of the size of a problem
- Consequences of rapid rate of growth considerations for computation
- Storage consequences of rapid rate of data growth
- The need to place an emphasis on simplicity
- Approaches to addressing the problems of coordination with increasing numbers of agents / processes
- Approaches to addressing the problems of scale while accommodating scalability

Skills

- Outline reasons for Big Data applications leading to increased complexity, and give guidance on the nature of that complexity
- Give reasons for the importance of placing an emphasis on simplicity, though not excessive simplicity
- Describe steps that can typically be taken to reduce complexity
- Evaluation of data scale and speed for applications according to the descriptions and survey

Dispositions

- Create recognition of the difficulties created by scale
- Creating confidence in addressing problems of scale

BDS-Big Data Computing Architectures – E

An historical perspective suggests the former existence of two communities: one engaged in I/O intensive activities, the other engaged in compute intensive applications. The systems (preferred hardware and software) used by these communities were largely separate and customised to meet their needs. Recent developments, e.g. those involving advances in machine learning and deep learning, have tended to bring about a convergence of these communities with them now sharing all the facilities.

Knowledge

- Mechanisms that support fast and efficient input / output
- The concepts and requirements of data-centric high performance computing
- Memory considerations: cache considerations including cache coherence

- The various parallel computing architectures, their strengths and their limitations: multi-core, grid computing, GPUs, shared memory, distributed memory, symmetric multiprocessing, vector processing
- Flynn's taxonomy
- Instruction considerations in support of parallelism
- Parallel storage hierarchy

Skills
- Identify approaches to achieving fast input / output
- Explain the nature of any impediments to achieving fast input / output
- Compare and contrast the various parallel computing architectures
- Describe the nature of the applications to which the various parallel architectures are best suited
- Choose the system architecture that best suits a particular computation model and framework as captured in the computation patterns and data features

Dispositions
- Confidence in addressing hardware issues in support of Data Science applications

BDS-Parallel Computing Frameworks – E

Important high-level support is provided through parallel computation models for the generation of parallel programs.

Knowledge
- Definition and purpose of a parallel computation model
- Classification of models
- Distributed systems
- Grid search
- Process interaction: issues of communication and coordination
- Problem decomposition: task based decomposition, data-parallel decomposition

Skills
- Modelling parallel computation systems
- Evaluating parallel computations for efficiency and effectiveness
- How to design and deploy large-scale data processing parallel systems

Dispositions
- Confidence in evaluating but also designing potentially complex systems

BDS-Distributed Data Storage

Big Data applications benefit from approaches to data storage that are scalable, accommodate vast amounts of data, possibly straddling various machines, and yet facilitating processing within an appropriate time frame.

Knowledge
T2:

- Approaches to storing vast quantities of data, including storage across a range of devices
- Storage hierarchies
- Ensuring clean, consistent and representative data
- Protecting and maintaining the data
- Retrieval issues
- The benefits and limitations of a range of techniques used in addressing the problems of scale such as hashing, filtering, sampling
- Data backup

Skills
T2:

- Explain the role of the storage hierarchy in dealing with Big Data
- Outline advantages of certain kinds of redundancy in Big Data
- Demonstrate how unwanted redundancy may be removed efficiently from a Big Data set
- Describe approaches to protecting and maintaining data for a Big Data application, ensuring that it remains current and useful

Elective:

- Develop a distributed data storage system, choosing and producing arguments that support mechanisms that will scale
- Design storage systems with related strategies such as backup, migration and compression for data-centric systems to ensure scalability, usability, efficiency and security

Dispositions
T2:

- Create a positive disposition to the design of storage mechanisms in support of Big Data applications

BDS-Parallel Programming – T2

Parallel programming, whereby several activities may take place simultaneously, is an important approach to increasing the efficiency of programs. Novel forms of programming constructs are required to support this. In practice, new kinds of programming errors may result and there are limitations to the efficiencies that can be achieved.

Knowledge

- Concurrency and parallelism, and distributed systems
- Limitations of parallelism including the overheads
- Differing approaches to addressing concurrency and parallelism

- Parallel algorithms and how they best fit particular hardware architectures; load balancing issues
- Typical parallel programming paradigm such as MapReduce
- Complexity of parallel / concurrent algorithms

Skills

- Explain the limitations of concurrency / parallelism in dealing with problems of scale
- Identify the overheads associated with parallelism in particular algorithms
- Identify and implement methods for data-centric parallel programs
- Develop and deploy data-centric parallel computation systems according to the data scale and data operations
- Develop and optimize data-centric parallel programs
- Design, implement and tune algorithm with parallel programming paradigm

Dispositions

- Recognition that the overheads of parallelism can become excessive in particular cases
- Creating a level of confidence in dealing with parallel systems in appropriate cases

BDS-Techniques used in Big Data applications – T2

A number of techniques have been devised and, if deployed carefully, have proved valuable in increasing the efficiency of application programs.

Knowledge

- The need for techniques to assist with handling Big Data
- Hashing
- Sampling, filtering
- Data sketch and synopsis
- Limitations of hashing, sampling and filtering

Skills

- Illustrate the role of *hashing* in dealing with Big Data
- Explain a range of criteria that may be used in guiding sampling and filtering
- Perform sample selection, to conform to given guidelines, for a particular application involving Big Data
- Critically review a variety of approaches to filtering, illustrating their use
- Design data sketch and synopsis structure according to the available space and permit accuracy loss, and analyze the performance

Dispositions

- Be conscious of pitfalls such as bias in performing sampling and filtering

BDS-Cloud Computing – T2

The Cloud offers a number of advantages (over clusters, for instance) in the context of Big Data. It is important to understand these and be able to exploit them effectively; they include web services.

Knowledge

- The nature of Cloud Computing and its advantages
- The architecture of data center
- Risks associated with Cloud Computing
- Different approaches to supporting Cloud Computing
- Distributed file-systems
- Cloud Services in support of Big Data applications
- Virtualization technology
- Security issues for cloud including cloud computing, cloud storage and virtual machines

Skills

- Outline main tasks performed by the cloud system
- Design data center
- Identify the range of Cloud Services typically supplied in support of Big Data applications
- Select and apply Cloud Services that support particular Big Data applications
- Design security strategies for cloud

Dispositions

- Developing a responsible attitude to the use of Cloud Services

Context dependencies

- Different sets of Cloud Services are available, for instance, from Amazon, Google, Microsoft

BDS-Complexity Theory – E

An understanding of how to measure the efficiency of algorithms, both sequential and parallel, as well as the theoretical limitations to efficiency underpin an informed approach to Big Data applications

Knowledge

- Problems of computation and the efficiency of algorithms
- The notion of computational complexity, its use in the context of concurrency / parallelism and its importance in the context of Big Data
- Limitations to the concept of complexity
- Evaluation of the complexity of a range of commonly used algorithms including those exhibiting concurrency / parallelism

Skills

- Explain why mathematical analysis alone is not always sufficient in dealing with efficiency considerations
- Given a problem description with data size, time constraints and resource constraints, analyze whether the problem could be solved or solved approximately with some ratio bound from the aspect of complexity
- Demonstrate how to evaluate the efficiency of an algorithm to be used in processing Big Data
- Select algorithms appropriate to a particular application involving Big Data, taking account of the problems of scale

Dispositions

- Generate a positive attitude and confidence in dealing with complexity
- Recognition that there may be limits to complexity gains

BDS-Software Support for Big Data Applications – T2

Having access to a suite of high quality software tools that can be deployed and work together effectively can simplify the task of processing large data sets and elevate thinking away from detail and towards greater insight and innovation.

Knowledge

- The need for programming environments to support Big Data applications and the nature of these
- Concepts of auto scaling and serverless computing
- Review of the availability of sophisticated web services for the support of data movement, analytics and machine learning in the context of Big Data

Skills

- Compare and contrast the use of auto scaling and serverless computing
- Identify the relationship between load balancing and auto scaling
- Outline the impact of buffer size on streaming applications
- Identify web services that facilitate applications of face recognition and video streaming

Dispositions

- Encourage a reflective approach to the use of web services including possible bias, and other such deficiencies
- Generate confidence in dealing with Big Data applications
- Encourage attention to simplicity, but not excessive simplicity, in the context of Big Data applications

Computing and Computer Fundamentals (CCF)

Modern Data Science relies heavily on computing and on computing devices: to gather and store data; to analyze data; to present analyses and conclusions; and to field systems based on analyses and results. Therefore, a Data Scientist should understand -- at least at a high level -- the structure of operating systems, file systems, compilers, and networks, as well as security issues related to them.

Note that many of the competencies in this KA are taken or adapted from CS2013.
Note also that the majority of competencies in this knowledge area are intended to indicate high-level understanding and appreciation of concepts, rather than deep technical understanding.

<table>
<tr><td>Scope</td><td>Competencies</td></tr>
<tr><td>

- Digital representation of data
- Processors
- Memory management
- Operating system functions and vulnerabilities
- File organization
- Network structure and communication
- Web programming
- Compilers vs interpreters

</td><td>

- Appreciate ways in which digital representations of data affect efficiency and precision
- Know that there are different types of processors and configurations of them
- Understand the tradeoffs between expensive/fast memory and inexpensive/slower memory
- Appreciate the important role of an operating system and the ways in which it is both vulnerable to and can be protected from attack
- Know how to create, organize, and protect files
- Understand at a high level how networks are organized and how they transmit information
- Appreciate the web as an application layer on the internet and know how to use it to gather information and build useful applications
- Understand that while compilers and interpreters are both translators of code, they have relative benefits and limitations

</td></tr>
<tr><td colspan="2">Sub-domains</td></tr>
<tr><td>CCF-Basic Computer Architecture – T1, T2
CCF-Storage System Fundamentals – T1
CCF-Operating System Basics – T1, T2</td><td>CCF-File Systems – T1, T2
CCF-Networks – T1, T2
CCF-The Web & Web Programming – T1, T2
CCF-Compilers and Interpreters – T1</td></tr>
</table>

CCF-Basic Computer Architecture

A data scientist will benefit from understanding the ways in which digital representations of data affect precision, as well as the ways that different processor types and configurations can affect the efficiency of computation.

Knowledge

T1:

- "Power wall"
- Bits, bytes, and words
- Representation of numeric data
- CPUs and GPUs

T2:

- Representation of non-numeric data
- Multi-core and multi-processing
- Basic organization of the von Neumann machine
- Parallel architectures (e.g., SIMD, MIMD)

Skills

T1:

- Explain the implications of the "power wall" in terms of further processor performance improvements and the drive towards harnessing parallelism.
- Explain how fixed-length number representations affect accuracy and precision. [x-ref KA: Programming]
- Describe the role of CPUs; compare and contrast with the specialized purpose of GPUs.

T2:

- Describe the internal representation of non-numeric data, such as characters, strings, and images.
- Describe the difference between multi-core and multi-processor systems.
- Explain the organization of the classical von Neumann machine and its major functional units.
- Discuss the concept of parallel processing beyond the classical von Neumann model.

Dispositions

T1:

- Appreciate the benefits and limitations of data representation and processor speed in modern computing devices.

CCF-Storage System Fundamentals – T1

In contexts where data scientists are analyzing large quantities of data, they will benefit from knowing how those data are stored and moved during processing. This may be of help both in understanding the time needed to complete large analyses as well as in selecting hardware infrastructure and configurations to enable such work.

Knowledge

- Storage systems and their technology

- Registers, Cache, RAM
- Virtual memory

Skills

- Identify major types of memory technology (e.g., SRAM, DRAM, Flash, magnetic disk) and their relative cost and performance.
- Describe how the use of memory hierarchy reduces effective memory latency.

Dispositions

- Appreciate the tradeoff between expensive/fast memory and less expensive/slower memory.

CCF-Operating System Basics

Given the important considerations of security and privacy in data science analyses and applications, the data scientist will benefit from a high-level understanding of operating systems and the ways in which they are vulnerable to attack.

Knowledge

T1:

- Role and purpose of an operating system
- Types of security threats and mitigation approaches

T2:

- Networked, client-server, and distributed operating systems
- Reliability and availability

Skills

T1:

- Describe the objectives and functions of modern operating systems.
- List potential threats to operating systems (e.g., software vulnerabilities, authentication issues, malware) and the types of security features designed to guard against them.

T2:

- Discuss networked, client-server, and distributed operating systems and how they differ from single-user operating systems.
- Discuss the importance of reliability and availability; describe methods of fault tolerance for ensuring both.

Dispositions

T1:

- Appreciate the important role of operating systems in providing an interface between humans and system resources as well as between system resources; also understand the ways in which operating systems are vulnerable to attack and what to watch for.

CCF-File Systems

File systems provide the mechanism by which data and programs are organized. A data scientist should be aware of how individual files are stored, how they are organized in relationship to each other, and how they can be protected for purposes of security and privacy. A data scientist should know how to select the appropriate file system for the size of the data to be accommodated (e.g., for Big Data, a local file system on a single server would not be a good choice).

Knowledge
T1:

- Files: data, metadata, operations, organization
- Directories: contents and structure
- File protection

T2:

- Files: sequential, nonsequential

Skills
T1:

- Compare and contrast different approaches to file organization, recognizing the strengths and weaknesses of each.
- Describe levels of file protection and mechanisms for setting them.

Dispositions
T1:

- Appreciate the importance of good file organization as well as the importance of protecting files from inappropriate access.

CCF-Networks

Data and applications are shared over computer networks. Knowing how they work is helpful for understanding the ways in which data and applications are vulnerable to the introduction of errors, loss of information, or attacks, as well as the ways in which data and applications may be protected from those. In addition, knowledge of networks is important to understand cloud systems, Big Data clusters, and performance.

Knowledge
T1:

- Components of networks: hosts, routers, switches, ISPs, wireless access points, firewalls
- Local area networks; LAN topology (e.g., bus, ring)
- Organization of the Internet: Internet Service Providers (ISPs), Content Providers, etc.

T2:

- Circuit- vs packet-switched networks
- Layered network structure
- Naming and address schemes (DNS, IP addresses, Uniform Resource Identifiers, etc.)
- Basic protocols: TCP, IP
- HTTP / HTTPS as application-layer protocols

Skills

T1:

- List major components of computer networks
- Recognize that LANs can be organized in a variety of topologies.
- Articulate (at a high level) the organization of the Internet

T2:

- Explain the difference between circuit- and packet-switching
- Describe the layered structure of a typical networked architecture
- List the differences and relations between names and addresses in a network
- Describe how basic protocols such as TCP and IP work
- Describe how application-layer protocols such as HTTPS work

Dispositions

T1:

- Appreciate the complexity of transmitting information over a network, as well as the mechanisms for mitigating issues that can arise during transmission.

CCF-The Web and Web Programming

Data are frequently obtained via web applications. A data scientist should be able to write and use web applications, as well as appreciate the potential pitfalls of doing so.

Knowledge

T1:

- Relationship between Internet and World Wide Web
- Web programming languages (e.g., HTML5, Java Script, PHP, CSS)
- Awareness of web application vulnerabilities and security attacks (e.g., SQL injection, Distributed Denial of Service Attacks)

T2:

- Security attack detection and mitigation

Skills

T1:

- Describe the relationship between the Internet and the World Wide Web
- Design and implement a simple web application
- Describe common web application vulnerabilities and security attacks

T2:

- Be aware of and apply methods to protect against security attacks

Dispositions

T1:

- Appreciate the potential risks of writing and using web applications in order to do both as securely as possible.

CCF - Compilers and Interpreters – T1

Whether for purposes of gathering data, doing analysis, or fielding applications based on analyses, data scientists use and write software. Appreciating the purpose of and differences between compilers and interpreters can be useful in selecting programming languages and tools.

Knowledge

- Programs that take (other) programs as input: interpreters, compilers, type-checkers, documentation generators
- Interpretation vs. compilation to native code vs. compilation to portable intermediate representation
- Syntax and parsing vs. semantics and evaluation
- Examples of languages that fall into interpreted vs. compiled categories

Skills

- Explain how programs that process other programs treat the other programs as their input data
- Discuss advantages and disadvantages of interpreted vs compiled code
- Distinguish syntax and parsing from semantics and evaluation

Dispositions

- Appreciate the speed tradeoffs of interpreted vs compiled code.
- Appreciate the flexibility tradeoffs of compilation to native code vs portable intermediate representations.
- Appreciate the utility of interpreters during code development.

Data Acquisition, Management, and Governance (DG)

As the base of data science, data should be acquired, integrated and preprocessed. This is an important step to ensure both quantity and quality of data and improve the effectiveness of the following steps of data processing. Thus, a data scientist must understand concepts and approaches of data acquisition and governance including data shaping, information extraction, information integration, data reduction and compression, data transformation as well as data cleaning. In our ever increasing reliance on the quantity and quality of data in all forms of decision making, the data scientist has an ethical responsibility of protecting the integrity of data and proper use of data.

Scope	Competencies
● Shaping data and their relationships ● Acquiring data from physical world and extracting data to a form suitable for analysis ● Traditional Data Integration Methods: Pattern Mapping, Data Matching, Entity Recognition ● Integrating heterogeneous data sources ● Preprocessing and cleaning data for applications ● Improving data quality ● Ensuring data integrity including privacy and security	● Construct and tune the governance process according to the requirements of applications, including data preparation algorithms and steps. (Process Construction and Tuning) ● Define and write semantics rules for data governance, including information extraction, data integration and data cleaning (Rules Definition) ● Develop scalable and efficient algorithms for data governance according to the property of data and the requirements of applications, including information extraction, data integration, data sampling, data reduction, data compression, data transformation and data cleaning algorithm (Algorithm Development) ● Describe and discover the static and dynamic properties of data, changing mechanisms of data and similarity between data. (Property Description and Discovery) ● Develop policies and processes to ensure the privacy and security of data.
Sub-domains	

DG-Data Acquisition – T1, T2 DG-Information Extraction – T1, T2 DG-Working with Various Types of Data – T2 DG-Data Integration – T1	DG-Data Reduction and Compression – T1, T2 DG-Data Transformation – T1 DG-Data Cleaning – T1 DG-Data Privacy and Security – T1

DG-Data Acquisition – T1

As the initial step in data governance policies, data acquisition is the process of obtaining raw data from real-world objects. The process of data acquisition should fully consider the physical properties of the subject, and at the same time `consider the characteristics of the data application. Due to the limited resources available during data acquisition (such as network bandwidth, sensor node energy, website tokens, etc.), it is necessary to effectively design data collection techniques to maximize valuable data within limited resources and minimize valueless data. Also due to resource constraints, the data acquisition process is unlikely to obtain all the information of the data description object, so the data acquisition technology needs to be carefully designed to minimize the deviation between the collected data and the real objects.

Knowledge

- The sources of data
- Pull-based and push-based approaches
- Various data acquisition with the features of acquired data
- Data acquisition acceleration techniques
- Data discretization method
- Security and Privacy standards and best practices

Skills

T1:

- Select data source for the applications
- Design techniques for data acquisition according to the features of data sources and applications.
- Plan following steps including data discretization, transmission as well as storage.

T2:

- Design the acceleration and parallelization strategies for data acquisition according to the applications

Dispositions

- An ability to assess the trade-off between accuracy and efficiency in data acquisition.

DG-Information Extraction – T2

Information extraction (IE) is the task of automatically extracting structured information from unstructured and/or semi-structured machine-readable documents. It is an important technique to acquire data from documents, web pages, and even multimedia.

Information Extraction is relevant to the requirements of data acquisition and governance, but is described elsewhere in this report. See Information Extraction in the DataMining KA.

DG-Working with Various Types of Data – T2

Data comes in many forms. Some projects will rely completely on numeric data. Others will require processing of text or image or other media data. The data scientist must have an overview understanding of all types of data representation and processing, and must be competent to interact with some types of data as an expert.

Knowledge

- Data representation: numbers, text, images, data precision
- Text data processing: bag-of-words, word-count, TF-IDF, n-grams, Lexical analysis, syntax analysis, semantic analysis, stop word filtering, stemming, basic applications
- Image processing: data representation: multi-dimensional matrices of integers, features, image operators, video operators. Object recognition. Higher order feature extraction

Skills

- Write programs to perform basic operations on data of each type. Compute summary statistics, extract n-grams, do modifications to an image.

Dispositions

- Recognize the importance of choice of data type for encoding information.

DG-Data Integration – T1

In the data acquisition process, since the data may come from an autonomous data source, it is difficult to ensure the consistency of the data mode, modality, semantics, etc.. However, in many applications, these data from multiple autonomous data sources need to be summarized and used together to generate new value, this is the task of data integration, which is a crucial step for data acquisition and governance.

Knowledge

- The concepts and application scenarios of government database, data warehouse and mediator-based information integration
- The concepts and approaches of schema mapping
- The concepts and approaches of data mapping
- The concepts and approaches of data semantic transformation

- The techniques of cross-domain data integration

Skills

- Choose the scheme of data integration i.e. traditional data integration VS. cross-domain data integration
- Choose the architecture of data integration according to the features of applications
- Select or develop appropriate algorithms for schema mapping, data mapping and data semantic transformation
- Develop proper algorithms for cross-domain data integration

Dispositions

- Understand the challenges brought by heterogeneous data sources
- Know the roles of AI in data integration

DG-Data Reduction and Compression

The goal of data reduction and compression is to eliminate the redundancy of data and decrease the size of data involved in the next data processing steps. This involves data sampling, filtering and compression.

Knowledge

T1:

- The role of reduction and compression in data process
- Various data sampling approaches
- Data filter techniques
- Data compression techniques

Skills

T1:

- Determine whether data reduction and compression steps are required
- Perform data sampling and filtering

T2:

- Analyse the properties of data sampling
- Select data compression techniques according to the computation, communication and storage requirements
- Develop query-friendly data compression approach

Dispositions

- Understand the trade-off between data computation effectiveness and efficiency.

DG-Data Transformation – T1

Data collected from data sources often have different dimensions and ranges. These data may be correct, but they cannot be directly used. It is often necessary to transform the collected data and

convert the data into "appropriate" form to understand the data or visualize the data to achieve effective application of the data.

Knowledge

- Data Transformation pipeline
- Simple function transformation methods and their applications
- Data standardization and its applications
- Data normalization and its applications
- Data encoding approaches and their applications
- Data smooth approaches and their applications

Skills

- Evaluate and compare the dimension and range of data and those of the requirements in the applications.
- Determine the process of data transformation
- Choose proper data algorithms for the task
- Evaluate the effectiveness of data transformation

Dispositions

- Understand the importance of data transformation to data usage
- Have an understanding of the links between data transformation and data quality

DG-Data Cleaning – T1

Data quality is an important aspect of data usability. There is a perception that if data is “suitable for its intended use in operations, decision making, and planning,” it is generally considered to be of high quality. There are also views that if the data correctly represents the real-world entities that it refers to, then it is also considered to be of high quality. Data quality issues and the resulting knowledge and decision-making mistakes have had terrible consequences on a global scale. Data cleaning is an important solution for data quality problems.

Knowledge

- The dimensions of data quality
- The approaches to improve data quality
- Data cleaning algorithms including entity resolution, truth discovery, rule-based data cleaning.
- Various forms for data quality rules such as functional dependencies (FD), conditional functional dependencies (CFD), conditional inclusion dependencies (CIND), and matching dependencies (MD)

Skills

- Evaluate data quality
- Write rules for data cleaning according to the requirement of applications and data semantics
- Develop data cleaning pipeline according to the data quality requirements.

- Develop algorithms for efficient and effective data cleaning

Dispositions

- Hold an awareness of the harm of data quality problems
- Appreciation of and ability to handle the role of data cleaning in data usage.

DG-Data Privacy and Security – T1

Knowledge

- The relationships between individuals, organizations, or governmental privacy requirements
- The cross-border privacy and data security laws and responsibilities
- A comprehension of how organizations with international engagement must consider variances in privacy laws, regulations, and standards across the jurisdictions in which they operate.

Skills

- Explain how laws and technology intersect in the context of the judicial structures that are present – international, national and local – as organizations safeguard information systems from cyberattacks.
- Explain requirements of the General Data Protection Regulation (GDPR). And Privacy Shield agreement between countries, such as the United States and the United Kingdom, allowing the transfer of personal data.
- Describe how [Section 5 of the U.S. Federal Trade Commission, State data security laws, State data-breach notification laws, Health Insurance Portability Accountability Act (HIPAA), Gramm Leach Bliley Act (GLBA), and Information sharing through US-CERT, Cybersecurity Act of 2015] and other laws impact data security

Dispositions

- Have an appreciation for the ethical implications of data governance policies and actions
- Hold an awareness of the harm of data loss due to security and privacy failures
- Maintain the upmost ethical standards regarding legal and social responsibility for data

Data Mining (DM)

At its core, Data Mining involves the processing, analysis, and presentation of data in order to gain valuable information. An important pre-requisite is that appropriate data of a high quality has been prepared and is relevant to the task at hand. The basic types of analysis include clustering, classification, regression, pattern mining, prediction, association and outlier detection with attention being given to various forms of data including time series data and web data. Many of these concepts depend on the notion of data proximity.

Scope	Competencies
• Data mining and its relationship to data preparation and data management • Data mining models for a variety of data types and applications • Selection and application of data mining algorithms for various tasks	• Equip students with knowledge about the range of techniques available for mining data as well as the related algorithms and their suitability • Equip students with the ability to identify and use tools and techniques for mining data which may exist in various forms • Engender in students a high level of well-founded confidence in mining data
Sub-domains	
DM-Proximity Measurement – T1, T2 DM-Data Preparation – T1 DM-Information Extraction – E DM-Cluster Analysis – T1, T2 DM-Classification and Regression – T1, T2, E	DM-Pattern Mining – T2 DM-Outlier Detection – T2 DM-Time Series Data – E DM-Mining Web Data – T2 DM-Information Retrieval – T2

DM-Proximity Measurement

Various possibilities exist for measuring differences as well as similarities among data points. For numerical data the methods are typically phrased in terms of distance between two vectors. But measures for other types of data may include different notions of proximity (such as cosine similarity for text) or correlation. Special definitions may be needed, customized to particular situations.

Knowledge
T1:

- Basic properties of metrics
- L_k measure; special cases – Euclidean distance, Manhattan distance
- Use of scores and rankings; desirable characteristics of scores and ranking regimes
- Normalization of data to support comparison

T2:

- Metrics involving text
- Metrics such as correlation coefficient for sequences of data
- Metrics such as SimRank for similarity based on relationships, as in graphs
- Graph based metrics
- Metrics for measuring the similarity of time series, e.g. dynamic time warping

Skills

T1:

- Describe and compare measurement concepts and their relevance to different kinds of data – nominal, ordinal, interval and ratios
- Select metrics appropriate for comparison of various kinds of data

Dispositions

T1:

- A careful and critical yet imaginative approach to the use of scores and metrics recognizing that typically many approaches exist

DM-Data Preparation – T1

The availability and preparation of high quality data is essential to data science. There is the initial gathering of relevant data, possibly from a wide variety of sources, and then ensuring the data set is fit for purpose.

Knowledge

- Gathering data, its relationship to problem solving, importance of expert knowledge and being open to the views of experts
- Sources of data including databases, the Internet of Things, photographs and videos, online information sources; adequacy of data for particular purposes
- Ethical considerations around obtaining and using data for particular purposes; privacy concerns around collocating data; concerns around potential bias in data
- Munging data - dealing with errors in data, gaps in data, cleansing data, validating data, profiling data, transforming data, and joining datasets as appropriate; quality considerations
- Methods of dealing with dataset issues such as imbalance, insufficient or extraneous attributes; automated and manual approaches and trade-offs between these
- The concept of a ‘feature’; feature extraction and representation; feature selection and feature generation

Skills

- Illustrate the connection between the process of framing a question with the process of obtaining data to answer the question.
- Demonstrate the ability to understand a particular domain that involves interacting appropriately with experts.
- Demonstrate the use of summary statistics and data visualization in exploratory data

analysis and in making inferences.
- Describe issues that may arise with datasets and indicate their possible impact as well as how these might be resolved.
- Identify various methods of generating features and explain the benefits and implications of each.
- Describe the similarities and differences between feature selection and feature generation and demonstrate how feature generation can result in fewer or more features.

Disposition
- Instill a level of confidence in the selection and preparation of data as well as an understanding of the importance of dealing with quality data.

DM-Information extraction - E

Information extraction (IE) is concerned with the techniques and processes used to extract structured information from unstructured data that exists in different forms. It is an important technique used to acquire data from documents, web pages and even multimedia.

Knowledge
- Applications where information extraction plays a useful role
- Entity and relation extraction
- Rule-based information extraction approaches and their applications
- Statistics-based information extraction approaches and their applications
- The possible problems in the extracted data

Skills
- Design a schema according to the application requirements and data
- Write information extraction rules for an application
- Apply learning algorithms for information extraction tasks such as rule or model learning and relationship prediction

Disposition
- Appreciate that there are various approaches to extracting information from data.

DM-Cluster Analysis

Clustering involves grouping together data points that exhibit some element of similarity. This implies some interpretation of proximity and there can be various interpretations of that. Clusters in 2- or 3-dimensions can can often be identified on the basis of visualization but that is not always readily available especially in higher dimensions. Generally clusters may be compact and well separated but again this is not always the case. (See also ML-Unsupervised Learning.)

Knowledge

T1:

- Identification of appropriate similarity measure for clustering activity
- Clustering quality evaluation
- k-means clustering algorithm, including iteration considerations
- Density-based algorithms
- Applications of clustering

T2:

- Mean shift clustering
- Agglomerative clustering
- Grid-based algorithms
- Clustering algorithms acceleration and parallelization strategies

Skills

T1:

- Explain the importance of feature selection for clustering.
- Provide guidance on the selection of initialization criteria for the k-means algorithm.

T2:

- Compare clustering approaches, highlighting relative benefits and shortcomings.
- Indicate the circumstances in which the various clustering algorithms should be used, and when other alternatives are preferable.
- Apply algorithms to a test set of data and compare the results.
- Provide illustrations to highlight the utility and value of clustering.

Dispositions

T1:

- Create a positive and informed perspective on the role of clustering in Data Science.
- Appreciate the importance of scalable and efficient clustering algorithms for real scenarios.

DM-Classification and Regression

There are many application domains that involve assigning a class value to a (possibly complex) instance of data. Similarly, there are many application domains that involve assigning a numeric value to an instance of data. The former is referred to as classification. Regression involves estimating the relationship between a dependent variable and one or more independent variables. Though these are different tasks, they are related, and many data mining approaches can be adapted to both scenarios. A distinguishing feature of both is that they require labeled training data – i.e., representative samples that have been assigned class / dependent variable values. (See ML-Supervised Learning and ML-Deep Learning.)

Knowledge

T1:

- Considerations regarding feature selection for classification
- Instance-based methods such as K-Nearest Neighbor (KNN)
- Decision tree methods

- Probabilistic models, Naïve Bayes

T2:

- Rule-based methods
- Support vector machines
- Neural networks
- Real world applications of classification and regression
- Deep learning and related software support (such as Caffe, TensorFlow, PyTorch)

E:

- Acceleration and parallelization strategies

Skills

T1:

- Explain the importance of feature selection for classification and regression.
- Describe criteria that might lead to selection of one method over another, such as predictive accuracy, comprehensibility of the learned model, etc.

T2:

- Identify the relationship between regression and classification.
- Identify critical situations that may benefit from the use of classifiers or regression models.
- Identify software to support each of the approaches and apply the software.
- Demonstrate the ability to select and justify an approach to classification and to apply it to an example of modest complexity.

Dispositions

T1:

- Appreciate the importance of scalable and efficient classification and regression algorithms for real scenarios.

T2:

- Depicting links between classification and regression, and more generally statistics, as well as machine learning.

DM-Pattern Mining – T2

This topic is concerned with seeking patterns within data. For data collections of considerable size, brute force approaches are often computationally infeasible but selected algorithms provide a way forward. (Pattern matching has important applications in biotechnology through genome sequencing but that is not developed here.)

Knowledge

- The concept of association pattern mining
- Computational complexity considerations
- Association rule mining; Apriori and Frequent pattern (FP) growth algorithms
- Sequential pattern mining; the GSP algorithms
- Efficient and parallel algorithms for pattern mining
- Application areas

Skills

- Itemize a range of areas in which the Apriori algorithm may be used to beneficial effect in day to day settings.
- Identify an implementation of the Apriori algorithm and apply it to a significant application.
- Compare and contrast the utility of pattern mining algorithms.

Dispositions

- Recognize that pattern mining is a very broad topic with widespread applications.

DM-Outlier Detection – T2

An outlier is a data point that exhibits very different characteristics from the vast majority of other data. It is desirable to identify such data points since excessive attention to these can lead to distortion (and may even suggest maliciousness); though it is also important to understand the domain well enough to determine whether there are (legitimate) exceptional cases. In what follows it will be assumed that data has already been cleansed and that a true outlier is present.

Knowledge

- Definition of the concept of outlier
- General approach - develop a model of the data and then note that a data point does not fit
- Parametric methods, such as z-score to identify numeric outliers in 1-D
- Use of probability distribution functions
- Use of depth first approaches - having identified the expected convex hull of a set of points, is it inside or outside; use of related graphical approaches

Skills

- Apply algorithms for a range of outlier detection methods.
- Compare and contrast parametric and non-parametric approaches to outlier detection.
- Explain how outlier detection methods might assist with plagiarism detection, cases of financial fraud, network intrusion detection or other application areas.
- Illustrate the importance of outlier detection through appropriate examples.

Disposition

- Development of a critical yet broad perspective on outlier analysis and detection.

DM-Time Series Data – E

For certain kinds of data, the inclusion of time or data stamps is important. For instance, this can be used in measuring growth over time, or measuring traffic congestion during particular periods. See also ML-Mixed Methods.

Knowledge

- The nature of time series data, including comparison with sequential temporal data
- Data transformation - noise removal, data normalization of time series data
- Stationary and non-stationary time series
- Converting time series data to discrete sequence data
- Time series forecasting - predicting future values on the basis of past values
- Time series motifs - frequently occurring patterns in time series data
- Time series clustering and classification
- Outlier detection in time series - point outliers and shape outliers

Skills

- Provide a range of situations for which there is relevant time series data and indicate the importance of mining that data.
- Indicate through examples when converting time series data to sequence data is desirable.
- Explain techniques used for the clustering and classification of time series data.

Disposition

- Recognize that the data mining of time series data is highly relevant in certain critical applications.

DM-Mining Web Data – T2

Increasing amounts of data exist on the web, along with mechanisms for mining that data. As always in doing data collection and mining, ethical considerations should be observed.

Knowledge

- The processes of scraping and spidering / web crawling associated with web access
- Ethical guidelines associated with accessing web data
- The structure and functionality of software libraries for accessing web data
- Knowledge discovery approaches for web data such as community discovery and link prediction

Skills

- Compare and contrast the capabilities and ease of use of two different libraries for web access.
- Demonstrate the use software to scrape precise data from publicly available sites.
- Develop software to fetch data from the web according to given constraints.
- Develop efficient algorithms to discover knowledge from the web.

Disposition

- Encourage and facilitate access to high quality data taking account of the ethical framework.

DM-Information Retrieval – T2

Information Retrieval includes a disciplined approach to identifying and retrieving information from a larger (usually unstructured) data set. This should be seen to involve searching documents themselves, searching for documents, or searching the web. The documents may take a variety of forms: text, images, videos, sound recordings, etc. The manner in which data is stored initially can significantly influence the efficiency and effectiveness of the processes of retrieving information. Information retrieval is particularly important in certain areas such as in the context of digital libraries, or in extracting information from medical health records. There are strong links with the Data Mining Knowledge Area.

Knowledge

- Techniques used for measuring the efficiency of retrieval processes
- Range of approaches to storing and organizing data so that information can be extracted efficiently; the use of encoding functions
- The concept of a search strategy; the related role of narrowing and broadening
- Keyword(s) selection for the retrieval process; use of Boolean operators
- Search of ordered data
- Techniques for searching text-based material
- Searching a set of documents; strategies for listing the names of selected items
- Feature identification and extraction for non-text based data; searching strategies used with photographs, sound, video
- Role of hashing, indexing and filtering
- Approaches to searching text-based material
- Techniques for creating and searching relational database systems
- Various relational, non-relational, and other database formats
- Web-based information retrieval; the web viewed as a graph of interconnected nodes; relevant measures from graph theory; PageRank and related metrics that facilitate web-based search

Skills

- Devise a search strategy for a given information retrieval task.
- Explain ethical concerns that may be associated with the information retrieval processes.
- Identify opportunities for the use of parallelism to speed up search.
- Outline the main elements of an effective strategy underpinning web-based search.
- Identify software that can be used in information retrieval tasks associate with images, sound recordings and video clips.
- Create and use a relational database structure using SQL.
- Explain the roles that information retrieval may play in the operation of digital libraries.

Dispositions

- Recognition and appreciation of the importance of a range of considerations that should underpin an efficient and effective approach to information retrieval.

Data Privacy, Security, Integrity, and Analysis for Security (DPSIA)

Issues around privacy, security, and integrity are cross-cutting – that is, they relate to competencies in all of the Knowledge Areas. Therefore, this KA is somewhat larger than others. It is organized into sub-KAs, which are then further divided into sub-domains.

Data Privacy (DPSIA/DP)

Data scientists should be able to consider data privacy concerns and its related challenges when acquire, process, and produce data. They should recognize tradeoffs of sharing and protecting sensitive information and how domestic and international privacy rights impact a company's responsibility for collecting, storing, and handling data. Within the extensive area of cybersecurity, there are a number of concepts and subdomains that crossed referenced within cybersecurity knowledge areas in addition to Professionalism and Data Acquirement and Governance.

<table>
<tr><th colspan="2">DPSIA / Data Privacy</th></tr>
<tr><td>Scope</td><td>Competencies</td></tr>
<tr><td>● Interdisciplinary tradeoffs of privacy and security
● Individual rights and impact on needs of society.
● Technologies to safeguard data privacy.
● Relationships between individuals, organizations, and governmental privacy requirements.</td><td>● Evaluate and understand the concept of privacy, including the societal definition of what constitutes personally private information and the tradeoffs between individual privacy and security.
● Summarize the tradeoff between the rights to privacy by the individual versus the needs of society.
● Evaluate common practices and technologies and identifying the tools that reduce the risk of data breaches while safeguarding data privacy.
● Thoroughly comprehend how organizations with international engagement must consider variances in privacy laws, regulations, and standards across the jurisdictions in which they operate. This topic includes how laws and technology intersect in the context of the judicial structures that are present – international, national and local – as organizations safeguard information systems from cyberattacks.</td></tr>
<tr><td colspan="2">Sub-domains</td></tr>
<tr><td>DPSIA/DP-Social Responsibility–T1,T2,E
DPSIA/DP-Cryptography – T1, T2</td><td>DPSIA/DP-Information Systems – T1, T2, E
DPSIA/DP-Communication Protocols – T1, T2</td></tr>
</table>

DPSIA/DP-Social Responsibility

Summarize the tradeoff between the rights to privacy by the individual versus the needs of society.

Knowledge
T1:
- Data sensitivity that can be exposed by using social engineering and social media
- Tradeoffs between the right to privacy and the need of transparency through information dissemination
- Ethical responsibilities about disclosing, transmitting, and sharing information obtained from analytics tools

T2:
- Legal codes that involve scenarios on privacy concerns of using data to perform certain actions
- International privacy laws that impact society and computing development assets

Skills
T1:
- Express awareness about data sensitiveness when data is processed as an input.
- Identify scenarios where data cleaning must be considered before processing information.
- Apply techniques to provide data privacy to raw data processing, such as provide ranges or salting techniques.

Elective:
- Express awareness of global policy and regulations such as HIPAA, FCRA, ECPA, that may affect decision making.
- Express awareness of well-known search engines and their information storage policies that identify and jeopardize computer users' privacy.

Dispositions
T1:
- Understand that data provided to any entity may impact the loss of data privacy.
- Recognize the public and private implications in society of inappropriately handling data through computing systems or channels.

DPSIA/DP-Cryptography

Summarize the usage of cryptographic techniques to emphasize data privacy.

Knowledge
T1:
- Importance of encrypting data before transmitting it through any channel.
- Computational time tradeoffs of using encrypted vs non-encrypted data for statistical analysis.

T2:

- Differences between symmetric and asymmetric algorithms
- Hash functions for privacy checking and protecting authentication data
- Encryption algorithms

Skills

T1:

- Identify tools/mechanisms to encrypt data to reduce the risk of data breaches while keeping in mind computational performance.
- Be able to train different entities such as individuals, organizations, and government agencies about data encryption processes that impact privacy requirements.
- Illustrate the use of cryptography to provide privacy, such as message authentication codes, digital signatures, authenticated encryption, and hash trees.
- Identify the tradeoffs between processing plain text data and encrypted data.

T2:

- Analyze which cryptographic protocols, tools, and techniques are appropriate for providing data privacy, protection, integrity, authentication, non-repudiation, and obfuscation.

Dispositions

T1:

- Recognize the need for different mechanisms of encryption.

DPSIA/DP-Information Systems

Summarizing the concept of information systems by contextualizing information and the privacy of such by using well-known models.

Knowledge

T1:

- Concepts and techniques to achieve authentication, authorization, access control, and data privacy.
- Layered defenses to achieve maximum confidentiality, integrity, and availability (CIA).

T2:

- Different access control mechanisms that enforce data privacy such as Bell-LaPadula, Chinese Wall, Clinical Information Systems Security, to resolve different privacy and transparency conflicts of interest.
- Well-known information system designs and implementations and the impact on data privacy.

Elective:

- Traffic analysis to demonstrate how private information can be jeopardized in a given secure system.

Skills

T1:

- Explain how data privacy of a system might impact the security of the system.

- Discuss the trade-offs between data transparency and data privacy.

T2:

- Determine what information should be provided to a computer entity, balancing usability and privacy and how to report information.

Dispositions

T1:

- Protect information in a given computer system.

DPSIA/DP-Communication Protocols

Summarizing how communication protocols can be used to guarantee a secure communication over channels (secure and insecure); the consideration of cryptographic protocols used in communication protocols; and recognizing the impact on data privacy by using well-known applications' protocols.

Knowledge

T1:

- The importance of security protocols that enable secure communication over insecure channels
- The importance of privacy protocols enable private interactions over secure channels
- Internet/communication protocols that can guarantee private communication between applications and servers

T2:

- Balancing security protocols vs privacy protocols by using and not using cryptography

Skills

T2:

- Use security protocols to set up secure channels using different cryptographic primitives.
- Apply privacy protocols set up private channels using secure channels.

Dispositions

T2:

- Become aware of which available secure protocols to use to ensure a private connection between utilities.
- Recognize secure protocols that interchange data sets without jeopardizing privacy characteristics.

Data Security (DPSIA/DS)

This knowledge unit is focuses on the protection of data at rest, during processing, and in transit. It requires the application of mathematical and analytical algorithms to fully implement necessary security objectives over data-driven applications. This unit allows deeper understanding of data security objectives along with various tools to achieve them.

DPSIA / Data Security	
Scope	Competencies
● Cryptographic concepts: ○ Encryption/decryption, message authentication, data integrity, non-repudiation; Attack classification (ciphertext-only, known plaintext, chosen plaintext, chosen ciphertext); Secret key (symmetric), cryptography and public-key (asymmetric) cryptography. ● Threat models for data driven applications ● The role mathematical techniques play in producing useful encryption knowledge. ● Public key cryptography for data security ● The data security part of CSEC 2017 document provides additional scope.	● Describe the purpose of cryptography and list ways it is used in data communications; and which cryptographic protocols, tools and techniques that are appropriate for a given situation. ● Understand cipher, cryptanalysis, cryptographic algorithm, and cryptology ● Explain how public key infrastructure supports digital signing and encryption and discuss the limitations/vulnerabilities. ● Exhibit a mathematical understanding behind encryption algorithms ● Explain the difference and applications of Symmetric and Asymmetric ciphers. ● Analyze threats behind real time applications that consume/produce critical data ● Utilize attack vectors and attack tree concepts to model threats ● Explain how data over web and network can be protected
Sub-domains	
DPSIA/DS-Data quality and handling for security – T1, T2 DPSIA/DS-Classification of cryptographic tools – T2 DPSIA/DS-Security and performance trade-off – T2	DPSIA/DS-Network and web protocols – T1 DPSIA/DS-Privacy and data governance – see DPSIA/DP

DPSIA/DS-Data quality and handling for security

Knowledge
T1:

- Qualitative metrics
- Security importance of data assets
- Type of security objectives needed
- Data sources and assets
- Controlling and manging accessibility to data assets

T2:

- Attack vectors and trees
- Threat models of different use cases
- Impact of threats on data sources

Skills

T1:

- Understand data flow in applications.
- Derive important security objectives to achieve.
- Explain reasons for selecting what data assets to secure.

T2:

- Using data flow in applications, derive possible threats.
- Develop intuition for extracting threats on data-driven systems.
- Implement access control mechanisms to restrict data leaks.
- Enable required authentication processes for securely accessing data assets.
- Assess significance of data assets based on external and internal factors.
- Perform threat analysis on practical systems.
- Categorize threats based on their impacts.

DPSIA/DS-Classification of cryptographic tools – T2

Knowledge

- Cryptographic techniques
- Usability of various techniques and tools
- Cryptographic protocol designs using discrete mathematical concepts
- Public key cryptosystems vs. secret key cryptosystems

Skills

- Apply various cryptographic techniques to achieve necessary security objectives.
- Compare merits and demerits of various techniques.
- Explain performance characteristics of various techniques.
- List attack models for each cryptographic technique.
- Implement data security mechanisms using available cryptographic schemes.

Dispositions

- Recognize the importance and unique characteristics of various crypto protocols.
- Be able to select the right protocol depending on application requirements.

DPSIA/DS-Security and performance trade-off – T2

Knowledge

- Performance requirements of data driven applications
- Impact of security schemes on performance of applications

Skills

- Apply design principles to balance performance while applying security measures.
- Investigate the operational environments to characterize critical parameters that affect both performance and security of a system.
- Develop mechanisms that enable high data availability while achieving necessary security.

Dispositions

- Understand the performance and security trade-offs among different protocols.
- Recognize which ciphering technique to opt for based on application requirements.

DPSIA/DSC-Network and web protocols – T1

Knowledge

- Insight on data transactions over networks for data-driven applications
- Network and web protocols
- Available and/or enabled security modules in communication protocols
- Operations (storage, retrieval, remote compute) on data network and web

Skills

- Dissect and tune communication protocols to enable security.
- Explain the unique characteristics and working principles of protocols.
- Understand how data gets communicated to various entities over the network or web.

Dispositions

- Appreciate the importance of security countermeasures on running network/web protocols.

Data Integrity (DPSIA/DI)

This knowledge unit focuses on the completeness, accuracy, and consistency of data over its entire life cycle starting from generation to transmitting, storing, retrieving, and processing of data. Preservation of data integrity is mandatory in the realm of data science since maliciousness on data can lead to incorrect inference and muddle the decision-making process. Data scientists must be aware of integrity preservation tools and techniques while understanding their roles and efficiency in order to correctly implement the integrity requirements in data science applications.

DPSIA / Data Integrity	
Scope	Competencies
• The accuracy, consistency, and validity of data • Need for integrity requirements from security perspective • Techniques and mechanisms to ensure data integrity • Common security threats in data integrity	• Explain the differences of data integrity, data security, and data privacy • Describe the main strands of knowledge needed to address data integrity • Demonstrate the skills to apply commonly-used methods to ensure data integrity • Instill confidence in dealing with security threats affecting data integrity.
Sub-domains	
DPSIA/DI-Logical integrity – T1 DPSIA/DI-Physical integrity – T1 DPSIA/DI-Security threats affecting data integrity – T1	DPSIA/DI-Methods to ensure data integrity – T1 DPSIA/DI-Data corruption and data validation – T2

DPSIA/DI-Logical integrity – T1

Knowledge

- The concept of logical integrity
- Types of integrity constraints in database systems
- Entity integrity, referential integrity, domain integrity, user-defined integrity

Skills

- Explain concepts in logical integrity

Dispositions

- Confidence in explaining logical integrity

DPSIA/DIT-Physical integrity – T1

Knowledge

- The concept of physical integrity

- Physical and hardware methods to ensure data integrity such as RAID, redundant hardware, uninterruptible power supply, error-correcting memory, and sever cluster

Skills
- Explain concepts in physical integrity
- Describe physical and hardware methods to ensure physical integrity

Dispositions
- Confidence in addressing physical integrity through hardware methods

DPSIA/DI-Security threats affecting data integrity – T1

Knowledge
- Common data integrity threats including human errors, software errors, transmission errors, malware, insider threats, cyber attacks, and compromised hardware
- Data and information poisoning
- Data provenance assurance

Skills
- List common types of security threats affecting data integrity.
- Describe the potential vulnerabilities behind different hash functions, such as SHA-1 and MD5.

Dispositions
- Confidence in describing common security threats.

DPSIA/DI-Methods to ensure data integrity – T1

Knowledge
- Role of hash algorithms in integrity preservation
- Role of Message Authentication Codes (MACs) and its variants
- CRC and checksum for achieving integrity
- Mechanism behind digital signature schemes (RSA and ECDSA)

Skills
- Explain how to use hash algorithms and MAC mechanisms to ensure data integrity.
- Describe digital signature schemes and their needs in integrity preservation context.
- Compare and contrast different integrity preservation techniques in terms of performance and security.
- Understand how to use the integrity models in multiple data ownership domain to ensure provenance and maintain data validity.

Dispositions
- Confidence in addressing data integrity through various methods and techniques.

DPSIA/DI-Data corruption and data validation – T2

Knowledge

- The concept of data corruption
- The concept of data validation
- Methods to prevent data corruption including checksums and error correcting codes
- Validation methods including input validation, data type validation, range and constraint validation, and cross-reference validation

Skills

- Explain concepts in data corruption and data validation.
- Describe methods to prevent data corruption and ensure data validation.

Analysis for Security (DPSIA/AS):

This knowledge unit focuses on data science analytical techniques including statistics, probability, machine learning, and data mining, with a specific focus on security and privacy problems. This unit allows deeper understanding of data science tools, algorithms and techniques for security and privacy.

<table>
<tr><th colspan="2">DPSIA / Data Analysis for Security</th></tr>
<tr><td>Scope</td><td>Competencies</td></tr>
<tr><td>● Understand security data telemetry and different security applications
● Statistical analysis for security telemetry data
● Machine learning for security telemetry data
● Explainable machine learning methods for security-critical applications
● Machine learning vulnerability and robustness</td><td>● Categorize different security-critical applications and understand various security telemetry data.
● In-depth knowledge and strong hands-on implementation skills in Machine Learning (ML) and statistical methods for security applications.
● Elementary knowledge in Foundations of probabilities.
● Strong knowledge in ML explainability and resiliency and knowing when to apply them.</td></tr>
<tr><td colspan="2">Sub-domains</td></tr>
<tr><td>DPSIA/AS-Machine learning (ML) algorithms and statistical methods for security – T1</td><td>DPSIA/AS-Machine learning (ML) robustness and explainability – T1
DPSIA/AS-Categories of security applications – T2</td></tr>
</table>

DPSIA/AS-Machine learning (ML) algorithms and statistical methods for security – T1

Knowledge

- Statistical methods for exploratory data analysis on security data including descriptive statistics, summary plots, outlier detection, point estimation, hypothesis testing, test statistics, linear regression, and generalized linear regression.
- Computer vision based approaches such as malware-as-an-image technique, transfer learning, hierarchical ensemble neural network (HeNet) built on hardware for both static and dynamic threat classification and malware detection.

Skills

- Implement ML models and translate security applications into problems that can use ML.
- Design malware detection solutions by employing malware-as-an-image, transfer learning and hierarchical ensemble neural network (HeNet) for static and dynamic detection mechanism.
- Explain decisions made by ML models for security applications to audience with different backgrounds.

Dispositions

- Understand different perspectives from computer vision, natural language processing and classical data analysis to approach threat detection, malware intelligence and exploit identification problems

DPSIA/AS-Machine learning (ML) robustness and explainability – T1

Knowledge

- Basic concept of adversarial machine learning, types of attacks against ML models, and protection frameworks for ML.
- Adversarial machine learning techniques such as fast gradient sign, iterative fast gradient, universal adversarial perturbation
- Defense techniques such as adversarial training to better protect ML models
- Explainable machine learning methods for security applications. Explanations include local explanation, which is per-sample based, and global explanation, which considers the dataset as a whole. Know how to employ model-agnostic explanations on natural images to vision-based malware detection mechanism.

Skills

- Evaluate ML resiliency in terms of identifying its blind spot and bypassing its detection.
- Improve ML resiliency by conducting adversarial training.
- Conduct explainability studies on ML algorithms and explain the reasons for ML model selection to security experts.
- Communicate with various stakeholders to define ML metrics to address interpretability and vulnerability.
- Understand that ML resiliency and vulnerability is a key metric for ML used in security and privacy applications.

- Apply explainable AI methods such as LIME, LEMNA, TCAV to ML models built for security applications and conduct model selection based on the trustworthy scores. Especially when using malware-as-an-image approach, be efficient at applying LIME for malware classification model interpretability.

Dispositions

- View ML beyond classification accuracy, false positive, precision, and recall metrics. When training ML, always consider robustness and vulnerability.

DPSIA/AS-Categories of security applications – T2

Knowledge

- Security-critical applications: network analysis, malware intelligence, malware triage, dynamic malware analysis, hardware telemetry analysis.
- Types of security telemetry data: dynamic logs, binary, static code, dynamic code.

Skills

- Recommend data collection step for design of experiments before building ML models.
- Know which ML methods to use based on the nature of security telemetry data.

Dispositions

- Be very knowledgeable about types of data that can be generated from various security applications and make the optimal use of the datasets as a skilled data scientist.

Machine Learning (ML)

Machine learning, sometimes known as Statistical Learning, refers to a broad set of algorithms for identifying patterns in data to build models that might then be productionized and possibly productized. These methods are critical for data science. Data scientists should understand the algorithms they apply and make principled decisions about their use.

<table>
<tr><td>Scope</td><td>Competencies</td></tr>
<tr><td>Broad categories of machine learning approaches (e.g., supervised and unsupervised).
Algorithms and tools (i.e., implementations of those algorithms) for machine learning.
Machine Learning as a set of principled algorithms (e.g., optimization algorithms), rather than as a “bag of tricks.”
Challenges (e.g., overfitting) and techniques for approaching those challenges.
Performance metrics.
Training and testing methodology.
Algorithmic and data bias, integrity of data, and professional responsibility for fielding learned models.</td><td>Appreciate the breadth and utility of machine learning methods, compare and contrast them, select appropriate (classes of) methods for specific problems.
Apply machine learning algorithms following appropriate training and testing methodology.
Exhibit knowledge of methods to mitigate the effects of overfitting and curse of dimensionality in the context of machine learning algorithms.
Provide an appropriate performance metric for evaluating machine learning algorithms/tools for a given problem.
Be aware of problems related to algorithmic and data bias, as well as privacy and integrity of data.
Consider and evaluate the possible effects -- both positive and negative -- of decisions arising from machine learning conclusions.</td></tr>
<tr><td colspan="2">Sub-domains</td></tr>
<tr><td>ML-General – T1, T2, E
ML-Supervised Learning – T1, T2, E
ML-Unsupervised Learning – T1, T2, E
ML-Mixed Methods – E
ML-Deep Learning – T1, T2, E</td><td>Note that Reinforcement Learning appears in AI-Knowledge Representation and Reasoning (Probability-based Models)</td></tr>
</table>

ML-General

Given the centrality of machine learning algorithms to many data science tasks, data scientists should be aware of a wide range of machine learning approaches, as well as the long history of work in this area. A data scientist should be aware of where to look for possible techniques to apply to new problems.

A data scientist should also be aware of cross-cutting concepts, such as the need to evaluate performance and general classes of challenges faced in machine learning.

Knowledge
T1:

- History of machine learning
- Major tasks of machine learning, including supervised, unsupervised, reinforcement, and deep learning
- Difference between symbolic versus numerical learning, statistical versus structural/syntactic approaches
- Learning algorithms as principled optimization approaches
- "Doing machine learning" as one method of data mining. "Doing machine learning" as a process.
- Importance of robust evaluation
- Challenges for machine learning, including quality of data, need for regularization

Skills
T1:

- Compare the goals, inputs, and outputs of supervised, unsupervised, reinforcement, and deep learning.
- Know that different types of data-driven questions can be answered by different approaches and be able to connect them appropriately.
- Explain at a high level that ML models and algorithms are principled techniques based on mathematical and statistical foundations.
- Trace the process of "doing machine learning" as a method of data mining: understanding the question / problem a client cares to solve, gathering the data relevant to solving that problem, converting raw data into features, selecting appropriate machine learning methods, tuning those methods, evaluating performance (often against a baseline), and presenting results and insights.
- Discuss the trade-off between fitting to training data and generalizing to new data and how model complexity, as well as the number of examples and features, affect this trade-off. Relate this to the role and setting of hyperparameters.
- Appreciate trade-offs across performance, interpretability, scalability. Recognize that different optimization functions and techniques may yield different trade-offs in this space.

T2:

- Follow the derivation of a simple optimization function and learning algorithm from first principles, e.g. decision trees using information theory, logistic regression using

maximum likelihood and stochastic gradient descent, PCA using variance minimization and eigenvalues.
- Analyze performance across models using bootstrapping and statistical significance testing.
- Understand how to efficiently transition a model into production and choose tools that support that transition from the onset.
- Understand which tools to use based on the size of the data -- for Big Data, it is essential to choose a machine learning tool that can run parallelized, otherwise, the learning process may take much longer than is acceptable.
- Be aware of the state-of-the-art machine learning tools available.

Elective:
- Describe the process of automated (or meta-) learning, specifically how to automate the machine learning pipeline, including data pre-processing, model selection, model structure search, and hyperparameter tuning.

Dispositions

T1:
- Appreciate that, though recently made popular, machine learning is not a recent innovation. Look for existing solutions before presuming a new invention is required.
- Appreciate that machine learning is not an ad-hoc set of "tricks" and that it should be used responsibly.
- Understand that "doing machine learning" is not, in the general case, a simple process of applying a machine learning program to a conveniently-formatted data set. It is a process toward a goal for a client.
- Appreciate that there are several dimensions along which learned models may be compared, ranging from empirical loss minimization to model size and complexity to human interpretability.
- Understand the responsibility to present results as fair and honest comparisons considering all aspects of model comparison (quality, efficiency, interpretability, etc.).

ML-Supervised Learning

One major class of learning approaches can be described as "supervised" and includes techniques for both classification and regression. A data scientist should be aware of these types of algorithms, including challenges and methodologies that are unique to this type of learning. Note the relationship of this sub-domain with DM-Classification and Regression.

Knowledge

T1:
- Major tasks of supervised learning: regression and classification
- Use cases of regression and classification
- Important considerations and tradeoffs in supervised learning, including the relationship between model complexity and generality; the trade-off between bias and variance; Occam's razor as motivation for simple models.

- The need for separation of training, test, and validation data. Define training error and testing error.
- Common evaluation metrics for classification tasks (e.g., accuracy, sensitivity, specificity, precision, recall, F1, AUROC, regret) and regression tasks (e.g., root mean squared error, mean absolute error, R^2)
- The need for validation data. Cross-validation procedure and its goals: tuning hyperparameters and measuring model performance.
- Criteria for assessing the quality of training, test, and validation data, such as number of examples, class stratification.
- Classification and regression algorithms, including at least one linear and one non-linear algorithm for each. (e.g., linear regression/classification, logistic regression, nearest neighbor, Naive Bayes, decision tree learning algorithms).
- Common extensions to basic algorithms, including polynomial features and ensembles (e.g., bagged models, boosted models, random forests).

T2:

- Approaches for determining whether a model has high bias or high variance, e.g. training vs test performance, learning curves.
- Reasons to augment or reduce feature set; at least two approaches for each and trade-offs.
- How supervised classifier-learning models can be applied to multiclass problems, including how binary classification models can be extended to multiclass tasks.
- How to express performance using macro- and micro- metrics.
- At least one advanced supervised learning algorithm (e.g. SVMs with kernels, neural networks).

Elective:

- Derivation of supervised learning algorithms from first principles.

Skills

T1:

- Express performance of a classifier model using a confusion matrix.
- Compare strengths and weaknesses of evaluation metrics for classification tasks and regression tasks.
- Compare the trade-offs of at least two applied classification algorithms; compare the tradeoffs of at least two regression algorithms.
- Apply at least two classification and two regression algorithms to small and medium data sets.
- Compare training and testing error in terms of what they tell us about learned models.
- Compare the performance of the algorithms using various metrics.
- Apply at least two extensions (e.g., ensemble methods) to small, medium, and large data sets; compare the performance of the algorithms using various metrics.
- Justify when extensions such as polynomial features and ensembles are appropriate based on the problems each is able to address.

T2:

- Apply at least two classification and two regression algorithms to a large dataset.
- Apply at least one extension to a large dataset.
- Apply methods to mitigate high bias or high variance.
- Apply feature augmentation and selection to a medium or large sized problem.

- Apply advanced supervised learning algorithm (e.g. SVMs with kernels, neural networks).

Elective:

- Derive a simple optimization function and learning algorithm from first principles, e.g. logistic regression using maximum likelihood and stochastic gradient descent. Extend these techniques to similar models.

Dispositions

T1:

- Appreciate the importance of algorithm choice and evaluation metric on the quality of a learned model. Know that these choices have implications for and must be made with important stakeholders -- i.e., those for whom models are being developed.
- Appreciate the importance of applying principled evaluation approaches for models in which we can have high confidence.

ML - Unsupervised Learning

A major class of machine learning approaches can be described as "unsupervised" and include techniques for clustering and dimensionality reduction. A data scientist should be aware of these types of algorithms, including challenges and methodologies that are unique to this type of learning.

Note the relationship of this sub-domain with DM-Cluster Analysis.

Knowledge

T1:

- Major tasks of unsupervised learning, including clustering and dimensionality reduction.
- Use cases for both tasks (e.g., data exploration/summarization/visualization, feature selection, data compression, data denoising, prototype learning, recommender systems, topic modeling).
- At least one simple clustering algorithm, e.g. k-means or hierarchical clustering.
- Trade-offs of connectivity-based vs centroid-based clustering.
- At least one simple dimensionality reduction algorithm, e.g. principal component analysis (PCA).
- Similarities and differences between feature transformation, feature selection, and feature projection.

T2:

- At least one advanced clustering algorithm, e.g. density-based methods such as Gaussian mixture models (GMMs).
- At least one advanced dimensionality reduction algorithm, e.g. independent component analysis (ICA) or non-negative matrix factorization (NMF).

Elective:

- At least one mathematical method for implementing algorithms efficiently, e.g. matrix factorization and singular value decomposition (SVD) vs eigendecomposition for PCA.
- At least one advanced algorithm, e.g. spectral clustering, kernel k-means, kernel PCA, latent Dirichlet allocation (LDA).

- The connection of PCA to autoencoders; generalization to non-linear dimensionality reduction.
- Derivation of unsupervised learning algorithms from first principles.

Skills
T1:
- Apply at least one clustering and one dimensionality reduction algorithm to small, medium, and large data sets.
- Express the performance of an unsupervised learning algorithm using various metrics (e.g., visualization; comparison to ground truth, if available; computing metrics such as cluster density; indirect metrics via utility towards another application).
- Explain and apply methods to describe how to choose hyperparameters, e.g. the number of clusters for k-means or the number of components for PCA.

T2:
- Compare the trade-offs of at least two clustering algorithms.
- Compare the tradeoffs of at least two dimensionality reduction algorithms.

Elective:
- Apply advanced unsupervised algorithms.
- Derive a simple optimization function and learning algorithm from first principles, e.g. PCA using variance minimization and eigenvalues. Extend these techniques to similar models.

Dispositions
T1:
- Appreciate the importance of algorithm choice and evaluation metric on the quality of a learned model. Know that these choices have implications for and must be made with important stakeholders -- i.e., those for whom models are being developed. [See ML - Supervised Learning]
- Appreciate the importance of applying principled evaluation approaches for models in which we can have high confidence. [See ML - Supervised Learning]

T2:
- Appreciate that unsupervised learning offers useful techniques for data exploration, understanding, summarization, and visualization.
- Appreciate that unsupervised learning can be a useful preprocessing step to improve the quality or efficiency of supervised learning algorithms.

ML-Applications that Require Mixed Methods – E

Some learning problems and domains have special structure that can be leveraged by specialized techniques. A data scientist should be aware of these broad classes of applications and should know where to turn for possible methods to approach them.
Note the relationship of this sub-domain with DM-Time Series Data.

Knowledge

- Examples of learning problems and domains in which the structure of data or interrelatedness of data points may be leveraged in the learned model. For example, time series prediction, sequence prediction, recommender systems.
- How time dependencies or assumptions of shared information across data points may be leveraged in learning.
- Shortcomings of using a supervised or unsupervised approach instead of a mixed approach, e.g. problems of model interpretability or performance.

T2:

- For one such problem, at least one standard approach for learning, e.g., Hidden Markov Models (HMMs) for sequence prediction or Collaborative Filtering for recommender systems.
- The need for separation of training and test data in this context.
- Common evaluation metrics for the selected task, e.g., recall, precision, F1 score for recommender systems.
- Criteria for assessing the quality of training, test, and validation data for the selected problem.

Skills

- Map one such problem to a framework for learning. I.e., map data to inputs and outputs, consider settings of hyperparameters, run an appropriate learning algorithm.

Dispositions

- Recognize that challenges (e.g., time inhomogeneity, data sparsity) present in ML models generally may be more salient in these contexts.

ML-Deep Learning

The availability of data, as well as the availability of computational processing power have led to new and powerful techniques for large-scale learning. A data scientist should be aware of these types of algorithms, including challenges and methodologies that are unique to this type of learning.

Knowledge

T2:

- How multilayer neural networks (including non-deep networks) learn and encode higher-level features from input features.
- Common deep learning architectures, such as deep feedforward networks, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and LSTMs; purpose and properties of each.
- Practical challenges of common deep learning approaches, e.g., choosing a deep learning architecture, having sufficient data / possibility of overfitting, length of learning time, interpretability.
- Examples of regularization methods for deep learning architectures, such as early stopping, parameter sharing, and dropout.

- Examples of methods for mitigating other challenges of deep learning, such as tools that work with GPUs or on distributed systems.
- Selection of appropriate tools that scale with the size of the data -- specifically, processing Big Data calls for Deep Learning tools that run in a parallelized way.
- Be aware of the state-of-the-art deep learning tools available.
- At least one commonly used algorithm for learning in the context of deep networks, e.g., how backpropagation is used in a deep feedforward network or how backpropagation is used to learn higher-order features in a convolutional network; how backpropagation through time is used in recurrent networks.
- The operation of convolution and why it may be useful, e.g., detecting vertical edges in an image.
- Pooling; examples of pooling functions such as max pooling and use cases.
- Challenge of long- vs short-term dependencies in recurrent neural networks; at least one solution, such as LSTMs.

Elective:

- Deep generative models, such as generative adversarial neural networks (GANNs) and applications for which they may be used.
- Practical challenges of such approaches, e.g., convergence, mode collapse, etc.
- Approaches for handling or mitigating the effects of the above.

Skills

T2:

- For a given data set and task, determine the type of deep learning approach(es) that would be most appropriate to apply.
- Use a deep learning toolkit (e.g., PyTorch, Tensorflow) to apply a learned model to a dataset.
- Use a deep learning toolkit (e.g., PyTorch, Tensorflow) to learn a model for a dataset, including configuring a network.

Elective:

- Use a deep learning toolkit to apply (from scratch) a generative approach for a specific goal.
- Configure a toolkit to work for a given system architecture.

Dispositions

T1:

- Understand the potential negative implications of using a machine-learned model that is difficult or impossible to interpret or explain.
- Appreciate that there are many problems for which the power of deep learning is more than what is necessary.
- Understand and appreciate social and political concerns around deepfakes.

Professionalism (PR)

In their technical activities, data scientists should behave in a responsible manner that brings credit to the profession. One aspect of this is being positive and proactive in seeking to bring benefit, to undertake positive developments and doing so in a way that is responsible and ethical. Much of this is amplified in general terms in [1]. The section below serves to highlight relevant issues of specific concern to the data scientist.

Scope	Competencies
• The meaning of competency and being able to demonstrate competency • The acquisition of competencies particularly relevant to the data scientist • Acquiring expertise / mastery or extending competency; the role of journals, conferences, courses, webinars • Technological change and its impact on competency • The role of professional societies in CPD and professional activity	• Recognise and be comfortable with the range of knowledge that underpins a professional approach to data science • Demonstrate the range of skills that underpins a current and ongoing professional approach to data science • Acquire a set of dispositions that underpin a confident, effective and professional approach to all aspects of data science as well as the wherewithal to maintain such an approach
Sub-domains	
PR-Continuing Professional Development – T1 PR-Communication – T1 PR-Teamwork – T1 PR-Economic Considerations – T2	PR-Privacy and Confidentiality – T1 PR-Ethical Considerations – T1 PR-Legal Considerations – T2 PR-Intellectual Property – E PR-On Automation – E

PR-Continuing Professional Development – T1

The essence of a professional is being competent in certain aspects of data science. It is the responsibility of the professionals to undertake only tasks for which they are competent. There are then implications for keeping up-to-date in a manner that is demonstrable to stakeholders (e.g. employers).

Knowledge

- The meaning of competency and being able to demonstrate competency

- Acquiring expertise / mastery or extending competency; the role of journals, conferences, courses, webinars
- Technological change and its impact on competency
- The role of professional societies in CPD and professional activity

Skills

- Justify the importance to professional data scientists of maintaining competence.
- Describe the steps that professionals would typically take to extend competence or acquire mastery, explaining the advantages of the latter.
- Argue the importance of the role of professional societies in supporting career development.

Dispositions

- Recognise that data science is a rapidly changing field where keeping current, as well as knowing how to stay current, are vital.

PR-Communication – T1

There are various contexts in which the data scientist is required to undertake communication with very diverse audiences. That communication may be oral, written or electronic. There is often the need to engage in discussion about the role that data science can play, to communicate multiple aspects of the data science process with colleagues, to convey results that may lead to change or may provide new insights. Being able to articulate the need for change and being sensitive to the consequences of change are important professional attributes. These activities may entail the ability to have a discussion about limitations in certain contexts and to suggest research activities.

Communication from the data scientist must be underpinned by an evidence-based approach to decision making. There is special significance to this in the context of machine learning and automation where the reasons for decisions may require clarification.

An important consequence of developments in machine learning is the ability of machines to understand natural language (and so voice input), which can then be employed in such contexts as robotics, word processors or intelligence driven search engines (e.g. Siri, Cortona, Google Assistant, Alexa).

Knowledge

- Different forms of communication – written, oral, electronic - and their effective use
- The technical literature relevant to data science
- Audiences relevant for communication involving the data scientist – including small groups, large groups, experts and non experts, younger groups, senior managers, machines – and the elements of effective communication in each case

Skills

- Evaluate aspects of the technical literature relevant to data science

- Produce a technical document for colleagues to guide technical development
- Produce presentations for a range of audiences who have an interest in aspects of data science
- Design and present situations to senior managers outlining significant initiatives stemming from a data science investigation including as necessary general issues associated with change management

Dispositions

- Keep current with relevant changing technology, know how to do so effectively and be alert to opportunities for new developments
- Reflect positively on the significance of new learning and new experiences
- Recognise one's strengths and weaknesses regarding knowledge

PR-Teamwork **– T1**

The data scientist will often become a member of a team. This may entail being a team leader, or supporting the work of a team (which may be sensitive). It is important to understand the nature of the different team roles as well as the typical dynamics of teams. In terms of teamwork, the data scientist often needs to be able to collaborate not only with data scientists with different tool sets but, in general, with a diverse group of problem solvers.

Knowledge

- Team selection, the need to complement abilities and skills of team members
- The dynamics of teams and team discipline
- Elements of effective team operation

Skills

- Outline steps that could be taken to deal with conflict situations within teams.
- Document and justify the considerations involved is selecting a team to undertake a specific data science investigation.
- Recognise the qualities desirable in the team leader for a data science research investigation.

Dispositions

- Be perceptive to sensitivities regarding the formation and operation of teams.
- Set aside unimportant differences when working with others.
- Demonstrate appropriate levels of flexibility.

PR-Economic Considerations – T2

Data scientists should justify their own positions as well as the kind of activity in which they engage.

Knowledge

- The cost and value of high quality data sets, and of their maintenance
- Justification in cost regarding data science activities
- Estimation of project costs
- Promotion of data science
- Automation stemming from data science activity

Skills

- Predict the value of data sets for organizations, taking into account any requirement for maintenance.
- Argue the case for the data that an organization should routinely gather; design a related data collection process identifying the attributes to be included and the form the collection should take having an eye to quality.
- Document the cost (in terms of resources generally) of collecting high quality data for a particular purpose.
- Justify the creation of data science activities within an organization and quantify the cost.
- Infer the value to an organization of undertaking a particular investigation or research project.
- Document and quantify the resources needed to carry out in-house investigations and compare that with outsourcing such activities.
- Evaluate and justify the costs associated with the automation of a particular activity.

Dispositions

- Adopt a responsible attitude to costs associated with data science activities.

PR-Privacy and confidentiality – T1

It is possible to gain access to data in a multitude of ways, by accessing databases, using surveys or questionnaires, taking account of conditions of access to some resource, and even with developments such as the Internet of Things, specialized sensors, video capture and surveillance systems. Although gaining access to all kinds of information is important, professionals must do this legally and in such a way that the information is accurate and it protects the rights of individuals, as well as organizations and other groups, are protected.

Note the relationship of this sub-domain and the Knowledge Area on Data Privacy.

Knowledge

- Freedom of information
- Data protection regulations including General Data Protection Regulation (GDPR) regulation – see [5]
- Privacy legislation
- Ways of maintaining the confidentiality of data
- Threats to privacy and confidentiality
- The international dimension

Skills

- Describe technical mechanisms for maintaining the confidentiality of data.
- Compare the privacy legislation from different countries, highlighting problems arising from any differences.
- Recognize the privacy and confidentiality issues arising from the use of video, voice and face recognition software.
- Having an awareness of the contexts in which particular privacy legislation should be applied, having an eye to international standards.

Dispositions

- Include and maintain privacy and confidentiality to ensure confidence in data science activities.

Contextual issues

- The legal framework associated with privacy and security can vary from one country to another.

PR-Ethical Considerations – T1

Ethical issues are of vital importance for all involved in computing and information activities as captured extensively in [1]. Underpinning these activities is a view that professionals should undertake only tasks for which they are competent, and even then should carry out such tasks in a way that reflects good practice in its many forms. Maintaining or extending competence is essential. A heightened awareness of legal and ethical issues must underpin the work of the data scientist. Professionals should consider the ethical issues associated with their decisions as a very important starting point that enables them to recognize themselves as "independent, ethical agents."

Knowledge

- Ethical issues associated with competence and the maintenance of that competence
- Confidentiality issues associated with data and its use
- General Data Protection Regulation (GDPR) regulation – see [5]
- Need for data, including samples of data, to be truly representative of a situation
- Awareness of, and the possible nature of, bias in data and in algorithms; mechanisms for checking and avoiding bias
- Algorithmic transparency and accountability

Skills

- Illustrate a range of situations in which a data scientist may venture beyond their range of competence and identify steps to mitigate such situations.
- Demonstrate techniques for establishing lack of bias in data sets or in algorithms.
- Reflect on the merits of joining a network of professionals in the data science area.

Dispositions

- Be alert to the deep ethical issues associated with gathering data and its use.
- Be aware of issues of bias and seek to remove these.

- Strive to be self-directed and self motivated in the advancement of data science.

PR-Legal considerations – T2

Computer crime has continued to increase both in volume and its severity over recent years. In many cases criminals have brought disruption, even chaos, to many organizations. Their threat cannot be ignored and professionals must take steps to counter the possibility of severe disruption. In many cases the law has adjusted to counter these trends but this is an ongoing area of continuous change and adjustment.

Knowledge

- Computer crime relevant to data science
- Cyber security
- Crime prevention
- Mechanisms for detecting criminal activity, including the need for diverse approaches
- Recovery mechanisms and maintaining 100% operation
- Laws to counter computer crime

Skills

- Illustrate and evaluate a range of mechanisms for detecting a stated form of criminal activity.
- Justify the desirability of having multiple diverse approaches to countering threats.

Dispositions

- Adopt a responsible but sensitive and caring attitude when confronted with possible criminal situations.

Contextual issues

- The legal framework can vary from one country to another.

PR-Intellectual property – E

Intellectual Property Rights (IPRs) such as copyright, patents, designs, trademarks and moral rights, exist to protect the creators or owners of creations of the human mind. Moral rights include the right to be named as a creator of intellectual property (IP), and the right to avoid derogatory treatment of creations. For the data scientist the items requiring protection, in possibly different ways, include software, designs including graphical user interfaces (GUIs), data sets, moral rights and reputation. Trade secrets may also be relevant.

Knowledge

- Patents, copyrights, trademarks, trade secrets, moral rights and trademarks
- What data science related IP can and cannot be protected, and what kinds of protection are available

- Types of data science related IPs that can and cannot have legal protection and which kind of protection is available
- Regulation related to IP, IP ownership, the territorial nature of IP rights including the effects of international agreements (e.g. the European Directive on trade secrets) and the issue of IP rights being time limited
- Kinds of IP rights that vest automatically and which require registration, including overview of the processes involved in acquiring registered IP rights
- Possibility of infringing the rights of others and validly utilizing protected IP

Skills

- Describe those kinds of IP that are relevant to data scientists.
- Argue the difference between patents, copyrights, designs and trademarks and illustrate their use in the context of data science.
- Describe the role of trade secrets in relation to data science.
- Illustrate the processes involved in registering IP rights.
- Describe and explain the issues relating to IP ownership and moral rights.
- Evaluate the risks involved in using protected IP and ways to overcome them validly.

Dispositions

- Sensitivity to the existence and importance of, as well as responsibilities and opportunities afforded by intellectual property.

Contextual issues

- Ethical and legal frameworks associated with intellectual property will vary from one country to another. Patent attorneys can typically advise.

PR-On Automation – E

Automation often creates concerns about loss of employment and, in general terms, about machines behaving unreasonably. Professionals should seek explanations about machine behaviour. Related issues are the subject of [3] and [6]. Automation can occur in critical situations where serious loss may be possible, and then typically there is an expectation that machines will operate according to a code of ethics that is in harmony with human behaviour.

Knowledge

- Automation, its benefits and its justification
- The particular concerns of automation in critical situations
- Transparency and accountability in algorithms

Skills

- Articulate to a non-technical audience the extent to which automated decision making occurs in a particular situation.
- Analyze the impact on a design requirement to provide insights into decisions made autonomously by machines.
- Argue the benefits of automation for different situations.

- Identify steps needed to ensure that a decision-making system is auditable.

Dispositions

- Sensitivity to issues of automation and its effect on employment.
- Adopting an open and highly responsible approach to issues of automation.

References

[1] The ACM Code of Ethics and Professional Conduct, published by ACM on 17th July 2018. See acm.org

[2] When computers decide: European Recommendations on Machine-Learned Automated Decision Making, published by ACM, 2018. See europe.acm.org

[3] ACM US Public Policy Council and ACM Europe Policy Council, "Statement on Algorithmic Transparency and Accountability," 2017.

[4] Directive (EU) 2016/943 on protection of undisclosed know-how business information (trade secrets) against their unlawful acquisition, use and disclosure. See eur-lex.europa.eu June 2016.

[5] The EU General Data Protection Regulation, see www.eugdpr.org. Approved by EU on 14th April 2016 with an implementation date of 25th May 2018.

[6] Simson Garfinkel, Jeanna Mathews, Stuart S. Shapiro, Jonathan M. Smith, Towards Algorithmic Transparency and Accountability, Communications of the ACM, September 2017,vol. 60, no. 9, page 5.

Programming, Data Structures, and Algorithms (PDA)

Data scientists should be able to implement and understand algorithms for data collection and analysis, as well as integrate them with existing software and/or tools. They should understand the time and space considerations of algorithms, as well as particular issues around numerical computing.

Note that this knowledge area draws from various CS2013 knowledge areas but does not duplicate them: Algorithms and Complexity (AL), Computational Science (CN), Programming Languages (PL), and Software Development Fundamentals (SDF).

<table>
<tr><td>Scope</td><td>Competencies</td></tr>
<tr><td>

Problem solving through algorithmic thinking.
Development and implementation of programs, including integration with various existing software and/or tools.
Use of traditional programming languages to integrate existing interfaces between datasets and applications.
Use of a programming language designed for statistical computing in the context of a data science problem.
Knowledge and use of Abstract Data Types (ADTs)
Knowledge and use of numerical computing algorithms
Algorithm design and analysis
Factors that influence algorithmic complexity and performance
Complexity analysis and comparison

</td><td>

Design an algorithm in a programming language to solve a small or medium size problem.
Write clear and correct code in a programming language that includes primitive data types, references, variables, expressions, assignments, I/O, control structures, functions, and recursion.
Implement good documentation practices in programming.
Use techniques of decomposition to modularize a program.
Use standard libraries for a given programming language.
Write appropriate database queries.
Select appropriate data structures for a given problem.
Select appropriate algorithms for a given problem.
Appreciate the importance of time and space complexity on the practical utility of an algorithm.

</td></tr>
<tr><td colspan="2">Sub-domains</td></tr>
<tr><td>PDA-Algorithmic Thinking & Problem Solving – T1, T2
PDA-Programming – T1, T2, E
PDA-Data Structures – T1, T2, E</td><td>PDA-Algorithms – T1, T2, E
PDA-Basic Complexity Analysis – T1, T2
PDA-Numerical Computing – T1, T2</td></tr>
</table>

PDA-Algorithmic Thinking & Problem Solving

In order to develop correct, efficient, clear, and usable code -- either in the process of data analysis and presentation or for production-level systems -- a data scientist should have fundamental algorithmic problem solving skills.

Knowledge
T1:

- Definition of an algorithm
- Importance of algorithms in the problem-solving process
- At least one formal technique for approaching problem solving
- Fundamental object-oriented design concepts and principles
    - Abstraction
    - Encapsulation and information hiding
    - Separation of behavior and implementation

Skills
T1:

- Express a problem solution using a formalism other than code (e.g., flowcharts or pseudocode).
- Express the flow of data (input, transformations, output) through a problem solution in some formalism (e.g., a data flow diagram).
- Identify the inputs (e.g., data, hyperparameters, user responses) and outputs essential to implementing a program to solve a problem
- Identify the data components and behaviors of multiple abstract data types (See PDA-Data Structures).

T2:

- Use at least one formal technique for approaching problem solving.

Dispositions
T1:

- Appreciate that algorithms are different from programs.
- Understand that there are principled approaches for breaking large problems into implementable solutions and expressing those solutions in some formalism.

PDA-Programming

In order to collect, analyze, and present data, a data scientist needs to develop programming skills and should be well-versed in fundamental programming constructs. Because the data scientist will interface with many systems, they should be able to develop programs that can either stand alone or integrate with existing software and/or tools.

Knowledge
T1:

- Core coding concepts
    - Variables and primitive data types

    - Expressions and assignments
    - Conditional and iterative control structures
    - Recursive functions
    - Functions and parameter passing
    - Simple I/O, including files or other static data sources
    - Exceptions
- Core practices
    - Documentation
    - Testing
    - Version Control
- Decomposition to break a program into smaller pieces
- Types of errors (syntax, logic, runtime), how they might occur, and how they can be handled
- Methods for querying and parsing data sources

T2:

- Regular refactoring and program maintenance
- Variety of strategies for testing and debugging
- Utility of APIs; when to look for one

Elective

- Advanced concepts
    - in-line/anonymous functions (e.g., Lambda functions in Python)
    - variable argument lists for functions and programs
    - classes and objects

Skills

T1:

- Read, write and debug programs that include core concepts and practices listed above.
- Trace the execution of code segments and articulate summaries of their computation.
- Apply techniques of decomposition to break a program into smaller pieces.
- Collect and parse data from selected sources (e.g., databases, spreadsheets, text documents, XML) utilizing selected appropriate techniques (e.g., database queries, API calls, regular expressions).
- Effectively design and implement program solutions using recursion and iteration.
- Use consistent documentation and program style standards that contribute to the readability and maintainability of software.
- Apply strategies for testing and debugging programs.

T2:

- Describe the need for regular refactoring and program maintenance.
- Refactor, maintain, and improve programs following core practices.
- Effectively design and implement program solutions using classes and objects.
- Construct, execute, and debug programs using a modern IDE and associated tools such as unit testing tools and visual debuggers.
- Construct and debug programs using standard libraries available with a programming language.
- Integrate and use typical Application Program Interfaces (APIs).

Elective:

- Effectively design and implement program solutions using templates and generic functions.
- Design and implement unique Application Program Interfaces (APIs).
- Collect and parse data using specialized techniques (e.g. for natural language processing, image processing, etc.). (x-ref KA: Data Acquisition, Management, and Governance)
- Read, understand, write and debug programs that include advanced concepts.

Dispositions
T1:
- Appreciate the importance of software engineering concepts and design principles on the practice of programming. (x-ref KA: Software Development and Maintenance.)
- Be comfortable going beyond what has been directly taught. Appreciate that programming constructs and methods are general and useful in many contexts. A data scientist should not be bound by tweaking existing solutions.

PDA-Data Structures

In order to write effective and efficient code, a data scientist should know a variety of data structures, be able to use them, and understand the implications of choosing one over another. Given their role in many data science applications, particular attention is given to matrix representations and operations here.

Knowledge
T1:
- Basic data structures and Abstract Data Types (ADTs) (lists, arrays, stacks, queues, strings, sets, records/structs, maps, hash tables)
    - purpose
    - usage
- Basic matrix representation structures (sparse/dense, row, column)
    - matrix representation types
    - pros/cons of basic matrix operations based on representation types

T2:
- Advanced structures (trees, graphs)
    - purpose
    - usage

Elective:
- Matrix operation optimization

Skills
T1:
- Select basic data structures appropriately in programming
- Appropriately use standard data type libraries for a given programming language

T2:
- Select advanced data types appropriately in programming
- Appropriately use standard libraries for a given programming language

Elective:

- Implement a coherent abstract data type, with loose coupling between components and behaviors
- Compare/contrast the time/space of standard operations (e.g., find, insert, delete) for various data structures

Dispositions

T1:

- Appreciate that implementation and data structure choice have an impact on usage, efficiency (time and space), and readability.

PDA-Algorithms

A data scientist should recognize that the choice of algorithm will have an impact on the time and space required for a problem. A data scientist should be familiar with a range of algorithmic techniques in order to select the appropriate one in a given situation.

Knowledge

T1:

- Simple numerical algorithms, such as computing the average of a list of numbers, finding the min, max, or mode in a list
- Sorting and Searching
    - Sequential and binary search
    - $O(n^2)$ (e.g., Insertion) versus O(n log n) (e.g., Merge) sorts.
    - Randomized algorithms for searching and sorting (e.g., Quicksort)
    - Potential efficiency benefits of hash-based searching and sorting
- Properties of graphs: connectedness, betweenness, centrality, etc.
- Graph algorithms
- Basic algorithmic strategies, such as greedy, divide-and-conquer
- Algorithms for solving linear systems

T2:

- Algorithms for combinatorial optimization problems
- Heuristic optimization techniques

Elective:

- Hashing and hash functions

Skills

T1:

- Apply simple numerical algorithms (e.g., computing the average, finding the min, etc.).
- Apply searching and sorting algorithms.
- Contrast the tradeoffs of various array-based searching and sorting algorithms.
- Perform a graph or tree traversal using the general framework of a breadth or depth first algorithm.
- Identify a shortest path in a graph or tree using an efficient algorithm, such as a greedy algorithm.
- Apply linear system solvers to appropriate problems.

T2:

- Identify a max- or min-flow through a graph or tree using an efficient algorithm.
- Use common algorithms for combinatorial optimization problems (e.g., Branch and Bound algorithms)
- Apply heuristic optimization techniques (Particle swarm, genetic algs, evolutionary) to appropriate problems.
- Implement Dynamic Programming solutions for appropriate problems.

Elective:

- Implement or use search/sort algorithms on distributed systems or data
- Implement and/or compare hashing functions
- Graphs
    - Implement traversal, shortest path, and flow algorithms
- Analyze randomized algorithms

Dispositions

T1:

- Be aware that there are often a variety of algorithmic techniques that can successfully address a problem.
- Recognize that the choice of algorithm has significant implications for efficiency.
- Appreciate the implications of efficiency (time, space, etc.) for all code stake-holders such as clients, consumers, and maintainers.

PDA-Basic Complexity Analysis

Data scientists should be aware of the time and space required to solve a problem and should know that certain problems may not be solvable in a reasonable amount of time. They should also take into consideration how the platform on which they may be running their code will schedule their tasks.

Knowledge

T1:

- Definitions of time and space complexity
- Differences among best, expected, and worst case behaviors of an algorithm
- Trade-offs in managing time and space complexity
- Taxonomies for analyzing algorithms, such as
    - Deterministic vs. Non-Deterministic
    - Time/Space hierarchies

Skills

T1:

- Perform informal comparison of algorithm efficiency (e.g., operation counts).
- Run algorithms on input of various sizes and compare performance.
- Provide examples demonstrating that implementation and algorithm choice has an effect execution time or space.
- Explain how problem representations / data structures and algorithms are related/coupled.

T2:

- Formally apply a variety of classification taxonomies to understand algorithms.

Dispositions

T1:

- Understand that there may be trade-offs in managing time and space complexity and appreciate the implications of those tradeoffs for clients/users of software.

PDA-Numerical Computing

The types of problems data scientists solve often involve numerical computing. Data scientists should be aware of the power and limits of numerical representations. They should also be aware of standard numerical computing algorithms and their uses.

Knowledge

T1:

- Random Number Generators (RNGs)
- Simulation of probability distributions
- Limitations of numerical representations with bits, and their impact on the accumulation of error (overflow, underflow, round off, truncation) in results
- Implications of numerical representations with respect to their computational complexities

T2:

- Algorithmic and mathematical methods involved in advanced numerical algorithms for data analysis, such as:
    - Principal Component Analysis (PCA)
    - Singular Value Decomposition (SVD)
    - Eigenvalue decompositions
    - Newton's Method
    - Monte Carlo Simulation
- Connection between good problem representations and mathematical models for solving numerical problems. For example:
    - The use of SVD in representing documents
    - The representation of graphs as adjacency lists or sparse matrices
    - The use of kd-trees to represent metric spaces

Skills

T1:

- Describe how numerical computing algorithms and processes affect the execution of simulations, data sampling, and data generation.
- Describe appropriate numerical computing algorithms to perform data analysis with a recognition of their limitations and numerically driven constraints.
- Effectively use random number generators and simulated probability distributions to
    - Allow reproducibility in data analysis with non-deterministic algorithms

    - Introduce non-determinism into algorithms to ensure proper statistical and numerical conditions

T2:

- Identify and apply appropriate numerical algorithms for solving a variety of problems. Algorithms may include (non-exhaustive, non-ordered):
    - Principal Component Analysis (PCA)
    - Singular Value Decomposition (SVD)
    - Eigenvalue decompositions
    - Newton's Method
    - Monte Carlo Simulation

Dispositions

T1:

- Appreciate the benefits and limitations of (pseudo)-random number generation
- Appreciate the limitations of numerical computing algorithms

Software Development and Maintenance (SDM)

Data scientists may be expected to build (or contribute to building) deployable systems either for the purposes of data analytics or to put into practice the results of data analytics. To this end, they should be familiar with fundamental software development principles and practices.

Note that this knowledge area draws from the CS2013 knowledge area on Software Engineering (SE).

Note that development and testing are addressed separately below. Testing is integral to the development process. They are separated below only for purposes of readability.

Scope	Competencies
• Software engineering principles, including design, implementation and testing of programs. • Potential vulnerabilities	• Implement a small software project that uses a defined coding standard. • Test code by including security, unit testing, system testing, integration testing, and interface usability.
Sub-domains	
SDM-Software Design and Development – T1, T2, E	SDM-Software Testing – T1, T2, E

SDM-Software Design and Development

A data scientist should understand design principles and their implications for issues such as modularization, reusability, and security. Design, implementation, and testing are tightly integrated components of software development. In this KA, we itemize design and testing competencies separately for the sake of readability.

Knowledge
T1:
- Coding and Design Standards
- Integration with Information Management/Database Systems
- Software lifecycle
- Data lifecycle

T2:
- Project management methodology

Elective:
- Integration with Embedded, Process Control, and/or Communications systems

Skills
T1:
- Explain project Coding Standards

- Explain project Design Standards
- Describe how to integrate or interact with Information Management/Database Systems
- Define and explain the scope and types of different testing paradigms/needs for all areas. [x-ref Testing below]
- Individually complete a small software project that meets design specifications
- Complete a team software project that meets design specifications
- Follow given design, documentation, and implementation standards
- Execute basic Software Lifecycle on a simple program
- Execute basic Data (Science) Lifecycle on a simple data product
- Integrate or interact with Information Management/Database Systems

T2:

- Follow a given project management methodology
- Plan and design a team software project that meets stakeholder specifications
- Lead a project to completion, meeting stakeholder requirements
- Implement data science lifecycle to build data-driven decisions in appropriate stages of the software lifecycle

Elective:

- Integrate or interact with Embedded, Process Control, and/or Communications systems

Dispositions

T1:

- Recognize the value of a team built on respect, diversity, and collaboration
- Recognize the value of adhering to project Coding and Design Standards
- Work well as a member of a team by demonstrating good listening skills, the ability to present an idea, and the ability to negotiate
- Approach data and software projects with a lifecycle mindset
- Recognize and value the benefits of using test-driven development [x-ref Testing below]

T2:

- Lead a project to completion following principles of respect, good listening, responsibility, etc.
- Promote and encourage adherence to project Coding and Design Standards

SDM-Software Testing

A data scientist should understand the importance of good testing in software development and deployment.

Knowledge

T1:

- Testing paradigms/needs for
    - Unit/Execution
    - Integration
    - Interface/User
    - Regression/Continuous
    - System
    - Security

T2:

- Potential security problems in programs
    - Buffer and other types of overflows
    - Race conditions
    - Improper initialization, including choice of privileges
    - Not checking input
    - Assuming success and correctness
    - Not validating assumptions

Skills

T1:

- Define and explain the scope and types of different testing paradigms/needs for all areas.
- Design and implement basic tests for:
    - Unit/Execution
    - Integration

T2:

- Use or extract representative data from Big Data datasets in order to test algorithms on a small scale before running at scale on a cluster, for example.
- Develop specifications and execute tests (built by others) for:
    - Interface/User
    - Regression Testing
    - System
    - Security
- Evaluate the results of a program using statistical significance testing
- Describe possible types of risks for a software system
- Describe secure coding and defensive coding practices

Elective:

- Design, Develop, and Execute tests for all areas

Dispositions

T1:

- Recognize and value the benefits of using test-driven development
- Approach basic software and data project development from a test-driven perspective, particularly as it pertains to unit/execution and integration tests

T2:

- Approach software and data project development from a test-driven perspective, particularly as it pertains to Security, Interface/User, Regression/Continuous, and System tests

Elective:

- Approach software and data projects holistically from a test-driven development perspective

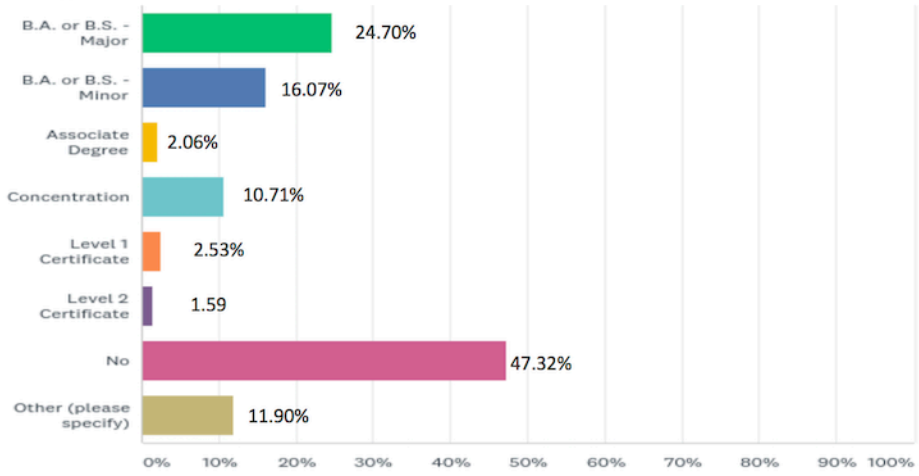
Appendix B: A Summary of Survey Responses

Here we include a subset of responses to the Academic and Industry surveys.

B.1 Academic Survey

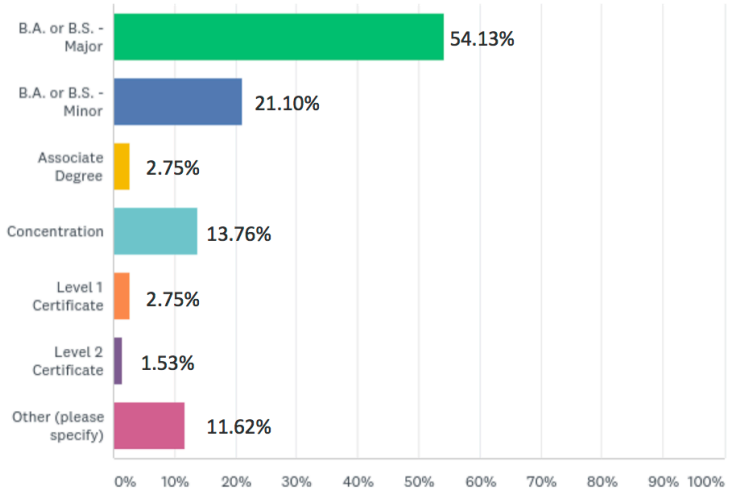
Q1: Does your institution offer an undergraduate program in Data Science or Analytics? (Check all that apply.)

Answered: 672 Skipped: 0



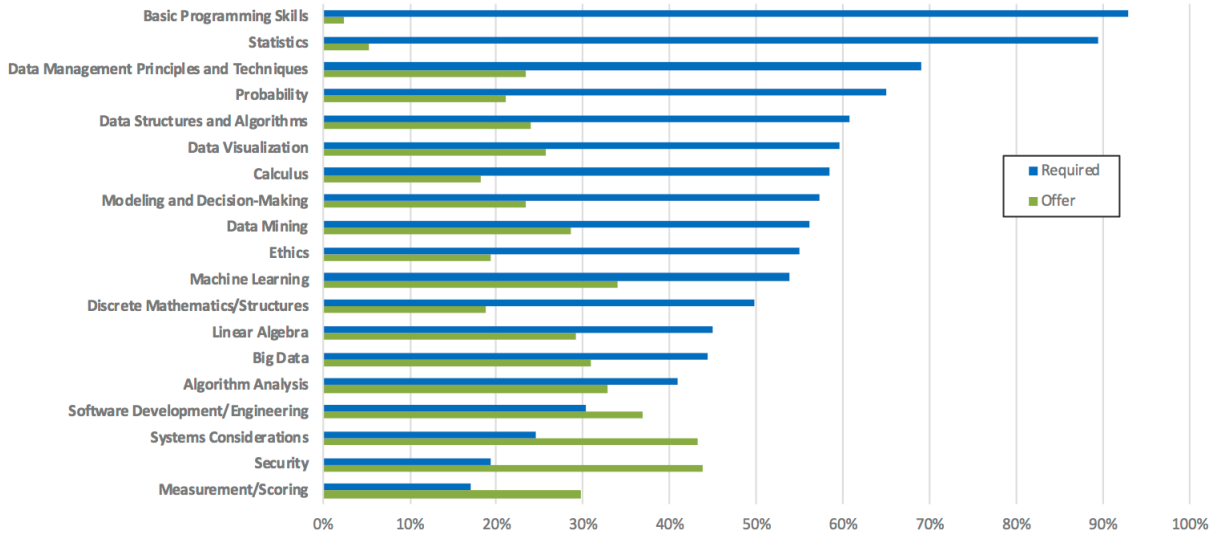
Q2: Please select one program for which you will answer a set of curricular questions.

Answered: 327 Skipped: 345



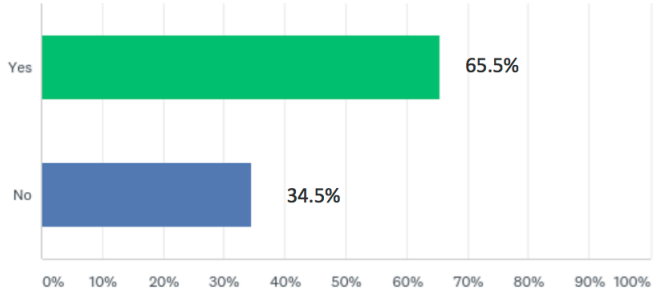
Q4: Does your program offer/require content from these areas?

Answered: 172 Skipped: 500



Q6: Does your program have a "data science in context" requirement? (i.e., a requirement to apply data science methodology to some area of application)

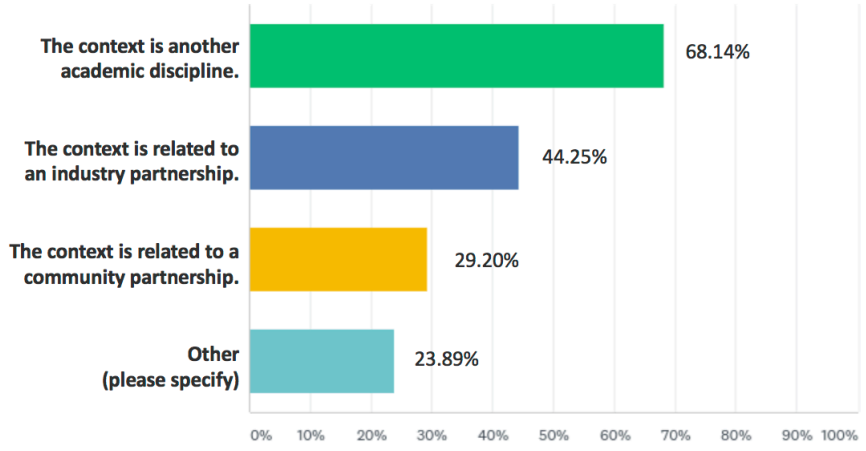
Answered: 171 Skipped: 501



Q7: Check all that apply

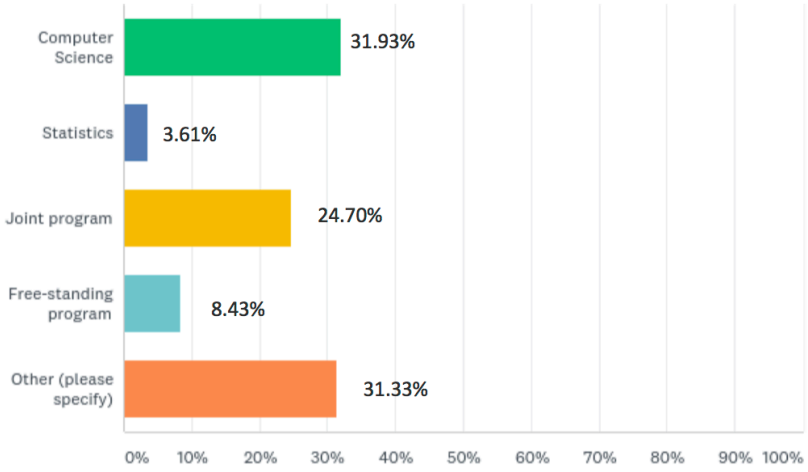
For: Does your program have a "data science in context" requirement?

Answered: 113 Skipped: 559



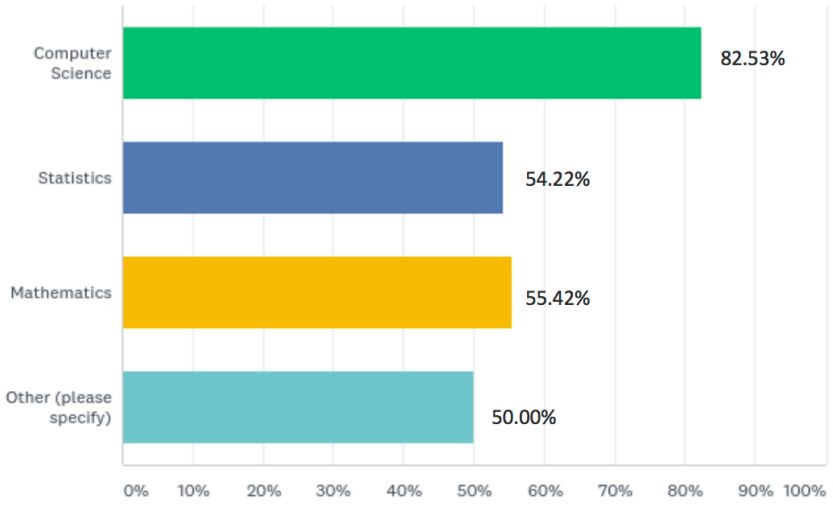
Q8: What is the academic home of your Data Science/Analytics program?

Answered: 166 Skipped: 506



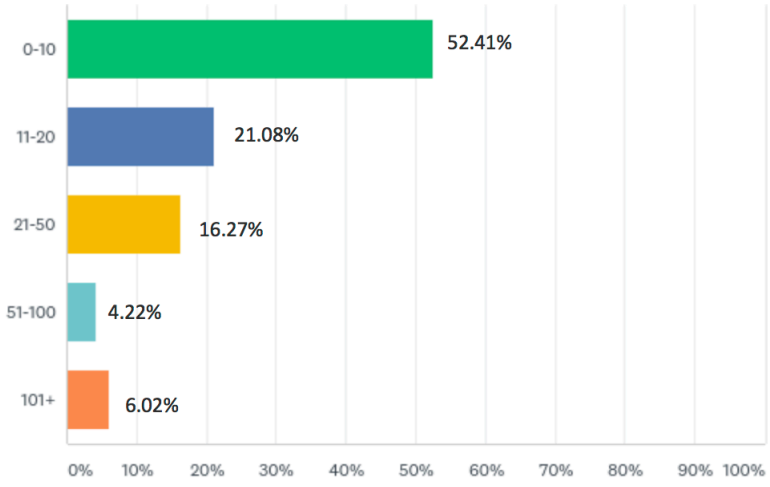
Q9: What academic units/departments contribute to your Data Science/Analytics program?(Check all that apply.)

Answered: 166 Skipped: 506



Q10: How many students graduate with this degree annually?

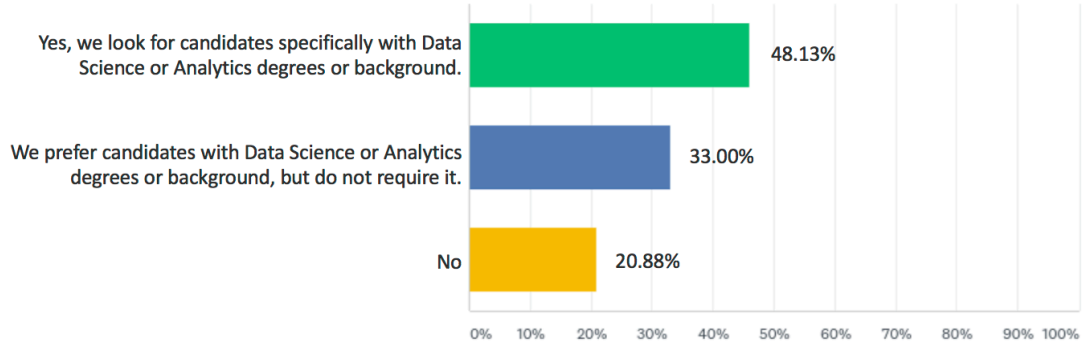
Answered: 166 Skipped: 506



B.2 Industry Survey

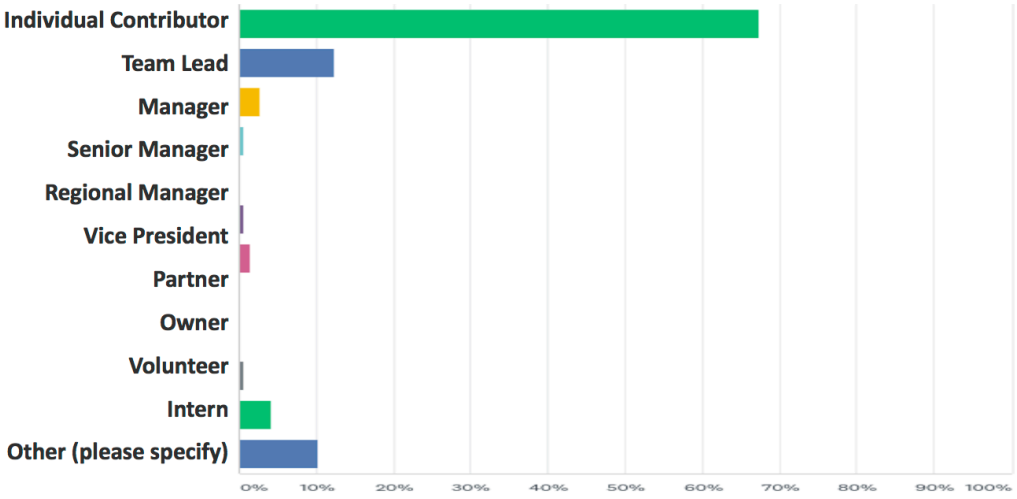
Q1: Do you look for job candidates (specifically new graduates out of undergraduate programs) with Data Science background?

Answered: 297 Skipped: 0



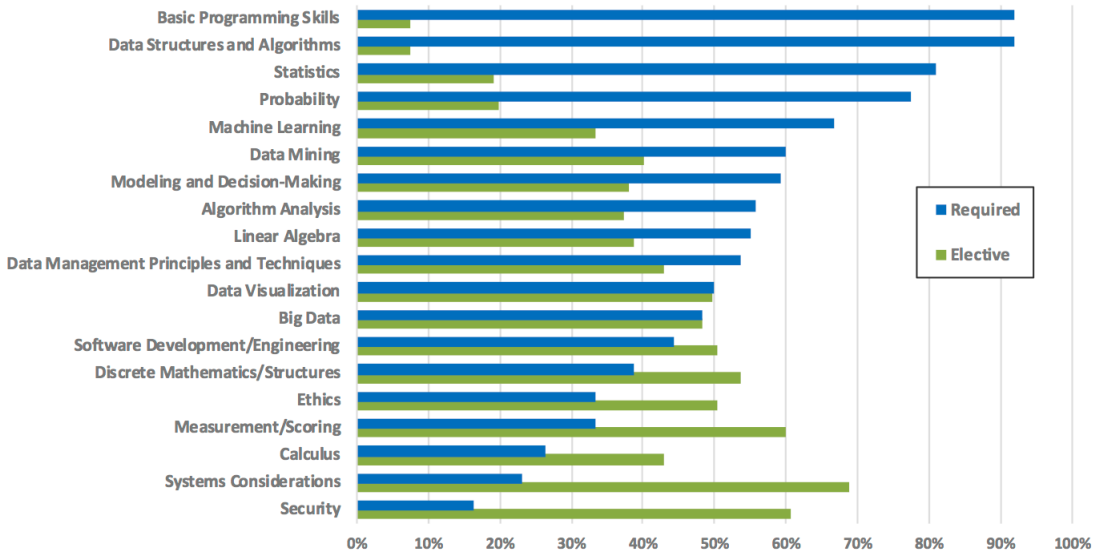
Q2: Provide the title of the job for which you require/prefer Data Science/Analytics degrees or background. The next few questions on the survey will pertain to that position.

Answered: 147 Skipped: 150



Q3: For the job title you provided, to what extent do you require experience in the following areas?

Answered: 147 Skipped: 150



Q4: In the list of COMPUTING-based data science areas above, what did we miss?
Open-Ended Response

AI, knowledge representation, text analytics, machine learning in perception, database (including NoSQL)
Analytical redundancy Binary standard - ramification
Basic applied domain understanding
Causal inference
COLAS
Data Driven Control Systems
Data analysis and interpretation; decision support for executives.
Data integration
Data warehouse management principles.
Design Thinking: the ability to find insights that matter into huge datasets
Graph theory
Heavy hands-on experiences. Exposure to very difficult problems.
Implementation. Life cycle management of the data science/machine learning algorithms.
Ontology engineering
Pattern recognition, Knowledge generation, General Intelligence, data gathering
Python SAS R
Simulation
SQL coding
Strong communication skills.
Study design and Interpretation of findings (These two are related).
Textual analytics
Too many data science education programs focus on mathematics, but not enough on actual computer programming.

Q5: Do you expect a job candidate to have experience applying data science in context? (i.e., experience applying data science methodology to some area of application)

Answered: 143 Skipped: 154