We are IntechOpen, the world’s leading publisher of Open Access books
Built by scientists, for scientists

Open access books available
International authors and editors
Downloads

Countries delivered to
TOP 1%
Contributors from top 500 universities

WEB OF SCIENCE™
Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com
Chapter

Best Practices in Accelerating the Data Science Process in Python

Deanne Larson

Abstract

The number of data science and big data projects is growing, and current software development approaches are challenged to support and contribute to the success and frequency of these projects. Much has been researched on how data science algorithm is used and the benefits of big data, but very little has been written about what best practices can be leveraged to accelerate and effectively deliver data science and big data projects. Big data characteristics such as volume, variety, velocity, and veracity complicate these projects. The proliferation of open-source technologies available to data scientists can also complicate the landscape. With the increase in data science and big data projects, organizations are struggling to deliver successfully. This paper addresses the data science and big data project process, the gaps in the process, best practices, and how these best practices are being applied in Python, one of the common data science open-source programming languages.

Keywords: Python, big data, data science process, best practices, open source

1. Introduction

Organizations use insights derived from data to stay competitive. The big data phenomenon has made deriving insights more challenging due to the changing characteristics of the data landscape. The increased volume, variety, and velocity of big data challenge traditional information technology (IT) processes to scale and support big data analytics and data science. Big data is used by organizations as a resource in data science projects to develop new business value and insights. Big data examples include sensor data, images, text, audio, and video data. These data sources can provide new insight opportunities alone and when paired with existing data sources such as organizational data warehouses.

Data science is a competency that leverages data processing, algorithms, and math to develop insights from data. Data science is a core competency that organizations want to develop to stay competitive. While data science as a competency is growing, the practices to ensure these projects are successful have not kept up with the pace. One primary challenge is using existing software development methodologies to deliver data science projects. Applying traditional software methodologies, such as the waterfall approach, is problematic and has been identified as the one contributing factor for data science project failure; organizations are treating data science projects like other IT projects [1].

According to Saltz, current research in data science and big data has primarily focused on the use and application of algorithms and generating insights, but little to no research has occurred about tools, methodologies, or frameworks used
to deliver data science projects [2]. Saltz outlined that existing tools, methodologies, and frameworks are not mature enough to effectively use in data science and big data projects [2]. This paper focuses on the changing data landscape, the data science process, and identifying best practices to accelerate the data science processing using Python. Python is an interpreter, object-oriented programming language and one of the most popular tools used in big data and data science projects.

2. The changing data landscape

The increased use of internet-connected smart devices has changed how organizations use information [3–5]. The Internet of Things (IoT) creates large amounts of data quickly from sensors embedded in devices, one of the contributing factors in creating the category of big data [5]. The rise of data science is primarily a result of big data, due to the need to analyze data, other than traditional structured data, such as text, machine-generated, and geospatial data [5]. Big data and data science go hand in hand; thus software development approaches used need to consider both [1, 4, 6]. Results of a data science project not only include insight, but also working software that needs to be deployed and supported. Analyzing the characteristics of big data highlights the challenges with traditional software development approaches.

2.1 Volume

The growth of data impacts the scope of data used in data science and software development. Scope increases project complexity where new technology is used to accommodate more data [3]. Large amounts of unstructured data cannot be easily ingested and processed using a traditional relational database for example.

2.2 Variety

Data variety becomes a concern for software development as the types of data sources to be used for development and analysis increase. The variety of data means increasingly complex forms of data such as structured and unstructured data [3–5]. Traditionally, structured data is created in rows and columns and easily understood; however, unstructured data comes in different forms, levels of details, and without clear metadata complicating the ability to understand and use [3–5].

Examples of data variety include images, IoT sensor data, clickstream, images, and event data. These data sources may be analyzed independently, but often analysis requires data to be integrated. Integrating multiple data sources with different structures increases the complexity of projects.

2.3 Velocity

The speed at which data is created is referred to as velocity. In 2014, Twitter averaged 1 billion tweets every few days [7]. Fresher data results in the ability to analyze new patterns and trends that were not possible before big data. With IoT applications, data that is 15 minutes may be too old for analysis [5]. Data acquisition becomes a challenge as traditional data acquisition focused on extract, transformation, and load (ETL) of data. Increased velocity changes the order where data is loaded first, then analyzed, otherwise known as extract, load, and then transformation (ELT) [3–5].
2.4 Veracity

Veracity refers to how accurate data is and how well the data is understood. Big data is not clearly analyzed prior to ingestion due to the volume and speed of creation, which results in data that may have credibility and reliability problems. Often metadata for big data sources do not exist. These challenges increase the complexity of deriving insights from big data sources [3–5].

Software development lifecycles have traditionally focused on requirements which drove the design, leveraging the design to develop software, testing, and then deploying the software. Projects using big data change the order in which these phases occur. Big data sources are ingested, stored, and explored first, and then requirements are determined which changes the traditional order of activities for project delivery. Using the traditional software development lifecycle for projects that include big data has failed further supporting that projects using big data need to adjust project approaches [1].

According to Mayer-Schönberger and Cukier, big data changes how the world interacts and means a disruption to what was considered normal. This disruption also means disruption to the software development processes that create the software that derives knowledge and value from data. Big data results in changes to IT processes, technologies, and people. One greater reliance observed is that data scientists need to address the complexity introduced with big data and help derive the knowledge from data [3, 4].

3. The data science process

According to Saltz, most data science processes focus on the tasks that need to be completed in data science such as the techniques to acquire and analyze data [2]. Saltz analyzed different data science approaches and found that most outlined the steps as data acquisition, cleansing, transformation, integration, modeling, analysis, and deployment [2]. The data science approaches are task-oriented, and no real evolution of the process had occurred since the cross-industry standard process for data mining (CRISP-DM) was introduced in the 1990s [2].

3.1 CRISP-DM

The most commonly used data mining process is CRISP-DM which is a process that conceptually described the stages used in data mining. Originally created to support data mining projects, it has been adapted by data scientists. There are six stages in the CRISP-DM process which are presented sequentially, but iteration is expected:

- Business understanding
- Data understanding
- Data preparation
- Modeling
- Evaluation
- Deployment
3.2 Business understanding

The start of the process, business understanding, focuses on the business value of the project. Once requirements and objectives are understood, the problem definition is created. Data science projects start with a problem to be addressed or a question to be explored. Tools to support identifying the problem include diagrams such as a decision model such as a fishbone diagram [8].

3.3 Data understanding

Once the problem to be addressed is identified, the next focus is on data source identification and collection. Data source identification includes identifying the sources, such as a transactional processing system, and the attributes needed to address the problem. Problems may involve several different data sources, which means data integration work is likely. After integration, data is profiled and statistically analyzed to determine quality, demographics, relationships between variables, and distribution. The outcome of data understanding is a definition of how the data can be used. The data understanding stage is often referred to as exploratory data analysis (EDA) [8].

3.4 Data preparation

The goal of data preparation is to create the data that is to be used in the modeling stage. Input from data understanding is used to determine the final set of attributes, often referred to as features that will be used in the model. Preparation includes integration, cleansing, and deriving of new attributes. Data preparation is iterative as the training and testing of the model may require new or changed data. The result of data preparation is the data set to be used as input into the modeling stage [8].

3.5 Modeling

As part of the modeling stage, different techniques and algorithms are used to determine the best model. Modeling goes through cycles of testing and training, where the data scientist adjusts parameters to produce the best outcomes. It may be necessary to return to data understanding and preparation stages if the model performance is not acceptable [8].

3.6 Evaluation

Model evaluation is determined based on the overall fit to the problem statement and business objectives. Evaluation is conducted by analyzing error rates, variance, and bias of the model. Often models using different techniques are compared to determine the best performing one. Once the best performing model is identified, a formal review is conducted to move to model deployment [8].

3.7 Deployment

The deployment stage is often overlooked and unplanned for but important as this is where business value is realized. Deployment focuses on a working software model that can be supported and executed on a regular frequency. Once deployed, models are monitored for performance as model accuracy will degrade because of organizational change and time. Models are often retrained once this occurs [8].
The description provided of CRISP-DM is a summary and does not highlight the granularity of effort needed for successful data science outcomes. Many organizations use CRISP-DM as a framework and add steps to each stage for software development teams to follow. There was an effort to create CRISP-DM 2.0 in 2007, but there is no new research or activity in this area [2].

4. The role of open source in data science projects

Landset et al. outlined that big data has caused IT departments to rethink how data is processed and the use of data science software [9]. Choosing the right processing framework and data science software can be challenging due to data science project requirements and that data complexity might require more than a single solution [9]. Additionally, tools available to conduct data science are many with partial functionality, limited ability to handle big data, and integrate into big data processing platforms, which contributes to the fragmentation and complexity of the data science and big data technology solutions.

Landset et al. called out that no single tool or framework covers all of the requirements of a data science project [9]. Data scientists and computer engineers need tools that support computing performance, usability, machine learning algorithm breadth, and portability. To support these needs, data scientists and computer engineers have turned to open-source tools to address the variety of requirements of data science projects. Open-source technology categories include data processing platforms and engines and machine learning tools. The Hadoop ecosystem is commonly used to address the data processing aspect of big data and data science [9]. The Hadoop ecosystem includes workflow, data collection, storage, and processing, for example. While there are many open-source machine learning tools, Python has become one of the leading programming languages used in data science as well as R and Mahout [9].

5. Challenges with big data and data science projects

To best understand what the best practices are in big data and data science projects, it is beneficial to highlight some of the challenges. Some of the challenges highlighted so far include lack of a detailed software development methodology and the characteristics of big data. Other challenges include that many data science projects are managed as a single project and the focus on reproducibility, collaboration, and communication is overlooked [10].

Andrejevic argued that big data changes how businesses and customers interact which disrupts normal business processes [11]. This disruption also means disruption to the software development processes used to create the data science models that derive data insights [11]. When organizations leverage big data, this results in changes to IT processes, technologies, and people [11]. One change observed is that data scientists are challenged to handle all activities related to the data science process including all the data wrangling activities which can consume most of the project timeline [3, 11]. Traditional IT projects normally have defined roles and activities involving several team members [3, 11].

Mayer-Schönberger and Cukier highlighted the impact of big data on organizations citing that the volume, variety, and velocity characteristics make data science and big data projects difficult to deliver using traditional IT project approaches [12]. Volume challenges how much data is ingested and consumed, variety highlights the need to support different data structures, and velocity outlines the need to ingest
data as soon as it is created [12]. The need for nontraditional (non-relational databases and storage systems) data processing platforms and software that can handle big data is the reason why traditional IT processes are unsuited for data science and big data projects [12].

Some of the challenges in delivering data science and big data projects include having clear business objectives, dealing with volume, identifying what data to use, understanding opportunities to store and process data, and having clear privacy and security requirements [13]. Mousannif et al. proposed a framework for a big data project workflow that included planning, implementation, and post-implementation phases [13]. Mousannif et al. highlighted a need to focus on reproducibility of data ingestion, processing, and storage to expedite future big data projects [13].

Lowndes et al. proposed that data science and big data projects can be accelerated by focusing on reproducibility, transparency, and collaboration by implementing new processes leveraging open-source tools [10]. Lowndes et al. published the results of implementing new processes to accelerate data science for the Ocean Health Index (OHI) project which is repeated yearly to address the change in global ocean health [10]. New processes were implemented in the categories of reproducibility, communication, and collaboration and broken down further into specific tasks [10].

Lowndes et al. outlined that the following areas need to be addressed for reproducibility: data preparation, modeling, version control, and organization [10]. Data preparation includes creating and leveraging common coding routines to sort, cleanse, transform, and format data [10]. Modeling focused on standardizing to a common programming language to ensure all were using the same algorithm methods which resulted in reduced iterations on validating results [10]. Version control ensured that the team was treating the data science process as a software development process where code was tracked and change management was put in place to improve software reusability [10]. Organization was addressed through leveraging in-code documentation standards, file naming standards, and treating each data science project as a single set of common code by implementing the project function in the programming language [10].

Lowndes et al. proposed that collaboration be improved by having centralized coding repositories, common workflows for promoting code, and a centralized repository for communication [10]. Git, a widely used cloud-based version control system, was used as the common coding repository, and a Wiki was used for documenting projects [10]. Having a common project management approach was also highlighted as a benefit [10].

Lowndes et al. focused on improved team communication and effectiveness through sharing data and methods [10]. The focus was on not redoing work if the work was already completed [10]. Data sharing covered centralizing cleansed data sets for reuse and creating common data pipelines to be used for data science projects (like the OHI project) [10]. Since the OHI project is completed yearly, much of the software development work could be leveraged from the prior year in place of starting the project from scratch each time [10].

There are several gaps and lack of maturity in the data science process as outlined by the research of Landset et al., Mayer-Schönberger and Cukier, Mousannif et al., and Saltz. Based on these gaps, best practices will be proposed to accelerate the data science process leveraging Python. While other open-source programming languages could be used in the same manner as suggested in the next section, the choice of Python is not meant to recommend that Python is the only choice or best choice to use in data science projects. Python was chosen due to its popularity, performance, and portability capabilities in the data science and big data space.
6. Best practices to accelerate data science with Python

People, process, and technology are contributing factors to data science complexity. Data scientists are expected to multiskilled resources knowing statistics, big data platforms, pipeline development, and deep learning neural networks. The primary process leveraged for data science is CRISP-DM which has not really changed since it was introduced in the 1990s. Lastly, there is so much available technology to use in data science it boggles the mind. While these problems will take time to solve, there are some best practices that can be leveraged to make data science less complex and more scalable in organizations. Best practices will be addressed in general as well as with Python.

6.1 Team collaboration

Data science initiatives tend to be done as independent efforts or one-off projects. Data scientists often work as the project manager, data wrangler, software developer, data engineer, tester, and do the data science as well. Data scientists cannot be effective assuming all these roles. With data wrangling (cleansing, transformation, formatting, etc.) taking up more than half the project, the data engineer has emerged as a key role in supporting the data science process. Assign a data engineer to handle the data wrangling, and let the data scientist focus on data science. Additionally, data science initiatives are projects. Ensure the data science initiative has a lead who can remove barriers, deal with stakeholders and get the required subject matter experts to support the data science project. Collaboration is key in making progress and valuable data science results [10, 14].

6.2 Why Python?

Python is an interpreter, object-oriented programming language introduced in 1991 and has emerged as one of the leading open-source data science tools used by data scientists [15]. Python has become a leading data science tools for several reasons. As an open-source tool, Python is freely available and modifiable, keeping costs low and promoting rich features through the open-source community [6].

Python is easy to learn. Although it is a programming language, the syntax is easy to adopt, especially by programmers, and through studies, the learning cycle tends to be shorter than a comparable data science tool—R. R is another open-source programming environment specializing in statistical computing and graphics. Both tools are comparable in many areas; however, Python has emerged as being more scalable and portable [15].

Scalability and portability are important in the data science community. With big data characteristics such as volume, velocity, and variety, data science tools need to be robust. Scalability refers to the ability to handle growing computing requirements, and portability is the ability to easily run on multiple computing platforms. Python has libraries and packages that support fast computing, and it runs on the common operation systems such as Linux, Windows, Unix, and macOS [15]. While Python is discussed as a data science tool, it is also a programming language used in the development of applications.

Team collaboration was mentioned as a best practice in data science; however, collaboration is inherent within open-source communities. Python has a deep reach in the data science community with data scientists contributing to creating new libraries and code routines. In addition, the tech companies often choose Python to release new functionality first. Google, for example, released its deep learning