Project Management for Data Science Activities

Introduction

Enterprises recognize the importance of data management and data science to a company, but it is often less clear to measure the ROI of establishing a solid data science team. Partly it is due to lack of understanding of the role and responsibility of a data science team, e.g. many perceive it as some sort of hybrid of data engineering, business intelligence and product management. As a result, many have incorrect assumptions about what the team does and believe that the same project management methodology that work for software engineering would also be applicable to data science. It is not entirely false, but the key to succeeding with data science is to understand the project management of data science activities. Data science requires its own lifecycle, e.g. understand data, choose projects, build models, manage deployment, maintain data pipeline, and refine model's post-deployment.

This discussion summarizes 5 popular project management approaches:

- CRISP-DM
- Waterfall
- Scrum
- Kanban
- TDSP

CRISP-DM

The Cross-industry standard process for data mining (CRISP-DM) was proposed in 1996 by representatives from SPSS, Teradata, Daimler, NCR, and OHRA. The purpose was to standardize a data mining process across different industries. Basically, CRISP-DM includes six major iterative phases, each has its own defined tasks and set of deliverables such as documentation and reports. It is widely considered to be a long-standing, traditional project management methodology that many data science teams adapt.

The six phases are,

1) Business Understanding: determine business objectives; assess situation
2) Data Understanding: collect initial data; describe data; explore and verify data quality
3) Data Preparation: clean data; construct data; integrate data; format data
4) Modeling: build model; assess model
5) Evaluation: evaluate and review process; determine next steps
6) Deployment: plan deployment; monitoring process; review project

Waterfall

Waterfall is a classic project management methodology, originated from manufacturing and construction and then was applied to software engineering projects starting in the 1960s. A Waterfall project management approach flows through defined phases such as 1) data/business requirement, 2) design, 3) execution, 4) evaluation, and 5) deployment. Some Waterfall models include variations of these phases that might include more comprehensive steps, e.g. conception, initiation, communication, planning, analysis, construction, development, testing, and deployment.

All Waterfall approaches start with an initial phase and then cascade sequentially in a forward linear pattern toward the final phase. That is precisely described by the name, i.e. Waterfall, from high to low in one sequential direction. Traditionally, revisiting prior phases in the project lifecycle is considered to contradict a true Waterfall approach. Many people would have seen it as poor planning or misstep. Gantt charts are often used to track project lifecycle using Waterfall approach. It is easy to visualize the dependency relationships between different objectives and current schedule status.

Scrum

Scrum divides larger projects into a series of mini projects. Each mini-project cycle, referred as a *sprint*, can last from 1 to 2 weeks to several months. It is noteworthy that each sprint has a fixed timeframe and deliverable.
Essentially, there are three major players in the project lifecycle,

1) Product Owner: set product vision, expectation and define potential product increments (also known as *product backlog*)
2) Development Team: professionals who deliver product increments, e.g. data scientists, data engineers, data analysts, systems analysts, software engineers, etc.
3) Scrum Master: manage the Scrum process as a servant leader

In each sprint planning, a product owner would define and explain the top feature priorities. The development team would follow up and propose what increments they can deliver by the end of the sprint. Subsequently, the team would come up a sprint plan to develop these increments. During the sprint, each member from the development team would coordinate closely and discuss plans or follow-ups at a daily standup. At the end of the sprint, the team demonstrates the increments to stakeholders and solicit feedback during sprint review. Scrum relies heavily on feedback exchanges between parties. In summary, this is a classic "divide and conquer" approach.

Kanban

A traditional Kanban methodology includes three set of project activities — '**To Do**', '**In Progress**', and '**Done**'. In the world of data science, additional columns can include 'In development', 'Coding', 'Testing', etc. Like the backlog concept of Scrum, Kanban starts with a list of potential features or tasks that are initially collected and placed in the 'To Do' column of a Kanban board, i.e. a visual representation of workflow. In a simple three columns Kanban board, a Kanban card (picture it as a sticky note) is moved from the 'To Do' to the 'In Progress' column when a task or activity being started. Once finished, it is moved to the 'Done' column. While CRISP-DM and Waterfall seen as traditional project management approaches, Scrum and Kanban are perceived more in

common as an agile management approach. However, Kanban puts more emphasis on work in progress and less on fixed dates and roles. This approach is more flexible and welcome by data scientists whose work cannot always be bounded by fixed schedule.

TDSP (Microsoft's Team Data Science Process)

In 2016, Microsoft proposed a Team Data Science Process (TDSP) and described it as "an agile, iterative data science methodology to deliver predictive analytics solutions and intelligent applications efficiently." Essentially, it is a hybrid model that puts Scrum and CRISP-DM together.

The TDSP's project lifecycle is highly like CRISP-DM and that includes five iterative stages:

1) Business Understanding: define objectives and identify data sources
2) Data Acquisition and Understanding: ingest data and determine if it can answer the presenting question
3) Modeling: feature engineering and model training
4) Deployment: deploy into production environment
5) Customer Acceptance: customer validation if system meets business needs

Each stage would include a list of specific objectives, outline of specific tasks, guidance on completion and clear set of deliverables. Most importantly, TDSP addresses the criticism of CRISP-DM's lack of team definition by proposing four distinct roles, i.e. **solution architect**, **project manager**, **data scientist**, and **project lead**, and clearly defining their responsibilities during each phase of the project lifecycle. In addition, Microsoft provides standardized project documents such as project charters and data reports, infrastructure and resources for data science projects, and tools and utilities for project execution. TDSP is arguably the most sophisticated CRISP-derived project management approach because of its focus and clear definition of team in addition to detailed documentation.

Conclusion

Data science project management must be highly flexible in order to work best with each organization in different scenarios. The author finds that agile project management style continues to be an effective framework that enables flexibility and productivity. Still, perhaps the best approach is to combine Scrum + CRISP-DM like process rather than a rigid, traditional Waterfall approach. Spring planning, daily standups, constant feedbacks and communication with different stakeholders are particularly useful and considered important requirements in many organizations. The Waterfall approach basically breaks down project activities into linear sequential phases. For data science projects, this linear dependency tends to become inflexible and work poorly because it does not respond well to spontaneous changes in business. It requires detailed and complete specifications upfront, whereas data scientists usually do not know in advance what data can tell us.

Scrum or agile project management approach is very popular in software development as it emphasizes on incremental delivery, team collaboration, and continual learning, instead of trying to deliver everything all at once near the end. It is highly customizable and adaptive to business challenges/changes. Data science initiatives are always project-oriented, so there is always a defined start and end, plus a series of step to follow. The CRISP-DM is an extensible methodology that offers an effective

framework, standard guidance for data science projects. Each phase of CRISP-DM e.g. goal definition, business understanding, data preparation, modeling, etc. can be columns or stages that align well within a Scrum project management tool, like JIRA ticketing system.

A hybrid data science project management approach can be highly flexible, and foster team collaboration. For example, a data science project may be planned as several agile sprints, each utilizing all CRISP-DM phases. The first sprint likely focuses on discovery, such as business objectives, data availability, data quality, etc. The next may be data preparation and make some baseline models for testing purposes. Subsequently, the following sprints can focus on finding the best model specification. The final sprint may hone and harden the deployment structure. Each sprint touches on every phase of CRISP-DM, but the central focus changes with each one. In this hybrid structure, the project remains time-boxed for each sprint. Each sprint is scheduled for constant review and there is a culture of team collaboration with relevant stakeholders. Most importantly, it adapts a "fail-fast" mentality, i.e. each sprint and development on the products is based on learning of the previous, and that leads to a final product, solution that is accepted and communicated well between stakeholders.

The TDSP process is a new, modern approach, i.e. a hybrid approach that is both agile and iterative. It develops and delivers requirements throughout the project lifecycle, such as focusing on regular collaboration with stakeholders to integrate their feedback into the design, embracing changing requirements, improving customer satisfaction throughout the process, etc. Perhaps it will become increasingly more popular and widely adapted in the data science world in near future.

Thoughts

1) What do you consider the most effective data science project management approach?
2) What is usually the bottleneck during a data science project lifecycle?
3) Is there any data science project management tool (such as Gantt chart, JIRA, Zendesk, Rapidminer, Salesforce, etc.) that you would like to recommend?

References

http://www.datascience-pm.com/

https://towardsdatascience.com/what-project-management-tools-to-use-for-data-science-projects-49c17c719cfe

https://www.dominodatalab.com/resources/field-guide/managing-data-science-projects/

https://dzone.com/articles/role-of-project-manager-in-data-science

https://docs.microsoft.com/en-us/azure/machine-learning/team-data-science-process/lifecycle