APPENDIX

A

Tools for Data Science

The Data Strategy Canvas

The data strategy canvas is a tool that is used in order to structure the implementation of a data science within a company. You will find this canvas especially useful if you are implementing data science for the first time within your company.

The canvas is covering the most important topics that you have read in this book:

1) Challenge—Here you have to specify the exact problem you are trying to solve. Don't forget to pose the problem as a question. This forces you to make the problem more concrete and will also help the data scientist better understand what it is that you are trying to do.
2) Data sources—Where are you getting your data from?
3) Appropriateness, nature, time, and cost—These are the four data considerations that were discussed in Chapter 2 about data management.

S. Kampakis, *The Decision Maker's Handbook to Data Science*,
https://doi.org/10.1007/978-1-4842-5494-3

4) Method—While a data scientist would be able to better advise you on this, it is still a useful exercise to try to think how you could solve this problem. It will at least help you make potentially better hiring decisions. However, do not get too stuck on your answer to this question. For example, it is likely that something that might look like a deep learning problem to you can be solved more easily with different techniques. Discuss with the data scientist the best approach, and let them have the final say.

5) Success criteria—When will you know that the project has finished? The wrong expectations can cause all sorts of problems. If you can't define clear success criteria, then at least think whether you can break down a project into smaller milestones that are more manageable.

6) People—Use the things you learned from the "Hiring and Managing Data Scientists" section in order to figure out which type of data scientist would be better suited to this problem. Also, think whether it is best to work with a contractor or a full-time employee. Many projects are one-off, so contractors can be a good choice.

7) Culture—Will a data scientist feel comfortable working with you, or will they feel like second-class citizens? You might spend a long time finding the right person, and then they might leave if the company does not provide the right cultural fit.

The data strategy canvas

Challenge
Data sources
Appropriateness (can the data be used to solve the problem?)
Nature (is the data noisy or other issues?)
Time (how fast can you acquire new data?)
Cost (how expensive is the data?)
Method (how will you solve the problem, machine learning, deep learning, statistical modeling, etc.?)
Success criteria
People (whom should you hire?)
Culture (do you have the right culture for data scientists to come and work for you?)

The Data Science Project Assessment Questionnaire

The purpose of this questionnaire is to help you clarify your objectives and factors that can affect the success of a data science project.

It is important to understand that a data science project is often an exercise in risk management. The problematic part in data science is the "science" bit. Quite often, it is impossible to know how well something will work in advance, until you try things out. The only cases where you can be confident of results before you embark on a project are

1) You have worked on the same case with very similar data in the past.
2) There is an extensive body of literature on this type of problem.

In any other case, the results cannot be guaranteed. Hence, the right project structure should focus on organizing an "attack plan," where each step should produce one of either two outcomes:

1) Successful completion of the project.
2) Learn something about the problem (e.g., maybe a family of methods is not appropriate, which can lead to improved exploration)

In order to better manage the project, expectations, and risk, the following questionnaire can help you list all the important requirements. This questionnaire is both for you and the data scientist, and it will help you build a common understanding of the requirements, the challenges, and how to mitigate risks. It is recommended that you do at least one round of back-and-forth between you and the data scientist in order to make sure that there is appropriate understanding on both sides.

The Data Science Project Assessment Questionnaire

Success criteria
What would you consider successful implementation of the project? Share your own thoughts.

Are there any benchmarks in performance? Please try providing a numerical answer (e.g., "anything above 60% accuracy is good, based on a benchmark X").

Risk factors
Is the data of appropriate quality? If no, what are the issues?

Is the data of appropriate size? If no, how much do you think this will affect performance? How will you mitigate that risk?

If the data is of high quality and size, then what could possibly prevent a good model from being built (e.g., maybe the domain is particularly difficult)?

Timelines
How time critical is the project?

If the project is difficult and no approach can reach the desired accuracy, which of the following plans seem the most attractive to you and why?

- *Keep on trying more advanced approaches (e.g., ensemble modeling).*
- *Fix data issues and try again (e.g., collect more data).*
- *Simply use a model that is "good enough," even if it doesn't reach the desired performance.*

Best/worst possible case
What would be the best possible outcome for you?

If the goals cannot be achieved (e.g., the model doesn't work to the desired performance or fails completely), what would you do?

Interview Questions for Data Scientists

Interviewing data scientists is not easy for many reasons. As the book explained in Chapter 10, there is not a single data science curriculum that someone should abide by. You get tribes of data scientists that can have different mentalities and use different tools. This makes interviewing data scientists very challenging. This becomes even more challenging when someone is your first hire.

These questions are designed to help you understand how good someone is.

What are the different types of machine learning? Can you explain them to me?

This is a very simple question. Someone who has spent time reading the subject will know that the main types are supervised learning, unsupervised learning, active learning, semi-supervised learning, and reinforcement learning.

Active learning and semi-supervised learning are not very popular, so maybe not everyone knows about them, but you can give bonus points to someone who does.

What is the difference between descriptive and inferential statistics?

Descriptive statistics refers to the statistics that most people do in high school, like summary metrics and graphs. Inferential statistics deals with the problem of inferring something about a population by only getting a sample out of it. Statistical modeling and hypothesis testing, they are all part of inferential statistics.

Which algorithm would be better in a given problem: Random forest or Naïve Bayes?

This is a trick question. Random forest is a much more successful and popular algorithm than Naïve Bayes. However, the no-free lunch theorem tells us that there is the possibility of simple algorithms being better than more complex ones in a given problem. This relates to the inductive bias that an algorithm might have.

While a full technical exposition of this argument is beyond the scope of this book, you will essentially expect two kinds of answers. Data scientists that lack the proper theoretical background will say "random forest." When asked why, they might not be able to properly justify their choice or say something like "it's a good algorithm." Data scientists that fully understand the theory behind machine learning might say something like "it depends" and will give a more thorough answer, which will demonstrate that they understand that a complex algorithm is not always the right choice.

You have a dataset of 1000 rows and 1500 variables. Which algorithms would be suited to this problem?

This is a particular type of challenge where the variables are more than the rows. These problems can be solved through algorithms that are good in handling a large number of variables. Some choices include

1) Elastic net
2) Random forests
3) Gradient boosted trees
4) Heavily regularized neural networks

If someone misses this question, it is not very important, but answering it probably demonstrates solid understanding of some important concepts.

Is more data always good?

This is a trick question. While more data in terms of rows is good, more variables is not always a good thing. There is a concept called "the curse of dimensionality." According to this, adding more variables can make sometimes a problem more difficult to solve, especially when the variables do not contain much information. Hence, you expect that someone who has the right background to answer that more rows of data are good, but more variables might not be necessarily good and whether they are good depends on the problem and the quality of the data.

Other things to ask about

GitHub repository, or samples of code. Samples of code are actually better indicators of someone's skill and interest in the area than abstract coding exercises, like the ones you see in most interviews.

Kaggle. While many brilliant data scientists might not really have the interest or time to participate in Kaggle competitions, having a Kaggle account is always a good thing.

A degree from a top university.

Side projects that might relate to your particular challenges.

The New Solution Adoption Questionnaire

It is often easy to get carried away by new technologies. Every 1–2 years, there is some new big name in town, and all the companies are racing to adopt the new tools. However, quite often, those solutions are expensive to implement, might not carry any benefits to you, and might be quite immature and untested. There have been countless businesses that rushed to use Hadoop, NoSQL, blockchain, and other technologies without really needing them. This simple questionnaire should help you understand whether you should adopt a new solution or not.

Step 1

Goal: Understand your objective and what you are trying to achieve. For example, what if you want to use a new kind of database, what are the issues with the current one? Long read times, scaling up, or something else?

Step 2

Enumerate each solution, context it was created in, and pros and cons. Read the white paper if there is one available. For example, I've filled in the responses for the Ethereum blockchain in the top row of the following table.

Solution	Goal of the solution	Creator/maintainer	Pros	Cons
Ethereum blockchain	*Decentralization, immutability, smart contracts*	*Ethereum Foundation*	*Very well tested solution*	*Blockchain solutions face speed issues (whether this is relevant depends on your particular challenge)*

Step 3

Analyze costs and risks. Every migration or adoption of a new technology contains unseen risks. Answer the following questions to map out risks and solutions:

Can your current engineers do this or you need to hire more people?

If you need to hire someone new, how easy is it to find someone, and how much would it cost?

What could go wrong? How can you mitigate the risks?

How much money would you save after the solution would be implemented?

How would your service or product improve by adopting the new technology?

I

Index

S. Kampakis, *The Decision Maker's Handbook to Data Science*,
https://doi.org/10.1007/978-1-4842-5494-3

E, F, G

T

U, V, W, X, Y, Z