The Data Science Method (DSM) — Documentation

Aiden V Johnson Follow
Jan 20 · 5 min read ★

This is the sixth and final article in a series about how to take your data science projects to the next level by using a methodological approach similar to the scientific method coined the Data Science Method. This article is focused on the documentation step, which includes:

- Reviewing the modeling results
- Presenting and sharing your findings (data storytelling)
- Finalizing code
- Finalizing model documentation

If you missed the previous article(s) in this series, you can go to the beginning here, or click on each step title below to read a specific step in the process.

The Data Science Method

1. Problem Identification
2. Data Collection, Organization, and Definitions
3. Exploratory Data Analysis
4. Pre-processing and Training Data Development
5. Modeling
6. Documentation

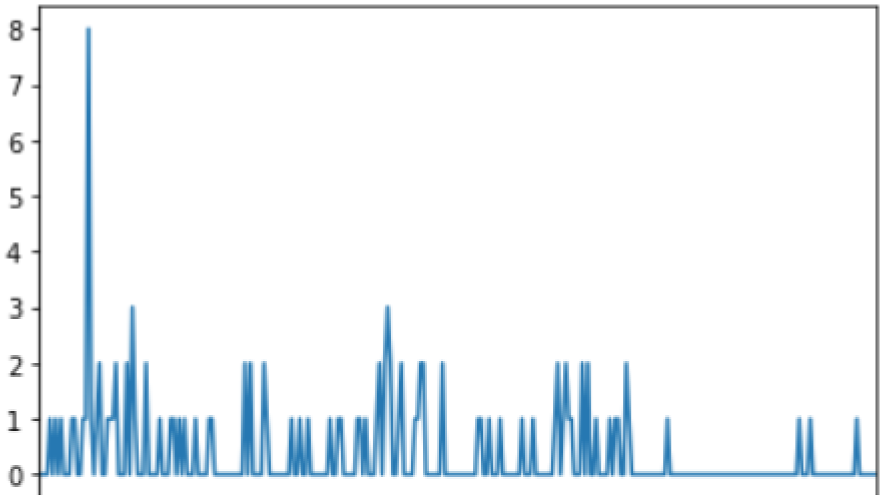
Reviewing Results

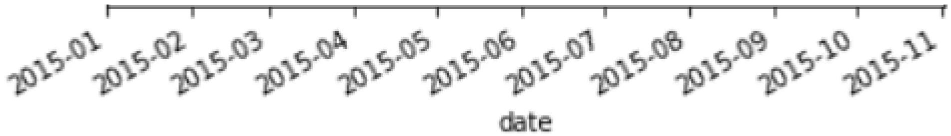
```
y_pred = model.predict(x_test)
print(explained_variance_score(y_test, y_pred))
0.92
```

When reviewing your data science project results you are first looking at how the predictive model performed from a mathematical standpoint, but also from the business insights perspective. For example, if we want to guide the management on how to price a product, we can build a model on comparable products and predict the new product price based on our model. The predicted result is the ‘expected price’ for the new product. In the code block above we built a model with an R-squared value of 92%, which provides evidence that the model is predicting well on our test hold out data set. Given the model is performing well we can now use the same model to predict our new product with the same features associated with it as those used in the model training.

```
predicted_price = model.predict(new_product)
print ("The expected price is $%s " % ')
The expected price is $88.72
```

Similarly, if we are looking to forecast maintenance in order to prevent manufacturing downtime we have to start with reviewing sensor data for anomalies in the equipment. An anomaly is an event of the sensor failing over some time horizon. This is considered an anomaly because the majority of the time-series sensor data show reasonable values, and it is not until the sensor fails that maintenance event is indicated. Often the sensor will return zero when everything is running smoothly.





Sensor data stream where anything other than zero indicates a failure

In this sensor example, we have eight failure events at the beginning of our time-series, which from a manufacturing perspective could cause a big disruption and it would be ideal if those failures could be forecast and preventive maintenance be applied before that event occurs to avoid manufacturing downtime. Therefore our data science modeling should be oriented toward effectively forecasting the failure events across the temporal dimension. This is different than building a model to effectively predict failures from non-failures, where the separation between the training and testing data set could be random.

When reviewing the results consider the key factors you identified in the problem identification step and develop data visualizations to communicate the relationship between the key factors and your predicted outcome from the lens of the business problem and recommended action.

Presenting and sharing your findings (data storytelling)

This is the most important part of your entire data science project. Doing diligent and thoughtful model development only matters if your models get used. Unfortunately, it is all too common for a data science project to never get put into action, even after weeks of work. Your goal should be to convert your audience into believers of a better future given your recommendation is implemented. This starts by stating the current state of reality per the problem identification step. Once you establish the current state, guide your audience to the future state of rainbows and butterflies, or higher revenues and lower costs. This concept was identified by Nancy Duarte as the secret structure of great talks. Nancy is an expert in presentation design and I highly recommend you watch her Ted talk. One of the most effective ways to communicate a future the audience is interested in is by tailoring the presentation to their perspective. This brings us to our first step in developing a persuasive data story; identify your audience and adapt your presentation in style and form. This might result in changing the medium you use to communicate, from Jupyter Notebook to Slidedecks. So let’s review the key components:

- identify the audience and adapt to the appropriate level of detail
- build a narrative around the status quo versus the fabulous future
- communicate your recommendations as established by your data science project.

For more in-depth information check out the HBR Guide to Persuasive Presentations.

. . .

Finalizing code

The primary purpose of the finalizing code step is to ensure that the reuse of your code by you or others is not burdensome. In most developed data science teams you will work in a version control environment such as Git where your code is merged into branches alongside other data scientists and engineers. This is a great place for the team to share code as commonly used scripts or utilities can be shared readily by the team. However, even if you are working independently it is good practice to save your code using Git and to create flexible functions and scripts that can be written once and used again and again throughout your work. Make sure to provide some documentation or comments within each function or script describing the dependencies, inputs, outputs, and possible todos. Adding a short description or a detailed name to a Jupyter notebook can help leverage it later for a different project or to share it with a colleague and please clean up any extraneous or non-functioning code.

Finalizing your modeling code may also involve moving the model to production or serializing the model for later access. Putting your model in production varies substantially depending on your organization's infrastructure and you will want to communicate with the data engineers and solutions architects about what additional scripts they may need to get the model production-ready.

. . .

Finalize model documentation

This is the time and place to corral the data science modeling project into a succinct document that gives the details of what you built and how well it performed.

Create a document called the Model Report. Here is the general layout to give you an idea of where start.

Problem Identification Overview

- Define the question specific to modeling activities

e.g Predict device failures.

- Identify the data needed and or available:

e.g Daily aggregated telemetry device failure data.

- Define the data Timeframe:

e.g. 01/01/2015–11/02/2015

- Describe the Modeling Response:

e.g. Binary, 0 or 1, non-failure = 0, failure = 1

-Unsupervised or Supervised Classification or Regression Model:

e.g. Supervised Classification

-What Deliverables will be generated:

e.g.PDF outlining modeling process from data exploration to best model results.

Data Preprocessing steps of note

- e.g. dropped duplicate rows
- e.g. created PCA for dimension reduction

Model Description

- Input data size and features
- Model Algorithm and Parameters

- Model iterations can be discussed here if they are not saved out separately

Model Performance

- R^2, RMSE, Confusion Matrix, Precision & Recall, AUC/ROC

Model Findings

- Feature Importances

Next Steps

- Ready for production?
- Test new hyperparameters or different modeling methods

. . .

Phew, that's all folks! You've now seen all the steps involved in the data science project following the Data Science Method framework. Adopt and adapt as you see fit. Happy Modeling!

Data Science Data Storytelling Documentation Machine Learning Python