Snapshots of modern mathematics
from Oberwolfach

№ 11/2018

Data assimilation: mathematics for merging models and data

Matthias Morzfeld[1] • Sebastian Reich [2]

When you describe a physical process, for example, the weather on Earth, or an engineered system, such as a self-driving car, you typically have two sources of information. The first is a mathematical model, and the second is information obtained by collecting data. To make the best predictions for the weather, or most effectively operate the self-driving car, you want to use both sources of information. Data assimilation describes the mathematical, numerical and computational framework for doing just that.

1 How we predict the future

Predicting the weather on Earth is interesting for many reasons. People might simply want to know if they need to carry an umbrella tomorrow. Safe operation of airports requires that one knows about the movements of storm systems. One must warn people in advance of severe weather such as flash floods, tornados, or hurricanes. The launch of a spacecraft also depends on weather conditions. In

[1] Matthias is supported by the Office of Naval Research under grant N00173-17-2-C003, by the National Science Foundation under grant DMS-1619630, and by the Alfred P. Sloan Foundation.
[2] Sebastian is supported by the Deutsche Forschungsgemeinschaft (DFG) under grant CRC 1114 "Scaling Cascades in Complex Systems" and grant CRC 1294 "Data Assimilation".

fact, Earth's weather is so interesting that the one scientist almost everybody encounters on a daily basis is the meteorologist that presents your local weather forecast.

Since there is so much interest in weather forecasting, there are many people who study it, using a variety of scientific methods and looking at the problem from different perspectives. For example, we measure temperature and humidity where we can, we use satellites to track storms, while airplanes and ships report the wind speeds they encounter. In addition, physicists and mathematicians work on developing mathematical models for the weather. Computational scientists work on how to use these models for simulations of Earth's weather. The weather forecast that you see on the evening news is a result of combining all that information.

Often, the combination of information from mathematical weather models and "observations" of the weather is done by using *conditional probability*. Our model gives us certain probabilities, say, the probability of rain on a given day, and the observations are used to improve these probabilities. Let us explain this using a simple example. Suppose that I roll a dice and you must guess which number comes up. You could equally well say 1, 2, 3, 4, 5, or 6, as all guesses have the same probability of being correct. If I tell you that the number I see is larger than 3, then you should not guess 1, 2, or 3, since these numbers have become impossible in view of the additional information I have provided. In other words, your guess of the number should change because you obtained new information.

The situation is similar when you forecast the weather. You use a mathematical model to make a forecast. Then you obtain measurements of current weather, and you want to modify your model, that is, modify the probability of the weather outcome, to account for this new information. For example, if you predicted sunshine for today, but today you observe rain, then you need to change your model and forecast for tomorrow in view of that observation. "Data assimilation" describes the mathematical foundations and numerical algorithms of how to make such changes to your model.

Weather prediction is one example where data assimilation is useful, but more examples can be found in almost every field of science and engineering. In robotics, for example in a self-driving car, data assimilation algorithms use sensor data to locate the car in a given map. The map itself must also be updated regularly based on the sensor measurements. For example, it is important that a pedestrian crossing the street becomes part of the map the car uses to navigate.

Another example is hydrology, where one studies the movement, distribution, and quality of water on Earth. When it rains, some of the rain flows across Earth's surface to end up in streams and rivers. Some of the water, however,

penetrates the soil and travels underground. If you want to predict where this water ends up and how it flows under the surface, you will need to know what the structure under the surface looks like. To get an idea, you might drill a few holes and make some measurements which you then want to combine with a mathematical model that describes the subsurface flow in between your measurements. The basic idea then is as before: you want to combine your model and data. You want to use data assimilation.

2 Mathematics of data assimilation

Data assimilation is done every day. In global numerical weather prediction, it is actually done every six hours. How? There are currently three main approaches to data assimilation.

1. Solve the actual problem.
2. Solve an optimization problem.
3. Solve a simplified problem.

An elegant way of performing data assimilation is to compute the conditional probabilities, as described above, that describe the mathematical model in view of the data you collected [1, 6]. These conditional probabilities taken all together describe the *probability distribution* of our model. We want to obtain what is called the "expected value" of this distribution.

Let us return to the example of the roll of a dice we introduced earlier. The dice has six sides, with the numbers 1, 2, 3, 4, 5 and 6. The appearance of one side, say 1, is as likely as the appearance of any other side. That means that the probability of rolling a 1 is 1/6, and the probability of rolling a 2, 3, 4, 5, or 6 is also 1/6. In this case, the distribution is called "uniform". The expected value is defined to be the sum of the numbers, multiplied by their probabilities:

$$E = 1 \cdot \frac{1}{6} + 2 \cdot \frac{1}{6} + 3 \cdot \frac{1}{6} + 4 \cdot \frac{1}{6} + 5 \cdot \frac{1}{6} + 6 \cdot \frac{1}{6} = 3.5$$

You can approximate this expected value by repeatedly rolling the dice, that is, by drawing "samples". Suppose you draw 10 samples and you get the sequence shown in the first two rows of Table 1 (we actually rolled a dice to obtain this sequence). An approximation of the expected value is the *sample average*, which means the sum of the ten outcomes, divided by ten:

$$\hat{E}_{10} = \frac{1}{10} \cdot (5 + 3 + 6 + 2 + 1 + 2 + 5 + 1 + 3 + 1) = \frac{29}{10} = 2.9$$

The approximation gets better when you have more samples. An approximation based on 60 samples, which are also shown in Table 1, is given by the sum of

Roll-number	1	2	3	4	5	6	7	8	9	10
Outcome	5	3	6	2	1	2	5	1	3	1
Roll-number	11	12	13	14	15	16	17	18	19	20
Outcome	5	3	1	4	1	2	4	5	6	3
Roll-number	21	22	23	24	25	26	27	28	29	30
Outcome	5	2	1	3	1	6	3	1	5	3
Roll-number	31	32	33	34	35	36	37	38	39	40
Outcome	3	6	3	2	1	6	3	3	1	4
Roll-number	41	42	43	44	45	46	47	48	49	50
Outcome	5	6	4	3	1	6	4	3	6	6
Roll-number	51	52	53	54	55	56	57	58	59	60
Outcome	1	5	5	5	6	1	3	3	1	6

Table 1: Rolling a dice 60 times

all 60 outcomes, divided by 60:

$$\hat{E}_{60} = \frac{1}{60} \cdot (\text{sum of all outcomes}) \approx 3.42$$

You can see that the approximation $\hat{E}_{60} \approx 3.42$ is now pretty close to the expected value $E = 3.5$. We encourage you to draw more samples, say 100, and you should get an answer even closer to 3.5. You can also tell ten of your friends to each draw 100 samples, then collect the results and average over all outcomes you and your friends observed. The result should be close to 3.5. If you were able to draw infinitely many samples, then you would obtain 3.5 exactly [3].

Returning to our general situation, our aim is to estimate the expected value of the distribution given by the conditional probabilities. The method of sampling, as illustrated above, is called the *Monte Carlo method*. More precisely, this method consists of estimating an expected value by drawing many samples from a given distribution and finding the average of all of these samples. As we take more samples, the estimate should be closer to the desired value. The name "Monte Carlo" comes from that of the famous casino in Monaco and dates from the 1940s, when this sampling method was used to solve problems in physics related to the development of the atomic bomb. However, the method has a longer history; we can find examples appearing in the mathematical and statistical literature starting from the 1900s.

The main justification for the use of Monte Carlo methods comes from what are known as the "laws of large numbers". There are two of these important results, the "weak" and the "strong" law. Roughly speaking, the weak law tells us that if we assume that the expected value we want to estimate exists, as

we take the average of more and more samples, the error between the sample average and the desired one shrinks to zero. The strong law gives us slightly more information, in that it says that for any chosen margin of error, no matter how small, there will always exist a number such that if we average at least that many samples, we will have an error that is at most the one we have chosen. Notice that we had to assume that the expected value we want to estimate exists. But actually we have already done this: if we didn't believe that this value existed, we wouldn't be trying to estimate it.

In practise, if you wanted to use Monte Carlo for data assimilation, then you would create ways of drawing samples from the conditional probabilities defined jointly by your mathematical model and the data. You can "use your imagination to do this" [3] and many scientists and mathematicians work together on finding new and effective ways to draw samples from probability distributions that are not necessarily well understood. Designing such methods for data assimilation is very difficult. Briefly, and we will explain in more detail in Section 3, the reason is that doing data assimilation in this way becomes increasingly difficult the larger the problem is, in the sense of the number of variables to account for and the quantity of data collected. This is called the "curse of dimensionality", and, for this reason, data assimilation techniques of this kind are only used in relatively "small" problems.

The other two techniques mentioned above, optimization and simplifying the problem, are often used in practice, even on very large and important problems such as numerical weather prediction. By an "optimization problem" we mean that you try to calibrate your model in a way that its output is as close as possible to the data you collected. You can quantify the mismatch, or error, between model output and data by what is called a "cost function", and your "optimal" model output is the one that yields the smallest error, which is also the smallest value given by the cost function. Typically, the cost function is related to a formulation of the data assimilation problem by conditional probabilities again [7]. Optimization-based data assimilation algorithms are in use in operational numerical weather prediction, and some people say that progress in data assimilation is one of the main drivers for the increase in forecast skill over the past decades [2].

However, since the implementation of these techniques in the form of computer code can be tricky, there are problems for which this optimization approach is too complicated and difficult. In these cases you can make use of a very powerful method for solving difficult problems: just don't do it. Instead, you solve an easier problem, whose solution is not too far from the difficult problem you really want to solve.

[3] This is a quote of somebody who creates such algorithms for a living. We collected this quote during a workshop at MFO on the topic of data assimilation.

In data assimilation, this approach can be applied as follows. If the mathematical model is simple enough, the conditional probabilities that appear are also simple. By a simple mathematical model we mean one that is *linear*. Recall that a linear function $f(x)$ is one that satisfies $f(\alpha x + \beta y) = \alpha f(x) + \beta f(y)$, where α and β are any numbers you want. Linear functions are usually easier to deal with than functions that are not linear. For example, it is much easier to solve the equation $f(x) = 0$ if we have the linear function $f(x) = 2x + 1$ than if we have the non-linear function $f(x) = x^2 + \cos(x)$.

Linear models that appear in data assimilation satisfy similar conditions, and linear models are also "easy" to deal with. A linear model gives rise to "easy" conditional probabilities, which are described by a "Gaussian" distribution. What makes Gaussian probabilities simple is that all you need to know about them are two numbers, namely, the *mean* and the *variance*.[4] The mean is the average value you expect to encounter, that is, it is another name for the expected value. The variance describes the variation around the mean that you should expect when you perform repeated experiments. In short, if you have a linear model and Gaussian probabilities, then you can perform data assimilation "easily" (see also the next section). Unfortunately, almost every physical process or engineered system does not appear naturally as a linear model, and the probabilities that you observe in practice are rarely Gaussian.[5]

One way forward is to "linearize" your model. That means you replace the actual model by a linear version of it. The linearized model now gives rise to Gaussian probabilities and so you are in business to solve this simplified problem. The solution you obtain can be very close to the solution of the nonlinear problem if the linearization is done carefully enough to capture most of the dominant behaviors of your model. The Kalman filter [5] is the original data assimilation method for a class of linear problems, and versions of it, including the "extended" Kalman filter and "ensemble" Kalman filter [4] are used in many nonlinear problems.

3 Why is this difficult?

Data assimilation seems "easy". The problem is elegantly formulated in terms of conditional probabilities and there are three classes of methods to solve the problem. What is difficult about this?

One difficulty lies in the problem formulation. In order to set up the

[4] This dependence on exactly two parameters is similar to linear functions, for which you only need to specify the slope and the y-intercept.

[5] There is an important result in probability theory, called the "Central Limit Theorem", which says that in some circumstances these probabilities are actually Gaussian. However, there are many situations where this theorem is not applicable. For more information, see [3].

conditional probabilities, one relies on assumptions about the distribution of errors. For example, one must specify how "far" one expects the model outputs to be from the collected data. This is inherently difficult, because it requires statements, often in a precise mathematical sense, about things we truly do not know. How can we define the errors of a measured temperature to the output of a mathematical model, which is invented and does not "know" what a temperature actually is? Sometimes, one tries a few (or many) assumptions that describe such errors and then uses the one that leads to the "best" results. Here "best" usually refers to the set of assumptions that lead to the most useful forecasts and model outputs. Mathematical or computational convenience may also play a role. If one is not sure what assumptions to make, one first can try assumptions which simplify the problem.

Another difficulty is that the conditional probabilities are often far from "standard", that is, far from probabilities that mathematicians or scientists understand well. This is mostly due to the fact that models of complicated physical or engineered processes are also complicated. Drawing samples from conditional probabilities that arise from complex mathematical models is a current topic of mathematical research.

A third difficulty is that many problems are "big". By big we mean that the number of variables that define the conditional probabilities is very large. Coming back to the example of numerical weather prediction, one can imagine that a model for the global weather requires that one specifies a lot of variables. In principle, you need to specify all meteorological quantities, for instance, the temperature, wind and humidity, at all locations on the globe. This is not possible because it would required infinitely many numbers. What one does instead is to specify meteorological variables on a "grid", which specifies a fixed but large number of locations around the globe. In this way, a global mathematical weather model can have several hundred million variables. The number of data points is also large. Every six hours, between two and ten million measurements are used to update the mathematical model.

The model itself, and the data assimilation method must then be implemented in the form of computer code. For a problem of the size of the global atmosphere, this requires careful algorithm design and coding. Today's data assimilation technology for numerical weather prediction is often based on optimization techniques or methods for simplified (linearized) problems. Many researchers also work on combining the three approaches.

In problems that are characterized by a large number of variables, as already mentioned in the previous section, one must often fight the "curse of dimensionality".[6] What is meant by this curse is that solving a problem becomes

[6] This is true for many problems, not just data assimilation.

increasingly difficult as the number of variables, or "dimension", increases, and the rate with which the difficulty increases is very high.

You may recall the exponential function $f(x) = e^x$ (where $e \approx 2.718$ is Euler's number), which grows very quickly with x. For example, $e^1 \approx 2.718$, $e^2 \approx 7.389$, $e^3 \approx 20.086$. The "curse of dimensionality" is that if x is the number of variables, then the difficulty level is e^x. As an example, in Monte Carlo methods, the difficulty level may by described by the number of samples required to compute an expected value with a given accuracy, for example, with 10% error (in the rolling the dice example above, we used 60 samples). For the data assimilation of numerical weather prediction, the number of variables is 200 million (or more). The difficulty level, as measured by the number of samples required is then $e^{200 \cdot 10^6}$ which is much larger than the number of atoms in the universe (estimated to be between 10^{78} to 10^{82}). It is impossible to draw that many samples, even on today's powerful super-computers. There is currently a lot of interest in the problem of somehow overcoming the curse of dimensionality, especially in Monte Carlo sampling and data assimilation, but at this time it is unclear how to do it. In a future snapshot, you might read about the solution of how to overcome these difficulties without making overly simplifying assumptions.

4 Summary

Data assimilation means to merge a mathematical model with information obtained from data (measurements). Many scientific and engineering problems require data assimilation. The list of problems where data assimilation is useful is very long but includes numerical weather prediction, hydrology, personalized medicine, cognitive science, and robotics. We have discussed a few of these applications but focussed on numerical weather prediction. We have also introduced the three main approaches to solving data assimilation problems, and explained why solving data assimilation problems is difficult.

The difficulties one encounters are of course different from problem to problem. You can imagine that solving a data assimilation problem in the context of numerical weather prediction is very different from solving a data assimilation problem related to a self-driving car. In fact every scientific or engineering problem has its very own challenges and characteristics. Challenges can include an immense number of variables, or probabilities that are far from those we understand well. Each data assimilation problem also comes with its own mathematical model and data types, and the computational architecture you can use to solve it may also vary. For example, you are probably given a large computer when you make a weather forecast, but to operate a self-driving car, you have access to much less computational power.

In view of data assimilation's wide use in different applications, it is surprising that all of these problems have essentially the same formulation in terms of conditional probabilities and three main avenues to success (Monte Carlo methods, optimization and linearization). It is up to the mathematician, engineer and scientist to combine these methods to from a suitable recipe for the solution of the data assimilation problem at hand. In this context, mathematics is particularly useful to identify "classes" of problems that consist of problems which are characterized by a certain set of common specifications. Once a problem class is determined, one can look for suitable algorithms for the solution of all problems within that class.

The authors currently focus on analyzing approximate solutions to data assimilation problems, studying computational requirements of the various solution techniques, and also designing new algorithms for the numerical solution of data assimilation problems. To do so, they use "classical" numerical analysis techniques such as "expansions in a small parameter", "optimal transport", and "numerical optimization".

References

[1] M. Asch, M Bocquet, and M. Nodet, *Data assimilation. methods, algorithms and applications*, SIAM, 2017.

[2] P. Bauer, A. Thorpe, and G. Brunet, *The quiet revolution of numerical weather prediction*, Nature **525** (2015), 47–55.

[3] A. J. Chorin and O. H. Hald, *Stochastic tools in mathematics and science*, third ed., Springer, 2013.

[4] G. Evensen, *Data assimilation: the ensemble Kalman filter*, Springer, 2006.

[5] R. E. Kalman, *A new approach to linear filtering and prediction problems*, Journal of Basic Engineering **82** (1960), no. 1, 35–45.

[6] K. J. H. Law, A. Stuart, and K. Zygalakis, *Data assimilation: a mathematical introduction*, Springer, 2015.

[7] O. Talagrand and P. Courtier, *Variational assimilation of meteorological observations with the adjoint vorticity equation. I: Theory*, Quarterly Journal of the Royal Meteorological Society **113** (1987), no. 478, 1311–1328.

Matthias Morzfeld *is an assistant professor of mathematics at the University of Arizona (USA).*

Sebastian Reich *is a professor of mathematics at the University of Potsdam (Germany) and at the University of Reading (UK).*

Mathematical subjects
Numerics and Scientific Computing

Connections to other fields
Chemistry and Earth Science, Physics

DOI
10.14760/SNAP-2018-011-EN

Snapshots of modern mathematics from Oberwolfach provide exciting insights into current mathematical research. They are written by participants in the scientific program of the Mathematisches Forschungsinstitut Oberwolfach (MFO). The snapshot project is designed to promote the understanding and appreciation of modern mathematics and mathematical research in the interested public worldwide. All snapshots are published in cooperation with the IMAGINARY platform and can be found on www.imaginary.org/snapshots and on www.mfo.de/snapshots.

Junior Editor
Sara Munday
junior-editors@mfo.de

Senior Editor
Carla Cederbaum
senior-editor@mfo.de

Mathematisches Forschungsinstitut
Oberwolfach gGmbH
Schwarzwaldstr. 9–11
77709 Oberwolfach
Germany

Director
Gerhard Huisken

Mathematisches
Forschungsinstitut
Oberwolfach

IMAGINARY
open mathematics