EPISODE 475

[INTRODUCTION]

[0:00:00.3] JM: A company's approach to data can make or break the business. In the past, data was static. There was not much data. It's sad in Excel and it was interacted with on a nightly or a monthly basis. Now, data is dynamic. It's real-time and it's huge. To tap into available data, many industries have oriented themselves to becoming data-intensive. With many new industry sectors becoming data-driven, a new field called data science emerged.

As a new field, data science has attracted a lot of attention from professionals with diverse backgrounds. Describing what is data science and who was a data scientist is not easy. As technologies surrounding the field continue to evolve and new verticals are added, the discourse surrounding the field has attracted different voices putting forward their definition of the field.

In this episode, Zacharias Voulgaris joins the guest-host, Sid Ramesh to discuss the developments in the field. He's the author of several data science books. In today's conversation, Zacharias explains what he means by the data science mindset including trends and misconceptions that people have on the field.

Thanks for listening, and I hope you enjoy the episode.

[SPONSOR MESSAGE]

[0:01:28.6] JM: Artificial intelligence is dramatically evolving the way that our world works, and to make AI easier and faster, we need new kinds of hardware and software, which is why Intel acquired Nervana Systems and its platform for deep learning.

Intel Nervana is hiring engineers to help develop a full stack for AI from chip design to software frameworks. Go to softwareengineeringdaily.com/intel to apply for an opening on the team. To learn more about the company, check out the interviews that I've conducted with its engineers.

Those are also available at softwareengineeringdaily.com/intel. Come build the future with Intel Nervana. Go to softwareengineeringdaily.com/intel to apply now.

[INTERVIEW]

[0:02:23.4] SR: Zacharias Voulgaris is the author of several data science books. Welcome to Software Engineering Daily.

[0:02:29.5] ZV: Hi. It's good to be here.

[0:02:30.7] SR: It's great to have you on the show. You have written several data science books and have reviewed many programming books. You have a data science blog and work as a CTO at a data science startup. In your latest book, *Data Science Mindset Methodologies and Misconceptions,* you write about heuristics, artificial intelligence and ethics. Ai is probably what people think of when they hear data science. It's not usually the case that data science book talks on either ethics or heuristics. What is heuristics and how does that relate to data science?

[0:03:05.8] ZV: That's a very good question, Sid. First of all, heuristics is something that is everywhere in programming, not just in data science. But in data science I believe it finds a very sorted application, and that's because it involves different metrics or simple programs that do some kind of transformation to make something into more a comprehensive form.

For example, there's a heuristic I have developed that helps us understand how discernible the classes of dataset are. That finds those applications in classification, for example. There are heuristics that help us understand how well a classifier or any other predictive analytics model performs, and these are commonly used in practice. They may not refer to them as heuristics, but people use them and they add a lot of value in whatever process of data science we are involved in.

[0:03:54.6] SR: Can you give us an example of a heuristic?

[0:03:57.2] ZV: Sure. For example, the F1 metric for assessing the performance of a binary classifier is a commonly used heuristic, which is basically a formula that combines the precision

and recall metrics in a way that takes both of them into account and doesn't get too carried away by a very good performance in one of them. It is more conservative it takes a value that is closer to the smaller one.

If you have a classifier performance really well in terms of precision, but not so well in terms of recall, F1 metric captures both of these but focuses more on the smaller one. This a very liable metric, because if you have a very high F1 score, as its software recalled the value of the F1 metric, then the classifier is pretty good in predicting that particular class.

[0:04:46.9] SR: When you say heuristic, you also have the optimization algorithms, which is usually what people use. Now, what's the advantage that heuristic offers over an optimization algorithm?

[0:04:58.2] ZV: That's a very good point. Optimization is closely linked to heuristics, because what they do optimize is a fitness function, which is usually the case that is a heuristic. If it's not the heuristic itself, it is linked to a heuristic. One, for example, you try to train a neural network, which is a common AI method in data science. Then you optimize in terms of error rate. You want to minimize that, and error, in a way, is a heuristic. Of course, it's a bit more complex than that. It's not as simple as finding the global optimum or some at least local optimum of the error rate function, but the heuristic error rate plays a dominant role in that process.

Of course, there is also the process of how you train each particular node, which makes it more sophisticated, but without the heuristic of error rate, it will be really hard to train a neural network effectively and efficiently.

[0:05:54.7] SR: When using a heuristic, usually there is the compromise of accuracy. Now, do heuristics make up for that or is that something one should keep in mind?

[0:06:03.0] ZV: Yeah. Often times when you use a heuristic, you do basically a back of the envelope calculation for something that you really care about and it's like a basic law in physics. They may not be super accurate, like the Newton laws of mechanics. They're not super accurate, but they're good enough.

For most of the cases, they work fine, but if you have some extreme cases, like things going close to the speed of light, they don't apply. Data science is the same. For the majority of cases, the majority of problems, heuristics that we use are fine. If you want something super complicated and the heuristics we use don't work so well, you may need to come up with some variations of them or some new heuristics altogether.

[0:06:45.4] SR: Another topic you start a conversation on is ethics. It's a very important topic, something that does not have a lot of spotlight in the conversation surrounding data science. Why do you see a need for that?

[0:06:59.5] ZV: Yes, because there's not enough spotlight. I think it's a good enough reason to get involved in this conversation, because it is an important topic in every profession, I believe, just because in data science there are so many other things that appear to me more interesting. It doesn't mean that we should neglect the ethics. Ethics is becoming more and more relevant nowadays, because data science ethics is based basically on the security and the privacy of the data we use. We need to keep those things intact. We cannot really risk exposing people or organizations behind our datasets. If we don't take that into account, somebody will get into trouble. Maybe not us, but somebody up in the organization hierarchy will get into trouble, because of a misdeed.

Ethics in data science has to do with all that. It is not only taking care of a data in a very methodical manner, like it is taught in many books and courses, but also taking into account the end-user and the people who manage that manage and make sure that nobody is exposed and nobody's privacy is exposed through an analysis and that nobody is feels insecure because of what we come up with, the conclusion we come up with.

[0:08:18.0] SR: You talk about considerations that one should have on ethics and you make a very important distinction between ethics and morality. Can you talk what the principles that you would look at when you're considering ethics on a data science problem?

[0:08:32.7] ZV: Yes. The main ones are basically keeping the data secure and keeping the data private, and that's ethics. Everybody can be ethical. That's my point. That's why it is discriminated from morality, because morality has a very philosophical stance towards things.

Some people may say, “Okay. Well, I’m immoral, or I don’t care about morality at all, or my morality is different than yours.” This is a valid argument that your morality is different, so you value different things as important in life, so you don’t feel obliged to follow my morality, which may be completely different, and that’s fine. In ethics, you can really say, “Okay. I don’t have ethics.” If you’re working as a data scientist, you have to have ethics. It’s part of being a professional.

Morality is important and everybody think should be moral, but nobody is obliged to be moral in data science. However, everybody has to be ethical if they want to be a professional in this field.

[0:09:26.9] SR: I agree. One other topic that you talk about, which is very interesting, is artificial creativity. Can you throw some light on this topic?

[0:09:34.7] ZV: Sure. Artificial creativity, the way I understand it, because I’m not an artist per se, is when you use AI to create something from scratch. Something that the system has never seen before. The computers themselves are not artists, like most people are not artists, but given enough data and enough wiggle room, they can experiment with that and come up with some new forms. This could be a new picture, for example, or a new piece of music. This can be something that we as humans can appreciate and see as something of some value. Maybe not some monetary value per se, but it has some value, some beauty in it, and that is often referred to as artificial creativity. However, I go beyond that and say that creativity is not just coming up with different forms of art, because that’s great, but some people don’t care about art.

Creativity is important, because it helps us come up with new solutions to problems, and that something that is valued more and more nowadays, because knowledge is more easily accessible than ever before, and everybody can access some databases or some knowledge-bases and become knowledgeable about a topic. But to become creative a topic takes effort, and that’s something very useful with computer systems as well, because we don’t want them to just blindly follow rules and to follow some methods we have devised. We want them to be able to think outside the box in a way and come up with solutions that are more novel.

I’ll give you an example of that. There’s this company, I don’t remember its name. That has developed new [inaudible 0:11:15.0] for cars that are based on the data they have collected

from various sensors that have been put into cars in different conditions of driving, and they have managed to optimize the structural makeup of these [inaudible 0:11:31.7] for the cars by themselves. Humans were not involved. The computer came up with some different designs and all of them were developed in such a way that the cars would perform well and they would also save on materials. So it would use less steels for example and still have the same mechanical properties of the [inaudible 0:11:52.6]. This is a creative solution which many people, many people involved in the production process of the car would take into account and use in their new designs. This way they can save up on materials and also not comprise on performance.

[0:12:10.0] SR: That's a very fascinating topic actually to me. I haven't had a lot to read on artificial creativity, but that certainly caught my eye in your book. Going on to the next question, on industry trends, you have meaty laid out the emerging trends with respect to technologies and job profiles. Now, what trends do you see in data-intensive industries?

[0:12:29.7] ZV: I see more and more involvement of text data, specifically data coming from places like Twitter, because data-intensive industries have traditionally focused on things they have been measuring already and these signals may be strong, but not strong enough. I have seen that many places, particularly in the financial sector, tend to incorporate social media feeds more and more and they do sentiment analysis and they use that in tandem with the stuff they have already been using to improve a signal they have. I don't know if that is also great or not. I haven't been involved in projects like that yet, but the fact that they're doing it shows that there is some merit in it.

I won't be surprised if more and more data-intensive industries would follow suite, because first of all this kind of data is easily available. It's not free, because it takes some time to download it and to download enough quantities for it to be meaningful. But nothing in data science is entirely free anyway.

Sometimes these kind of datasets, some of which is curated, so you have to pay more for it. Sometimes all these new datasets add a little value and this is something that people always care about. They don't care so much about paying for it, as long as it brings about some value. In the sense it becomes more and more mature as a field, it is able to incorporate different data

streams much more effectively, and this is something that we see now as a trend and I believe will continue in the near future.

[0:14:03.2] SR: Interesting. In technology surrounding data science, a lot has improved, and you talk of the different new technologies and the new alternators for Hadoop. What trends do you see in the technology realm?

[0:14:18.3] ZV: I see that GPU is becoming more and more relevant. Also, there's a possibility which seems to be a trend of this tiny computers becoming relevant as well in data science. Not so much in training a system, but applying their results on a system, because they are super cheap and easily deployable in different places. If somebody steals one of these machines, it's not the end of the world. Of course, you wouldn't want it to be stolen or damaged, but it is such a low cost that it makes it much more scalable in a variety location, and also in locations where you wouldn't normally have a computer performing some data science system.

For example, you can deploy some of these with some batteries in some remote location that is not close to the city, and these can collect data or analyze even data. I don't know. It depends on the application that would otherwise not be easy to do. Basically, this allows us scale our systems on different locations and perhaps do applications where they were not feasible before.

That's a trend that is worth noting down different technology in data science.

[0:15:31.9] SR: You talk specifically about the rule of AI in the years to come. What will you see for AI in data science?

[0:15:40.2] ZV: I see it continuing being relevant as it is today. Maybe they will wear off a bit and people would care about other things, like the data science mindset, but the AI trend is something that is I believe is here to stay. More and more AI systems will come about. The ones that we have right now may be refined a bit or a lot depending on the technologies as well. Because, for example, Intel comes up with a new kind of processor that is ideal for these kind of systems, maybe the systems will the adopt to the processor as well and try to optimize its performance.

Also, I believe Google has developed its own CPUs of sorts that are designed especially for TensorFlow. If TensorFlow continued being a popular option, it may be the case that it will grow in these kind of applications where we'll have TPUs in place. That's not the only case. I believe that the cloud has a lot to offer as well in data science, and this is a trend that is bound to remain when it comes to AI applications, because the cloud and the AI are two super compatible technologies and many people realize that and take advantage of that. Even if someone doesn't have a very good computer or don't have access to computer cluster to deploy their AI system, it's not difficult for them to go and rent some computing power from Amazon or some other cloud service and do their analysis there, and the bill may be not negligible, but it's still not too high either.

[SPONSOR MESSAGE]

[0:17:20.8] JM: You are building a data-intensive application. Maybe it involves data visualization, a recommendation engine, or multiple data sources. These applications often require data warehousing, glue code, lots of iteration, and lots of frustration.

The Exaptive Studio is a rapid application development studio optimized for data projects. It minimizes the code required to build data-rich web applications and maximizes your time spent on your expertise. Go to exaptive.com/sedaily to get a free account today. That's exaptive.com/sedaily.

The Exaptive Studio provides a visual environment for using back end algorithmic and frontend component. Use the open source technologies you already use, but without having to modify the code, unless you want to, of course. Access a k-means clustering algorithm without knowing R, or use complex visualizations even if you don't know D3.

Spend your energy on the part that you know well and less time on the other stuff. Build faster and create better. Go to exaptive.com/sedaily for a free account. Thanks to Exaptive for being a new sponsor of Software Engineering Daily. It's a pleasure to have you onboard as a new sponsor.

[INTERVIEW CONTINUED]

[0:18:52.3] SR: Mindset is a topic that we'll come to speak shortly. Before that, I actually wanted to touch on this point that you have previously written a book which was probably one of the earliest books on the field. It's called *The Definitive Guide to Becoming a Data Scientist* and published back in 2014. That's when data science as a field started to get some traction.

What do you find missing that you decided on writing your new book?

[0:19:20.9] ZV: First of all, the new book is not based on the first one. I'm not trying to redo the first one by the new technologies. The first one is called *Data Scientist*, and it's all about how to become a data scientist and the things you need to do as a professional to become someone who can offer something to the data science field.

The new book is about the field itself. Whether you are a data scientist or not if you care about data science as a professional or you care about data science as a manager, it doesn't matter. It's all about getting you to understand better the field and appreciate some things so that if you're a manager, for example, you can manage a data science team better. The book is not for data scientist only. While the other one is for people who want to go in data science as a profession.

[0:20:05.8] SR: You also have a Ph.D. in machine learning. Tell us a little bit about your research problem.

[0:20:12.7] ZV: How much time do I have, because I can talk about this all day.

[0:20:16.6] SR: Okay. If you can broadly talk about what your research problem was and give a bird's eye view, because it's in machine learning. I'm kind of interested in knowing what your problem and how you went about it.

[0:20:29.1] ZV: Cool. The problem that I tackled was classification, all kinds of classification, data datasets across different industries. Most of them were very small, because at the time the computing power was quite scarce. Things like cloud computing were not really available to everyone back then. Maybe in some research centers there are some companies, but in the

university I was in, I only had my own computers to work with, basically a desktop at the university and my laptop. Both of them are not really that powerful. Sometimes I'll have to have experiments running overnight. Other times I will have to have experiments day and night before I could come up with some publishable result, because it was partly my fault because I was using MATLAB at the time, which was a popular option for many researches, and in many cases it still is for research purposes. The experiments would take a while to run, and that's why I had to think about efficiency as well when I was designing something.

Originally, my thesis was about intrusion detection, but I didn't go down that path, because it was too specialized I found. Also, there were people who were very much into network security, so they knew things about it already, and to catch up with those people and to be able to publish something of value would take a while and I didn't want to take my time finishing the Ph.D. I want to finish it as quickly as possible so I could get to work.

After talking to my advisor, eventually we decided that it will be best if I had something that was more generic, but not too generic. I wasn't trying to solve all the problems of classification, but I was trying to do something that would improve classification in different areas, not just network security.

I tried different things and they were not seemed to be more meaningful and more scalable across different applications was the discernibility concept, which was basically a metric, a simple metric that would tell us, "Okay. You have these classes in the dataset. Okay. Based on that and how much they overlap regardless of what distribution the data follows. How discernable are they? How easy it is to tell A from B." That doesn't mean just these two classes. If you have a dataset that has 10 classes, it would still work.

This doesn't matter what the machine was either. This would work when you have one dimension or you could have — It will work if you had a thousand dimensions. Of course, it would take longer if you had big dimensionality in a dataset, but at that point it didn't really matter, because I just want to prove a concept. This proof of concept approach is quite common in data science, but as a first attempt. After you have the proven the concept, you have to scale it up.

After I did the first iteration of this metric, the index of discernibility as I called it, I did a variant of it. It was a bit faster. I will not be as insightful, because the original index of discernibility didn't just provide an overview of the hole dataset, but also individual data points, how discernable they are. Again, this is all about classification. This doesn't apply for regression problems or other problems of data science, but I did show that indexing desirability can help a variety of classification systems in data science. I even went so far as to say, "Okay. Well, based on this idea, if you implement it properly, you can even improve the ensemble performance." This was like a big thing at the time because it was very new relatively.

I showed how this thing fits in very well in different applications and how it can bring value. Now, whether this brings value to bigger dataset is questionable, because the index of discernibility at that time was very slow for bigger datasets. Through time, I have improved it a bit and now I can safely say that it can scale very well.

[0:24:12.2] SR: That sounds a very interesting problem to me. I'm sure you would have had a lot of fun solving your MATLAB code. I've, in my pervious — When I was doing my grad studies, I had to deal with METLABS, so I know pretty well the world of academia there. In your latest book, you have thrown a spotlight on developing mindset, a data science mindset as a way of working with data problems. What's so special about the mindset?

[0:24:37.8] ZV: Well, everything is special about the mindset, because it's not the specific thing you do. It's the way you think. If we take a parallel and look at how things are in the programming world, you see that there are two kinds of programmers out there in broad terms. There are people who are super successful and they make six-figures wherever they go and the people who manage to make a living. Everybody does their best. I'm not judging anyone, but the difference between the very successful program and the average one is the mindset, because when you hire a programmer, you don't hire someone to write code for you just. You hire someone who can solve problems and find the possible code for these problems.

It's the same in data science, because data science is strongly linked to programming. It's not a mathematical only approach to problem solving, because you can be very good at the math part, and if you don't do the programming well, the systems will not scale so well or they may not be so efficient. The mindset is what in my view allows someone to go this extra step and

become not just a data scientist, but a good scientist. I think the world needs good data scientists more than data scientists in general. This sounds like an oxymoron, because the world needs data scientists. There's no doubt about that, but I have seen many people who are very good at data science in terms of knowhow. They know all the methods very well and they have experience, some experience with them and they can't get a job. I'm wondering, "Why is this the case?"

After talking to some people, I see that these people were very good, very adept even at the math part and some of them are maybe good at the programming part as well, but they don't really understand what they're doing. They're just very familiar with the different scikit-learn functions and the different classes that it has and they tell you, "Okay. Well, this seems to be the regression problem that this function helps us solve.

I don't understand that maybe the same problem should be formatted better as a classification. Maybe we should do some tweaking in the target variable so that it is more of a classification problem, and so there's as such, or they don't understand that sometimes, "Oh, the features that we have are not that good. Even if we find the best possible regression system out there, it will not perform so well, so maybe need to come up with some new features or drop some features or combine some features. Do something with the features so that we have a very strong signal there to work with.

All that is part is part of a data science mindset, and without the right mindset, even if we have the best tools available to us, we may not be able to do much with them. That's why I believe it's super important to develop the mindset along with the technical knowhow.

[0:27:14.6] SR: What do you recommend that people do to get started on developing their mindset?

[0:27:19.4] ZV: First of all, understanding what the science is and what problems it tries to solve, not just now it solves them. It is fundamental. We often tend to think of the technical aspect of the things we need to work on and forget that we do all that for a reason, and the reason is to bring value to an organization. If we have this in our focus, then whatever we do would be more practical, because we don't need to do the perfect solution. We just need to do

one that's good enough for the resources we have, and these things can be quite different. The perfect solution would take five days for example to implement, but a good enough solution would take one day, and that difference in time of implementation may be crucial, because the organization that hires us may not have the funding to do long scale projects, and they just want to prove a concept for now. Once they do that, they may get some funding and then they can go in more depth about this problem and maybe they can get some more resources so we can do a more thought analysis on the data.

But whatever the case, we have to remember what we're trying to solve and why before we focus much on the how. The how is important too, but without these holistic approach, we cannot really do much. That's why I think we have to understand business first. We have to focus and talk to the people who are calling the shots in the project and understand their pain points before we start implementing processes, before we start developing approach pipelines, because the pipelines are great, but if they're not aligned with a business objective, what's the point? They're just a nice data science projects that we can publish perhaps a paper on, but that's all. As data scientist, we have to be more practical.

First and foremostely I believe we're engineers. We're train to solve a problem in a practical way, and that's the mindset. The mindset of the engineer who tries to solve a problem effectively and without using too many resources. We're not trying to publish papers here. If you want to do that, there's plenty of room for that in academia, and that's great. We need papers. We need new research, but as data science professionals in the industry, we need to have the mindset of developing solutions and implementing them in an efficient and effective manner.

[0:29:30.9] SR: That's very true. You've already said that data science is a combination of statistics or math, let's say, and programming. What should people focus when they're starting to develop the mindset?

[0:29:43.9] ZV: I believe that we need to focus on both of these. You don't just do everything that is statistics-based or math-based and then go to programming. You have them parallel in a way. Just like in many universities, there are courses that do statistics and linear algebra and calculus at the same time, and there are also course that do programming.

The human mind is very agile. So we don't need to do one thing at a time. Of course, we don't need to overwhelm ourselves either, but when we're starting data science, I believe a best approach would be to do different things at the same time, so when they somehow relate to each other, not just how they stand on their own. I hope that answers your question.

[0:30:22.2] SR: Yes, it does. But what will you recommend as a first step? Because you have to then — If someone has to pick, then what should they pick?

[0:30:29.5] ZV: I think they shouldn't just pick one thing. They should do both at the same time, and there are many courses out there that do that, because statistics is a bit theoretical the way it is done. It is very practical when you apply it, but when you're learning it, it seems very theoretical. Unless you really understand how this things work in practice, they wouldn't make much sense, and in some cases they may be very boring too. Unless you understand what problems they're solving and how they're solving them and how they're adding value to the whole process, you won't appreciate them and you won't be so motivated to learn them.

The same with programming. Programming is great. It's very practical, but unless you link it to a particular problem you're trying to solve, unless you see the problem of how it is solved very efficiently with a program that you write, you won't really appreciate it either. You just think that it's just a bunch of techniques. It's much more than that, and every good programmer knows that.

The same with the science. When you do math and programming, you don't do them as isolated things. You do them in combination with each other and also in relation to the problem you're trying to solve.

[0:31:30.9] SR: I agree. I like how you orient the book towards making the reader think in terms of signal and noise. Can you explain how this is related to the mindset?

[0:31:39.7] ZV: Yes, that's a very good observation, although I didn't use the word signal and noise often in this book. I think it's a key point, and the signal is what you're trying to uncover in that sea of noise that you've given. Anything that is not a signal, it is a noise. Anything that

obstructs you from understanding what's happening is a noise, and it's inevitable that you find more noise than signal in most of the data out there nowadays.

Our work, as I was saying, this is to be able to create something, to build something that takes the data. It is very mixed and has some signal, some noise, but mostly noise, and bring out something that is more signal than anything else. Bring out something that everybody kind of understand and perhaps use directly. That's the key thing. It's like a transmutation of sorts. Not just transforming stuff, but also transmuting it. Changing the whole nature of the whole thing.

It's the same thing that happens when you have an efficient engine. For example, a car engine. Most of the engines nowadays, they use gas still. They take this mixture, this chemical mixture and they transform it and they transmute in a way and they make it into something useful, like kinetic energy and heat. Much of the heat we don't need, but sometimes all that is essential, because part of that is also charging the battery. WE have the ability to run some basic appliances in the car as well. All that is something useful, something that is closer to what we call signal in data science.

[0:33:17.4] SR: Most of the books that I usually come across talk on the methodologies explaining how data processes work. You've gone a step ahead and you talk of misconceptions that people have on the field. Broadly speaking, what misconceptions do people have on data science?

[0:33:35.5] ZV: Well, there are several, and what each person conceives wrongly is different. For someone who's more inclined in mathematics, they may have misconceptions related to programming. For someone's who's adept at programming, he or she may have misconceptions related to the math part of data science.

Each person has different misconceptions. What I do in this book is draw some general lines, some general trends in terms of misconceptions. The ones that they found more relevant to most people at least are — First of all, that the scientist can do anything with a data as long as they're given enough resources. That's a misconception. Data scientists can do some things and they may do great things based on the data they have based on the methods they use based on the resources they have, but if there's not enough signal in the data, there's not much

they can do. You can give some random data there that you have installed some minute signal in it and expect them to find it. They're not magicians, and that's a misconception any people have. They think that data science can solve all the problems out there and do that in a very efficient and cost-effective manner and they get disappointed sometimes, because the data scientist, even if they try their best and they do everything that's directly possible, they still don't get enough results. People needed to understand that, that data science is not a magic wand. It has major misconception that many people have across different areas, not just in the technical profession, but also in managerial professions as well.

Other misconception that I find very popular is that AI is the only thing that matters in data science, and many people are overwhelmed with how much value it offers. In some cases it does offer a lot of value. People who have come up with AI systems are brilliant and they have done a lot of work to make this thing scalable, but AI doesn't solve all the problems out there either. Unless you have certain kinds of data, and in many cases you have to have a lot of data. It won't add that much value. It may, but is it cost-effective? That's debatable.

In data science, we need to be more realistic about things. If you got solve a problem in a simple model, you might as well use that. We don't have to use the most fancy model out there just because it's there and we know how to use it. That's another misconception, and there are more and more that go on for a while, but I think these are the two major ones.

[SPONSOR MESSAGE]

[0:36:18.0] JM: For more than 30 years, DNS has been one of the fundamental protocols of the internet. Yet, despite its accepted importance, it has never quite gotten the due that it deserves. Today's dynamic applications, hybrid clouds and volatile internet, demand that you rethink the strategic value and importance of your DNS choices.

Oracle Dyn provides DNS that is as dynamic and intelligent as your applications. Dyn DNS gets your users to the right cloud service, the right CDN, or the right datacenter using intelligent response to steer traffic based on business policies as well as real time internet conditions, like the security and the performance of the network path.

Dyn maps all internet pathways every 24 seconds via more than 500 million traceroutes. This is the equivalent of seven light years of distance, or 1.7 billion times around the circumference of the earth. With over 10 years of experience supporting the likes of Netflix, Twitter, Zappos, Etsy, and Salesforce, Dyn can scale to meet the demand of the largest web applications.

Get started with a free 30-day trial for your application by going to dyn.com/sedaily. After the free trial, Dyn's developer plans start at just $7 a month for world-class DNS. Rethink DNS, go to dyn.com/sedaily to learn more and get your free trial of Dyn DNS.

[INTERVIEW CONTINUED]

[0:38:17.0] SR: You also have done some myth-busting in your book. You have made some important restrictions within statistics, BI, which is business intelligence and data science. Are they all not the same?

[0:38:30.4] ZV: I believe they're not, and it's not like they are completely different. There are mutual exclusive in any way. But statistics, specifically the statistics that a statistician does in practice, it revolve around certain kind of variables and they revolve around working with these variables using statistic tools.

A BI professional does that to some extent. They may not have the full breadth and depth of knowledge that a statistician, but they take that and they apply it into business problems and they work a lot with visuals as well, and that's great. So they add more value than just the models of statistics that a statistician uses.

A BI person is more focused towards a business, but they're not going to do much depth either, because they don't have that many predictive models. If they do have, it's not that elaborate as in data science, because in data science, sometimes you're giving some data that other people, particularly in statistics, they don't know how to work with data that is around text, for example, natural language text, or data that's around some sensor readings that are completely messed up and you can't really work with them in any meaningful way using statistics. A data scientist will go beyond the data and try to see the data from a different domain altogether. For example, sensorial data that is temporal data can be transformed into the frequency domain and be

analyzed as frequencies and develop features based on the frequencies. Sometimes combinations of the frequencies and the time in features, so create composite features that they're more sophisticated and capture more signal in that data.

A data scientist tries to solve a problem. It doesn't care where the data is coming from. It doesn't care about what distribution it follows. He does care about the distribution, but he doesn't feel limited by the distribution and say, "Okay. Well, data is like that and we have to use that approach only." No, they try to use different approaches and work with the data as is.

There's this distinction, for example, in data analytics between data-driven models and model-driven approaches. The first school of thought has to do with using data as is and not caring too much about the distribution it follows, not getting so much about the statistical models that may apply or may not apply. People like that usually go into AI approaches or machine learning approaches in general. People who focus towards the model-driven approach, they tend to see what models they can apply and try to solve the problem using some probability based method. This may work or may not work depending on the problem. But if you have a generic dataset that doesn't seem to comply to any particular set of distributions, then it's really hard to solve it with [inaudible 0:41:17.8] on methods of statistics. A BI person and a statistician would not be able to solve this properly, while the scientist might.

[0:41:25.4] SR: Are you talking about the two cultures paper when you're talking about the model and the data-driven culture, because that's a famous paper, I guess?

[0:41:33.8] ZV: I make reference to that, but I don't use that paper exclusively.

[0:41:39.9] SR: Okay. No problem.

[0:41:41.6] ZV: Because I think it's very useful to think about these things independently as well, not just focus on one approach to them based on a researcher. It's great to read papers, but it's good to also be able to understand things on your own too.

[0:41:56.1] SR: Sure. You make a very important distinction between data engineering and data science. Can you tell us the difference between both?

[0:42:05.8] ZV: First of all, data science includes the engineering. It's not an alien part of the field. Data science is data engineering, data modeling and other things, but these are the two main parts. Data engineering focuses more about the massaging of the data, the transformations of a dataset, the cleaning app of the dataset and a lot of ETL processes. Data engineering also focuses about acquiring data from different sources. All that is essential. However, the data engineer tends to focus more about the technology than the actual modeling. A data engineer might be adept at using a data governance system and maybe able to process data very well and store them very efficiently and retrieve them as well and do all sorts of data engineering tasks, but not do models well. There is a need for both data engineer and a data scientist.

Ideally, data science will have data engineering expertise as well, but nowadays we see this trend of specialization in data science. There are some people who are very good at modeling and nothing else and some people who are very good an engineering, but can't really make very decent model.

I'm not agreeing with this approach to this things, but I see that there's merit in it in some cases, like if some big company wants to hire 10 people in a data science team and they want to have four data engineers for example, those people have to be very good at what they're doing, like expert level. Beyond anything that the data scientist who is more well-rounded can do.

Of course, those data engineers will only do their engineering in that case in that team, but they would be working in parallel with data scientist, would do the data modeling part. So everybody wins. But in a smaller organization, this is not an option often, because the funding is limited. A data scientist who works there has to be more well-rounded and do both data engineering and data modeling and all sorts of data science tasks.

[0:43:59.2] SR: You also write about what data scientists do not do. I think that's a very unique perspective that you give. Most of the books talk about what data scientists do as supposed to talk of what they don't do. Can you talk on this a little bit?

[0:44:13.4] ZV: Yeah, of course. I think make three points in that particular section. First of all, if you're given some data that has very low veracity, it doesn't have much to offer, as in the sense you cannot make it talk. You cannot force a data to give a signal that it doesn't have, and that's something that people have problem understanding and that's part of the misconception idea that I was talking about earlier, because data is low veracity. You can't really do much with it. You can improve their signal. You can bring what it has there, but you can't expect miracles. That's something that data science cannot do. You need to work with other people as well, maybe enrich the data with using different data streams parallel to the ones you have. This way work that program. With a data that is low veracity, you can't do much.

Also, if you need to develop some kind of application based on the data model that data science develops, you can't expect that a data scientist would do that as well very well. They may be able to do something, but it may not be so professional. If you expect the data scientist to do a software engineering as well and software development, that's a bit unrealistic, because some people may do it, some people will not, and that's expected. Just like you can't have an athlete who is very good at different sports. It's very unusual for this to happen. Of course, there are some [inaudible 0:45:39.0] data scientists out there who can do data analysis and software development really well, but there are usually the exceptions. A data scientist does not develop software. They may be able to do something simpler, but not something super professional.

I'm thinking about another point about this, the tools. Yes. A data scientist does not develop new tools and does not develop new processes either. They usually apply the things that they already know and does so in a creative way. A data scientist is not the researcher always. Sometimes there's an overlap between the two. If you're doing, for example, in-depth data science in a company working at a research of that company or you're developing some new approach to things, usually the data you have. Then you may be doing some research there as well. Even if you don't publish papers. That's when you develop tools.

In most cases, you don't develop new tools. You don't develop new metrics. You don't develop anything new. Just use this stuff that is out there, but use them in a way that brings value to the organization. You just analyze the data it has and present the results and that's it. You can't expect every data scientist will develop new tools. Some of us do, and that's great, but you can't expect everyone, because that's not part of the job description always.

[0:47:01.7] SR: You're talking in depth on how to ask questions and use statistics to find an answer. You talk a little bit bout experiments and how the importance of hypothesis and how people can connect experiments with data as a way of tackling the problem.

Can you talk us through a problem? Let's say that I want to buy a home in the Bay Area. How can I use data science to solve the problem?

[0:47:26.9] ZV: That's a very big problem to solve. I don't know if I can do justice in a short interview, but let's say for example that you have a certain budget and some other limitations, like you want this to be up to three miles from the ocean. You don't want something that is in the area, but not close enough to the ocean, and you have a budget of, say, $2 million. I think that's a reasonable for Bay Area, right?

What you can do is survey, first of all, what's out there. Gather data, and by that I mean gather different things about different potential houses that you want to buy, and this can be different types of houses as well. Maybe you want to buy a condo or you can buy a whole house if possible in a different area perhaps and not so close to the ocean, or you can just buy some land there if it's available and build something there. That's also an option to consider perhaps depending on how much time you want to dedicate to this whole house thing, house project.

After you have gathered all the data, you can also try to expand datasets in a depth of time and see how the prices were in that particular area of that place a few years back and analyze that overtime and see how the trend is, because when you buy a house, you don't just buy it for the time being, you buy it for the future as well.

For some people, a house is also an asset. It's not always an asset, but you may buy a large enough apartment that you can rent out some rooms using Airbnb or something. In that case, it brings you income. You have to consider this aspect as well. When you put all that down, then you have a well-defined problem. The requirements of that problem may change overtime, but for the time being when you're trying to solve the problem, you have these requirements, you have to stick to them.

Now, in two years' time you may reconsider the whole thing, because things may have changed a lot. Things change a lot in the housing market, especially in areas where there's a lot of demand, a lot of movement. When you're trying to solve the problem, you do it with a certain horizon. Let's say for example for the next two years I want this house to be all right. I want this house to be the best choice possible.

If you're framing it like that, then you're basically solving an optimization problem and your limitations are the budget and the location. You try different options and see how each one of them performs. Maybe you device some kind of metric to see how good the place is. This can be something very custom, most likely it's going to be something custom, because you care about things in different ways. You may care that the place is very quiet more than the fact that the place is very central.

You put all these things done in some measurable way. For example, the quietness can be some scale of 1 to 10. Let's say the more quiet it is, the higher the metric of quietness, and you put these as part of that equation you're trying to optimize. Thus, once you do that for all the different factors you have to consider that they're important to you, then you solve the problem and try to find optimal solution and see how the solution pans out overtime, because things change overtime. Take that into account using that data from the past and see how this factors value change in the next two years.

For that two year period you can say, "Okay. This particular solution, A, works best, because over this two-year period, it has the best overall value. That's just a simple example, but depending on the data you have, you can do more sophisticated things as well.

[0:51:06.7] SR: I think the first problem here is the gathered data. I think that's the first step that people would get stumbled on.

Moving on, you make an important distinction between programming, bugs and mistakes. So I would think that bugs are mistakes, but are they not?

[0:51:23.3] ZV: They are mistakes in a way, but I make the distinction because I have to deal with these different kinds of mistakes in different ways. The bugs are mistakes that are easy to

identify. Usually, the program crashes if there's a bug, or if it doesn't crash, it'd give you some warnings of sorts in letters that you cannot really defy. Usually it is some red font as well, depending on the language always.

The bugs are crucial. You can't really solve a problem if it has bugs. The bugs need to be dealt with before you can complete the running of the program. These may be quite irksome and time consuming to deal with, but they are easier in general, because a mistake in the methodology, which is more high level kind of error is really hard to deal with. It may take a long time and sometimes you may not be able to solve it by yourself. It's good to differentiate between these two kinds of errors, because the high-level mistakes that are methodology related and you may get away with, and that's the worst thing, because the mistake is still there, you just — You haven't seen it. It's like you have some kind of infection in your body and it hasn't manifested the symptoms. It's still there and eventually will cause a problem. It's a matter of time. The problem may not be easy to deal with when it comes, but you should be able to deal with a problem before it manifests and that's where mistakes come in as something that you need to be concerned about. It's not like you don't ever finish a project because you always try to eradicate all the possible mistakes you have made, but if you finish a project fast and you have the time, it's good to think about where things may be wrong with the whole thing even if they don't give you an errors, if it don't give you any show stoppers.

[0:53:20.7] SR: You talk of mistakes and then there is the idea of using heuristics to reduce time. When you're trying to choose a heuristic, how do you stay away from mistakes when selecting heuristics?

[0:53:32.2] ZV: Well, that comes with experience to some extent, because if you know what has worked in the past in similar projects, then you're less likely to make a mistake when you choose a heuristic, but choosing a heuristic — Even if a heuristic is good, there's always a room for error. So it's good to have that mind.

Then mistakes you may make with a heuristic is that the heuristic may give you a value, but this value might not relate to the phenomenon that you're trying to mirror in that heuristic. These are things you need to take into account and the only way to deal with them effectively is through trial and error overtime, and sometimes you don't have the time to do it properly, but sometimes

good enough is just good enough, and that's why we have to do often times many iterations with the science process so that the mistakes it may have come about in the first iteration are corrected in the next one. As you do this whole thing again and again sometimes with new data available, then you refine the whole product more and more.

[0:54:36.6] SR: Then can you talk us through how to evaluate data then pairing it with a model? You have a bunch of data and you are trying to see which models fits it right. How do you evaluate the model?

[0:54:49.0] ZV: Evaluation of a model is a very long topic, and it has to do with kind of what are the performance and how long it takes and also how many resources it uses. If you take this into account, these three factors, that's how you evaluate the model. In different cases, you may have different ways for these factors. It's really hard to put them down in one formula and evaluate the whole thing, but often times we look at the performance of the model in terms of accuracy rate and error rate and things like that. That's a whole field by itself, because this has been studied very much over the years.

However, when you're evaluating a model, you have to take into account also how long it takes, because sometimes these extra 5% accuracy, for example, that a new model has may not be worth the extra time it takes to train and to test and sometimes the extra resources that these may use may not be worth the while either. There's always a tradeoff. The new systems that are more computationally expensive and they require more resources, they often times take more time as well, but they yield a better performance. You have to ask yourself, "Is that extra performance worth it?" And that's something that you have to be able to gauge beforehand ideally so that you don't go into the process of training a model for a week, maybe not a week, but depending on the resources you have, it may take you up to a week. Then realize that actually I don't really need that extra performance so much. You have to think about this in advance. You have to plan ahead. Measuring the performance of the model is one thing, but the overall effectiveness is a different thing sometimes. Someone who the science mindset right, they will be able to tell a difference and deal with it beforehand.

[0:56:37.5] SR: Perfect. You make specific mention on the right questions to ask. There is also a risk in asking the wrong question. What exactly is the risk and how does one know what's the right question?

[0:56:51.0] ZV: Well, the risk has to do with the questions themselves. If you ask something that's very generic that is impossible to answer with a single experiment, then you're asking a wrong question. You're asking a question that is interesting but may not be testable.

In data science we have to be more specific often times, but not too specific, because if you are answering a very specific question, it may take you a while to answer all of them. What I would recommend is somebody comes up with some research question in that project and then breaks it down. Then the questions they're to answer are more specific ones.

Then if those questions yield the results that are very useful, then you can break them down even more or do variants of them. I know this sounds very general, but if you think about the problem, you can actually come up with your own set of questions and potential answers and test those and that's where statistics especially comes in very handy, because part of the scientific process is testing hypothesis. Actually, the biggest part of a scientific process is testing hypothesis. If you can't formulate hypothesis based on a question, then it's really hard to test something scientifically, if not impossible.

[0:58:03.3] SR: I agree. When we think of programming languages related to data science, we think of R and Python. They are the two very huge programming languages, but you have a book on Julio, and you hold that Julia is best for data science. Can you tell us a littel bit about that?

[0:58:21.0] ZV: You want me to talk more about how Julia I believe is best or the different languages in general?

[0:58:26.5] SR: How Julia offers more than R and Python.

[0:58:29.4] ZV: First of all, we have to look at things in context. Julia is a great data science language and it has been from the very beginning, but it was in potential back then. Now it's

actually manifesting as a good data science language. Of course, there are lots of libraries, lots of Julia packages that solve specific data science problems. Before, it was just an idea that some people believe and some people didn't believe. Now, there's no doubt about it. That's not to say that other languages are not good anymore. That's the problem that many people have when they're comparing languages.

No. you can still like R. you can still like Python and you can still like some other language out there. You don't have to choose — In some cases you do. You can't do any language that's incompatible with another language. So if you want to go with Go for example, then you have to get all the other languages.

Maybe you can still work with C or Java, but if you go with Julia, you can still work with Python at the same time and R. There are packages out there in both R, Python and Julia that allow bridging among the different languages. You can still have a script in Julia and call it form you Python script and vice versa. You don't have to choose in a mutual exclusive manner, and this was like that from the very beginning, and people haven's realized that and some people say, "Okay. Well, this may work, but I don't want to go through the effort of learning their language," and that was the biggest impediment, I think, in the development of Julia language, because it's a bit different in the way it deals with data. It doesn't work with objects that much. It has the potential of doing objects and many people use it as an object-oriented program language, but it is a functional language. If you see that as a functional language, it has lots of potential not just in data science, but in different applications.

There are people who love it and people who don't want to hear about it, but those who have tried it and honestly have done an effort to learn some things and try things out, they're open to it. They may not use it every day, because maybe their day job, there is no room for something else. There's no room for experimentation, but those places where there is room for experimentation, they have embraced it very well. I think that shows something. It may still be in version 0.6.1, but it's already getting there. It's already performing well enough to be used in many real world scenarios.

I think it was last year when Julia computing and Microsoft made some kind of arrangement. Julia is available on the Microsoft cloud now. That is a lot. Remember that Microsoft is a

company that has its own programming languages, like C#, F# and other languages. Still, it is open to using Julia on its cloud. For me, that says a lot. If people choose to believe that their language of choice is the best one, that's fine. That's their opinion, but it's just an opinion.

[1:01:25.5] SR: You also talk about the emerging data science profiles, like the versatilist and the researcher. Can you talk about the different roles you see emerging as a vertical on their own?

[1:01:36.3] ZV: I believe these two roles that I mentioned there are very popular and more necessary than before. The data science researcher existed in the past but not so much. Nowadays, there are more and more people looking into different data science specific topics and they research them and they try different things and they publish some papers in some cases. Whether they publish or not, they're researching, and that's a very important thing to do. We may call them thinkers, not researchers in some cases, because they don't really understand everything that they're doing, but they have this research mindset, which is very useful, because this allows them to do something new. I think that's very admirable.

This trend is not going to slow down anytime soon, because nowadays it's easier than ever before to do this kind of work. It may not be as robust as academic research, but it is getting there. Those people may eventually do a Ph.D. and they will become more qualified researchers, but you don't have to have a Ph.D. to do research. That's the point. If you know what you're doing and you apply the scientific method, then you're doing research, but it is much, much easier if you do it through a university or a research center.

Now, the versatilist is an interesting concept, because I've been hearing about this for several years now not just in data science. The first appearance of a versatilist I believe was in the technological field, that's why it's more commonly seen in technology related professions, but it wasn't in data science. The first versatilist cases I have examined were people in the web development arena. This were people who were very good at different kind of web development. They were very good with handling PHP and also HTML and also CSS and also JavaScript and several other things in there at the same they could also write content or they could understand some things about design and they could do some designs. They will not do the perfect designs, but they would be able to create a good website from scratch by themselves.

That's very, very amazing if you think about it, because these are very diverse things in a way. They're related to each other, but it's rare that you find someone who's good at the backend and the frontend. In data science, it's very similar. It's very hard to find someone who's very good at modeling and data engineering and product development, data products. It's even harder to find someone who's good at all the technical stuff and at communicating stuff effectively and creating visuals, good visuals. I think that's something that is necessary, because if a smaller company wants to hire you as a data scientist, they can't really do much with you if you only do one thing well, and if they can hire a whole team, that's fine, then you can do your thing well and somebody else can do things you don't do well and they can do those things well themselves.

You can either have a team of specialists and that will work great, but in many cases you may not have this bandwidth to have all these resources, all these common resources. So you may have a couple of this and it's only — And they have to do everything themselves. In that case, a versatilist can shine.

Also, a versatilist who is in a team with data scientist can be a good team leader, because that person understands all aspects of the data science pipeline and can manage everything very well and very fairly, because in many cases you see data science leaders who are managers rather that they don't understand everything and sometimes they can't really manage a team very effectively. It's a whole can of worms opening if I were to go this in more depth, but let's just say that a versatilist is very agile as a resource. Can be used on her own and she can do different things in the business pipeline or she can even manage a team or both, because the varsatilist doesn't just do one thing. That's the key thing. They can lead a team and still write code or they can write code in one part of the pipeline and they can still do analysis on another part of the pipeline in the same day.

[1:05:41.8] SR: Agile definitely is a keyword that attracts a lot of eyes these days. So versatilist — I'm more tending towards the versatilist as opposed to the researcher, but I take encouragement when you say that you don't have to have Ph.D. to do research. I think that's a very important point, which often gets missed.

Lastly, on your — You have mentioned multiple times about mentoring. Most people recognize the importance of a mentor but find it hard to find a mentor who's willing to help. First, that's what made me take the podcast and books on tech so I can be updated on the tech scene. What are your thoughts on mentoring and finding a mentor?

[1:06:20.8] ZV: That's a very good question, because many people want or need a mentor, but they don't know what to do with it. I have written some articles about this as well on my blog, and the more I learn about mentoring, the more accessible it seems as something to do, because nowadays more than ever before, people are connected to each other. Maybe the connection is not super deep. It's more like a superficial professional connection out there, but that can be a good starting point.

If you really want to be a mentee, if you really want to learn and take someone else's experience into account in your professional devilment, because that's what mentoring is about. It's about professional development, and also development of professionalism, because this go hand in hand in my view. You don't just mentor with someone so that you can learn the knowhow. That's one of the things you learn, but the most important thing in my view is becoming a better professional, becoming a more a responsible individual in your work life and a more responsible resource for your manager or whoever uses you as a resource.

Mentoring teaches you all that, and finding a mentor is not easy, but if you're a good mentee, you're more likely to get someone. I have two solutions for that. If you're in Seattle, there's one tech mentor's group where people meet and sort of mentor each other, but there are some people who are more experienced than others, and these are usually the mentors in that setting, but this is more an informal situation. It's a good place to get a taster. If you want to do mentoring seriously in data science, the only place I know so far is Thinkful. It's an online company that does data science education. Part of the pipeline of the courses they offer involves mentoring. There are other places where you can actually find mentors while you do a course, but in those cases the mentors are just there. They may help a bit once or twice, but in Thinkful, it's every time. Every week you're expected to have at least two meetings with a mentor.

In smaller courses, like introductory courses, you may just need one, because you don't really need that much guidance. It's more about getting other things on board. In the data science courses, the courses of the company, you have to meet with mentors two or three times a week and get guidance on specific and general things, and that's what I really value about mentoring, is that a mentor can tell you specific things, like, "Okay. You have to change this in your cover letter. You have to change this in your code. You have to present yourself differently when you make a profile on a profession or a social medium.

They can also general things like, "Okay. If you want to become a very good data engineer, for example, if you want to improve your data engineering skills, these are the stacks that are best suited for you. The mentoring is a very personal thing as well. It's not generic things that you can read in a book or listen to in a podcast. These can be a first step if you can find a mentor to help you out in your regular work. These are the next best thing, having a good book or a good podcast. If you have a mentor, it's much better than anyone of these resources. That's something I think everybody can benefit from regardless of the stage they are in their career. Even I am learning new things, and it's useful to have a mentoring even if you don't meet very often or if you just exchange emails every now and then. That's better than nothing, because that keeps you grounded in a way.

It's also good for the mentor, because the mentor can easily get carried away with his or her work and stuff in general and may lose touch with how the rest of the people in the field are. Having mentees allows you to understand how the field is right now for people who enter it right now and to appreciate new things that you may not be aware of or appreciate new challenges that you were not around when you were there, because they were like footnotes and now they're serious things that people take into account when they learn data science. It's important to have both the mentee and the mentor in your career at one point at least. That really allows you to connect with a field in a different way, in a more in-depth way.

[1:10:47.1] SR: You also write about how or what professionals should do to remain relevant in the field. Can you talk a little bit on that?

[1:10:54.6] ZV: Sure, yeah. There are different things you can do to remain relevant and one of them is mentoring, of course. But there are other things like you can always educate yourself,

because even if you know the stuff that you need in your everyday work, there's always new things going on or new technologies or new systems or new developments in general. It's good to be aware of these things, because nobody is going to fire you if you don't know them, but it is possible that your work will improve in quality if you know these things. Maybe not in the next month, maybe not even in the next year, but gradually it will have some benefit, because learning new things is not just about expanding your knowledge, it's also about expanding your perspective and that's something that people forget, because the one thing that separates people who are successful from people who are not successful but talented is not so much the knowhow as the way they think. If somebody is exposed to knowledge a lot and new ideas and new ways of doing things, they naturally get more intelligent.

Intelligence is partly genetics, but it's also a lot of nurture. If somebody is exposed to new knowledge, new ideas, new things and think in different ways, they gradually become better of what they're doing, and that's something essential in data science, because the field enhances constantly and sometimes it's hard to keep up. But if you're always open to new things, you will always be relevant because you will know these things and you will know how to think in those new ways.

[1:12:29.4] SR: I agree that it's definitely a field that constantly changes. Well, it was great speaking with you. Now, to wrap up the interview, do you have any shout outs, like any books or courses, videos or podcast that you listen to related to data science?

[1:12:45.2] ZV: There are a couple of things. First of all, all my books from Techniques Publications are worth checking out. The Thinkful Course of data science, both introductory one and the core one are very good as a resource if you want to learn data science and you have just some minor exposure to it already. Other things are — There's this very good course from the University of Washington, if you're in the area of Seattle. It's worth checking out. It's Gear for Professionals, so you can still do the course while you do your day job. There are other things as well, like I think it was Joel Grus who has a very interesting podcast. Worth checking out. He's a data scientist/AI professional. He has some very interesting views of the field. His whole style is very easy to follow. He's very entertaining at times. At the same time he's very serious about the things he talks about. It's definitely worth checking out his stuff.

Also it's good to always be on the lookout because I know a few things, you know few things and someone else may know a few things about data science that are good as resources, but it's always a good idea to keep an open mind about what is out there that nobody else knows. This curiosity is essential for sure in data science and it applies in these as well.

[1:14:08.4] SR: Now, for your page on the importance of developing a mindset.

[1:14:11.2] ZV: Right. I'm going to paraphrase this quote from this guy, Ian Malcolm, from The Jurassic Park. It's heavily paraphrased, so you may not recognize it from the movie, but I really like how he says that we should stop and wander not just about whether we could do something or not, but also whether we should do it. That I think summarizes the mindset of a scientist in general and that applies in data science as well, because just because we can't do, for example, an advanced AI model on a particular dataset, it doesn't mean that we should. Just because we could some simple very easy thing as well, it doesn't mean that we should not do that either.

We ought to think about whether we should do this or the other method. Whether we should do this or the other approach before we do anything regardless of what we can or we cannot do.

[1:15:00.9] SR: Fantastic. Thank you, Zach. It was great talking with you. Thanks for coming on the show. Folks, check out the latest book written by Zach, it's Data Science Mindset Methodologies and Misconception. Thank you, Zach. It was great.

[1:15:13.0] ZV: Thank you, Sid. It was great for me as well.

[END OF INTERVIEW]

SparkPost has a range of pricing options, from free self-service packages to sophisticated enterprise support and services. Start sending emails to your users today. Go to pages.sparkpost.com/sedaily to send 100,000 emails a month for free.

Thanks to SparkPost for being a new sponsor of Software Engineering Daily. If you want to send 100,000 emails a month for free, go to pages.sparkpost.com/sedaily.

[END]