

**SDS PODCAST
EPISODE 337:
HADLEY WICKHAM
TALKS
INTEGRATION
AND FUTURE
OF R AND PYTHON**



- Kirill Eremenko: This is episode number 337 with Chief Scientist at RStudio, Hadley Wickham.
- Kirill Eremenko: Welcome to the SuperDataScience Podcast. My name is Kirill Eremenko, Data Science Coach and Lifestyle Entrepreneur. And each week we bring you inspiring people and ideas to help you build your successful career in data science. Thanks for being here today and now let's make the complex simple.
- Kirill Eremenko: This episode is brought to you by SuperDataScience, our online membership platform for learning data science at any level. We've got over two and a half thousand video tutorials, over 200 hours of content and 30 plus courses with new courses being added on average once per month. So all of that and more you get as part of your membership at SuperDataScience, so don't hold off. Sign up today at www.superdatascience.com. Secure your membership and take your data science skills to the next level.
- Kirill Eremenko: Welcome back to the SuperDataScience Podcast everybody, super excited to have you back here on the show. Today, we have none other but the legendary Hadley Wickham. This is a person who doesn't need much introduction. He's the author of ggplot2, of dplyr, of tidyverse, of many, many, many R packages. He's a professor. He recently received the COPSS award, which is a very prestigious award. It's the equivalent of the Nobel Prize for statisticians. It's the first time in history it's been awarded not for theoretical development in statistics, but actually for software development.



- Kirill Eremenko: This is a person with tens of thousands of followers online who's written multiple books, makes appearances at conferences and runs presentations. Hadley is one of the key people driving RStudio and R programming language forward. So very, very excited. I was very excited to talk to Hadley and we covered off a lot of topics. So we talked about packages in R and how they compare to Python and specifically we talked about the differences between R and Python. I learned quite a lot of new things for myself. Production development in R, looking at things from a fresh mindset, different conferences. We talked about the useR Conference and the RStudio conference, and then I actually posted on LinkedIn a request for questions for Hadley and quite a lot of questions came in, so I asked them and you will get to hear not just what I'm interested in learning from Hadley, but also what your peers, other fellow data scientists listening to this show are interested in hearing from Hadley.
- Kirill Eremenko: So you will get answers to a lot of those questions which are diverse, ranging from the questions about the future to career questions, to more technical questions, to community questions. Well, to sum it up, it was a lot of fun having Hadley on the podcast, I learned a lot and I'm sure you will learn a lot too from one of the most influential people in data science right now. So without further ado, I bring to you, Chief Scientist at RStudio Hadley Wickham.
- Kirill Eremenko: Welcome back to the SuperDataScience Podcast, ladies and gentlemen, super excited to have you on the show.

And today I have a legendary guest, Mr Hadley Wickham is with us today. Hadley, how are you going?

Hadley Wickham: Good thanks. How are you?

Kirill Eremenko: I'm very good, very good. And you are in Houston today? When was the last time you went to New Zealand?

Hadley Wickham: I was just there in... No, I can't even remember what month it was. In early December.

Kirill Eremenko: Early December. That's really cool. I was there end of November, early December as well. I've got to say, I love your country. It is just the most beautiful place, especially out of... Like last year I did North Island at the start of the year and South Island at the end, they're both beautiful. North Island is by far my favorite. It's just incredible. How come is it so beautiful all the time?

Hadley Wickham: Yeah, it is beautiful. But the downside is it's just so far away from the rest of the world.

Kirill Eremenko: Yeah. Yeah. I guess maybe that's the trade-off that you have, but it's so neat. Like driving from Oakland down to Hobbiton and from Hobbiton down to Rotorua. It's just like everything is lined up. Every single bush, every single tree is in a line. It's just incredible, and the hills. Have you been to Hobbiton yourself?

Hadley Wickham: I have not. It's actually pretty... I grew up in Hamilton, so it's actually fairly close to where my parents live. But I have not visited.

Kirill Eremenko: Are you keeping it for later?

Hadley Wickham: Possibly never.

Kirill Eremenko: Oh, okay. Gotcha. Well, just wanted to mention that it's really, really cool country and if anybody listening hasn't been to New Zealand, highly recommend. Very exciting. But you are now in America. How long have you been in America for?

Hadley Wickham: I think it's coming up to 15 years.

Kirill Eremenko: 15 years. Wow. So since you went to do your PhD there, you stayed there.

Hadley Wickham: Yeah, that's right.

Kirill Eremenko: And how do you like it there?

Hadley Wickham: Yeah, I don't know. It's kind of like home. Now I feel like I'm sort of a New Zealand/Texan now. I became a citizen two years ago, so.

Kirill Eremenko: Wow. Congrats.

Hadley Wickham: Kind of made a life here. I miss New Zealand, but now when I go back to New Zealand there's things about here I miss as well.

Kirill Eremenko: Gotcha. And so you've moved around the US quite a bit, haven't you?

Hadley Wickham: Not too much. I lived in Iowa for my PhD and then I've been in Houston for the last 10, 11 years.

Kirill Eremenko: Okay. Gotcha, I was listening to a podcast with you recently, and it was interesting to find out that ggplot2 actually came out of... It wasn't the main reason for your PhD. It was just a side effect of your PhD, and

then you switched your PhD to work in that. Is that the correct story?

Hadley Wickham: Yeah, so I mean, I wouldn't say switched so much, it was sort of something I discovered on the course of my PhD. So, the funding for my PhD was I did a consulting assistantship, which means that I would help PhD students from other departments do their statistics, and in the course of that, it just really brought home to me that often the modeling part of the problem was often that... Just felt like the easy bit you did the end after you've done all of this data munging and done a bunch of visualizations to figure out what's actually going on. Then you could do the model, but it just felt like that modeling wasn't the hard part, which is really weird to me because that's what I'd always been taught in all my classes that modeling was hard.

Kirill Eremenko: Oh yeah. I think that's the case with most data scientists listening to this. We spend 70 to 80%, at least from my experience and from some of our students' experiences, that we spend 70 to 80% of our time just preparing the data. Why do you think that is the case?

Hadley Wickham: I don't know. Part of it I think is just that most data is not collected for the purpose of analysis. And so a lot of the time you are analyzing data that's been collected for some other purpose, for some other set of constraints. It wasn't collected to make your life easy. It's collected for some other purpose, and now you're trying to get some other value out of it. And I think, that just means part of the processes is just getting to

grips with how the data is and figuring out how to get into the form that would be most useful for you.

Kirill Eremenko: So you have two creations there in that space you have ggplot2, at least two, and you have the whole tidyverse and also dplyr. So in what sequence did you come up with these? Because they all addressed the same issue. ggplot2 allows you to visually see the data and helps you explore and see any outliers of patterns in advance. dplyr allows you to help you work with the data actually better and put it especially into.. Link it up with structured data sources and then finally the tidyverse is just a whole collection of things. In which order did they come about?

Hadley Wickham: Kind of a little backwards. I mean I think, ggplot2 was really my first major package. I'd used the visualization tools in R before, which are pretty good, but I found them... There were just a few things that I found really hard to do. And so then I worked on ggplot2 and it made those hard things much less difficult and then a new set of things bubbles up to being more challenging. I think one of those things that was particularly challenging was getting the data in the right format, this idea of tidy data, which I can now explain really easily. You want the columns in your data set to be variables.

Hadley Wickham: I think that's really natural to you if you're a statistician or a data scientist, but it's not something that people born knowing. So a lot of data you'll get is in some other format. You'll look at it and you'll be blown away by how crazy their format is, but it makes sense to the person who collected it and you've got to

get it into that form. So then I spent this time working on Reshape, which then reshaped into an entirely other [inaudible 00:11:22] with that.

Kirill Eremenko: But can use an example data where columns are variables how can that not be the case?

Hadley Wickham: So one dataset, I still vividly remember from my time when I was [inaudible 00:11:40] the columns were actually days. So each column was one day when the PhD student had gone out into the field and recorded something. And so then the column headers are January 1st, February 1st, March 1st. That's a fine way to record the data, but it's really, really difficult to analyze it that way. Because you want one column that's date you want another column that's a thing you actually measured.

Kirill Eremenko: Yeah. Gotcha. So this is kind of like an unpivoted view would be human friendly but not machine friendly.

Hadley Wickham: Yeah, exactly. Or like recording friendly, but not analysis friendly.

Kirill Eremenko: Oh yeah, yeah, yeah. Gotcha. So that was the... Oh, I can see how that would be extremely useful, in the case of research because as you just pointed out that it's easy to record that way. It's much harder to record the other way.

Hadley Wickham: Yeah. I recently gave a talk about this in Australia and so I actually went back to that. I remembered this. So I went back and looked at the code. I don't know, I maybe wrote 50 or 60 lines of R code, like a bunch of functions, a bunch of for-loops. It was a real sort of

programming challenge to get the data in the right format and then I rewrote it using some of the tools I've been working on lately out of the tidyr package. It's six lines of R code now. I think it's not just the number of lines of code, but it's their mindset. It's not a programming problem anymore. It's now a data science problem. When you're thinking about how do I pivot this data into the form that I actually want.

Kirill Eremenko: Interesting. Yeah, and on one of your talks, you also mentioned that SQL hasn't been changed in 30 years, and so even though it's very broad and powerful, the amount of data we have now and the veracity of data, it should inspire different ways of thinking about it. Is that something behind your dplyr package?

Hadley Wickham: Yeah, absolutely. I mean SQL is an amazing language and the fact that it has... I guess it must be coming up on what 40 year plus years old and it's still being used by hundreds of thousands of people. It's incredible. And I don't know, I still feel it's very arrogant of me to claim that dplyr maybe might be better than SQL in some ways. But I think it is, because it's trying to solve a much, much smaller problem than SQL is trying to solve.

Hadley Wickham: Where SQL the goal is to be able to handle very high throughput of data captured reliably and handle all sorts of problems. Whereas I think data science, the problems of data science or at least the problems that I think about the most, or a little bit, a little bit simpler. You've just got maybe one to five different tables of data. You've got maybe 20 variables. You might have hundreds of thousands and millions of observations.

So your data is very long, but it's typically quite narrow and it's not changing that often. Maybe it's changing every hour, every day, but it's not changing every millisecond.

Hadley Wickham: So things like that, fundamentals change. I think you can rethink the language and the interface, and of course, we've learned a bunch about programming and programming languages and the 40 years since SQL has been around. So I think there's some really nice things about dplyr that just make life a little bit more pleasant.

Kirill Eremenko: Gotcha. Speaking of data science, how would you define data science? Curious to get your thoughts on that.

Hadley Wickham: I mean my definition is data science is like data analysis by programming. Which of course begs the question of what data analysis is, and so I think of data analysis as really any activity where the input is data and the output is understanding or knowledge or insights. So I think of that pretty broadly. And then to do data science you're not doing it by pointing and clicking. You're doing it by writing some code in a programming language.

Kirill Eremenko: Interesting. Interesting. I guess our definitions would differ on that a little bit, because for me data science is for instance somebody who can't program and is just really good at communicating insights to business decision makers or government decision makers, I would call that person also a data scientist. But in your definition I would be more of an analyst.

- Hadley Wickham: Yeah, I think it's not like being an analyst is not a bad thing. It's using a different set of tools and that's a really important set of tools and the ability to communicate what you've discovered incredibly, incredibly valuable. I just wouldn't call that person a data scientist in my personal definition for whatever that means.
- Kirill Eremenko: Okay. Okay. Gotcha. All right. Very, very cool. And speaking of dplyr, I wanted to ask you, I'm guessing the alternative in Python would be pandas, any comments on how the two compare?
- Hadley Wickham: Yeah, I think one of the things that's interesting about Python versus R as a language is that because of the design of the language of Python, there's this subtle pressures to have larger more monolithic packages. So in some ways pandas in Python is actually equivalent to dplyr or tidyr and readr and a handful of other packages in R. It's a much, much bigger total. And you can see the same thing with scikit-learn. You've got scikit-learn, which is equivalent to maybe 10 or 20 different modeling packages in R, and I think it's interesting because it's both a strength and a weakness. It's great to have these sort of single artifacts that have a unified vision.
- Hadley Wickham: They can be much more consistent internally, but it's harder to grow them over time. It's harder for people to contribute just pieces of functionality or experiment. So I don't know. It's just one of the things I find really interesting about the differences between Python and R, and the other thing I think is really interesting is the Julia community. Julia as a language, in many

ways, is more similar to R than to Python. And there's a really nice talk, I forget the name of it, at JuliaCon about this idea that there's actually a surprising... When you're used to languages like Python, it's a surprisingly high amount of code reuse in Julia because of the way the object [inaudible 00:18:52] programming is designed, which makes it much easier to reuse code across package boundaries, which I think is really, really interesting.

Kirill Eremenko: That's in Julia.

Hadley Wickham: Yeah. And then in R, it's the same. It uses a style of... I mean the basic differences in most object oriented languages, like Python methods belong to classes or to objects, but in R and Julia methods belong to functions or generic functions. It just seems like such a subtle distinction, but somehow that makes it much, much easier to share code across packages, because one package can provide the definition, the interface of a function in many, many different packages can provide implementations for different types of objects. And that seems to be a really good fit for data science somehow.

Kirill Eremenko: Interesting. But in Python couldn't you just take an object and have a dummy object with just the function inside and use that to define it the same way as you do now?

Hadley Wickham: Yeah you can work around it. It's not like this is something that's impossible to do, just what parts easier and what's harder in the language and it's just a little bit higher friction and then you've got to use this

inversion of control type techniques to [crosstalk 00:20:13].

- Kirill Eremenko: Interesting. Wow, I didn't know that. That's very, very cool core difference that then goes into various things. I'm just curious how often do you use Python yourself? Everybody knows you're one of the most famous R users on the planet. But how often do you open Python?
- Hadley Wickham: I mean basically never. I never write code in Python. I try and read a moderate amount. I'm always looking to see what's happening on the Python, and how people are expressing themselves just to see what's going on, what ideas can we take and what ideas can I steal. But it's not as if I really dislike Python and I like to keep on... It's not like I'm following every detail, but just keeping an eye on Python and Julia and what's going on in Rust and what are all the hot new exciting programming languages people are excited about at the moment.
- Kirill Eremenko: Yeah. It's interesting that I also heard you saying one of your, I guess, dreams or wishes for the next five years is to find better integrations between R and Python. Could you talk a bit about that? What's the purpose of bringing these two languages closer together?
- Hadley Wickham: Yeah. So I think one of the things that's interesting to me about R and Python is that the way you write really, really fast Python code is basically the same way you write really fast R code. You just write C codes. But that's a little bit of a simplification. But I

think a lot of the really high performance computing in both R and Python is implemented in C. And both R and Python have really good tools to talk with C and if that's the case, why not team up? Sure, we maybe want to work with... R programmers and Python programmers see the way a program's going to interact and compose a little bit differently. But if the underlying engine is the same, it just seems to make sense to share that effort.

Hadley Wickham: That's one of the ideas behind the Arrow project, which McKinney's working on. Let's team up. Let's put a bunch of thought into the underlying design of the memories of the data structures and the memory of the CNC++ code and then let's provide interfaces for R and Python so you use whatever makes you most effective as a data scientist.

Kirill Eremenko: Actually somebody mentioned this so I posted questions on LinkedIn that we're going to have this interview and quite a few students, over a dozen students posted questions for you and one of them was actually about that. That you're working with URSA Labs on this Apache Arrow project. How's that going so far?

Hadley Wickham: So far I have not been doing much work on it. But that's one of the things I have planned for this January is to start working with the URSA Labs team on an Arrow backend for dplyr, because one of the things that I think is particularly neat about dplyr that separates out the interface of the way you described the operations that you want to do to the data set from the actual implementation. So dplyr has this native R

backend that works on data frames. It's got a backend that translates dplyr code to use data table, which is another really fantastic data manipulation package in R, and it also can convert your R coding into SQL code, so you can work with the database.

Hadley Wickham: But so the next step is to do the same thing for Arrow so you can write dplyr code, the same dplyr code you're used to. It gets translated into [inaudible 00:24:16] to Arrow and then that works on this shared memory data structure where the data could be... You could be working simultaneously potentially and eventually on the same data set in R and in Python at the same time.

Kirill Eremenko: But wouldn't the whole notion that in R you have just the whole vectorized structure of R, wouldn't that get in the way of ultimately integrating the two languages?

Hadley Wickham: I think for most data scientists, most of the tasks the data scientists do, that kind of vectorization actually helps you, because you end up writing higher level statements of intent. And that I think that's generally easier to optimize into or translate into some other language. When you are working with low level for-loops. For-loops are a very, very general tool. So you've got to apply a lot more thinking and smarts to be able to translate for-loops into something that is really efficient and in another language. Whereas when you're working with this vectorized operations with sums and ranks, there's maybe, I don't know, 30 or 40 of these vectorized operations that allow you to solve, I don't know, 90%, 95% of the challenges you face as a data scientist. And so I think this works really well for

data science. It doesn't, I think, work well as a general programming tool, but for data science somehow I think that just this idea of vectorized functions matches the problem well enough that it works out pretty well.

Kirill Eremenko: Okay. Okay. Yeah. I agree with you. I think that in many cases the vectorization of R can be beneficial, more beneficial to data science specifically. However, I'm curious, how do you reconcile that with Python that doesn't have that vectorization? Is it a major roadblock in this project that you're undertaking?

Hadley Wickham: I think so. It's basically, because it's easier to translate vectorized to non-vectorized code because basically all we have to do is add a for-loop. But going in the opposite direction is much, much harder. Like if I wanted to translate Python code into that R equivalent, I think that would be really challenging just because for-loops is just so general it would be very challenging to implement it to translate them back into the equivalent efficient R code. But going from R to Python is much simpler because you tend to have higher level expressions in R.

Kirill Eremenko: Gotcha. And in one interview you actually said that you see a company in the following way that the data science team uses R and the data engineering team uses Python. And we're going into more and more of a world where I even... Oh, by the way I mentioned this before the podcast, but I want to say it again for everyone who is not aware, congratulations on the COPSS prize. That's a huge, huge accomplishment. How did you feel about that?

- Hadley Wickham: I mean that was fantastic. I mean obviously it was a great recognition of my work. But I think the thing that was particularly exciting was that I'm the first non-theoretical statistician to win it. If you look at the previous winners they've all contributed to statistics by proving theories basically. And it's very clear that that is not what I do. So I think it was a really neat signal from the statistics community that programming and data science is important and it is really the core part of statistics too.
- Kirill Eremenko: Yeah. And I've heard it's an equivalent of the Nobel Prize for statistics. So huge accomplishment. And what I was going to actually say is that I've heard you say before that it's an interesting shift that the COPSS prize was given not for theoretical development and statistics, but rather than that it was software and product development in programming. And so the question I had was, you see R being used by the data science team, Python more by the data engineering team, how can we actually use R for developing software developing products, or is it purely going to stay as an analytical tool?
- Hadley Wickham: So I think, you definitely can use R to develop products, and people do. I think you mostly see this split. Not due so much to the fundamental differences between R and Python as programming languages, but more in terms of the background of people involved and where the communities have spent their effort over the last 20 something years. So people with existing expertise, and DevOps coming now to apply their skills to data science, they already know Python

and they want to keep using it. I think a lot of it's as simple as that. And so part of the reason that Python feels really natural in production is that so many people have already put it in production.

- Hadley Wickham: There's a lot of existing knowledge in the community. No one ever got fired for using Python. Thinking, I'm sure people have. But it seems a sight language now. Whereas, R, lots of companies are using in production now, but still the understanding and the knowledge is not percolated out into the community so much. And that's something we really think a lot about at RStudio, like how can we help people to put their R code into production more robustly.
- Kirill Eremenko: So what's the plan? How do you think you'll tackle that?
- Hadley Wickham: There's a few different teams working on this. So one way we tackle this is RStudio makes money by selling software. And some of the software we sell is, or one of the tools we sell is called RStudio Connect and basically that just makes it really, really easy to deploy R code so it runs robustly in the same environment day after day, after day.
- Hadley Wickham: One of the ways my team is working on that problem is, I think, there's a switch in mindset from going from analysis and exploration to production that you have to stop thinking about this very general what the heck is going on with this data and how do I uncover that important signal as quickly as possible, to thinking about how do I write code that is going to work day in, day out for the next couple of years. And I think

there's mind shift. There's definitely different languages... Features of the language naturally help you think in one of those mindsets, but also whatever language you're in, I think acknowledging that there's different techniques, different approaches that you want to tackle.

Hadley Wickham: And so, one of the things that my team thinks about is, how do we help R users who typically don't have a software engineering background, how do we give them the key skills of software engineering, how do we help them learn about whether it's peer programming or source code control or test driven development. How do we give them the key tools of a software developer? They're never going to become the best software engineer in the world, but how can we give them the key tools to start thinking about writing robust production code.

Kirill Eremenko: And would you say that's an important skill to have for a data scientist?

Hadley Wickham: I think so. I think you're always better off getting really good at one thing and then expanding your skill set to become better and better at other things. Rather than being mediocre at a bunch of things. So I don't think you need to feel bad if you don't have a bunch of software engineering skills. But I think that is something that over time, if you develop those skills that really increases your impact as a data scientist.

Kirill Eremenko: Okay. Yeah, gotcha, gotcha.

Hadley Wickham: [crosstalk 00:33:39] Improving your communication skills does as well.

- Kirill Eremenko: Yeah. It's like we live in a world where analysis allows you to extract insane insights, but at the same time, I would say software development is a skill that allows you to build leverage, so that then you can impact, not just one company or one organization or a small group of people, but you can scale your impact to hundreds of thousands and millions of people. If you know how to write good software that is going to be used worldwide or is going to just going to keep working in the background and can scale that's how you scale your impact.
- Hadley Wickham: Yeah, absolutely. And I think that's something I tell people in academia as well. If you really want to have an impact on the world, I think writing high quality software that people actually use, that is just as impactful, if not more impactful, than writing papers that get a ton of citations.
- Kirill Eremenko: Yeah, yeah, totally agree. As I mentioned before, I posted the question on the LinkedIn for people to ask you questions and we've got quite a few come in. Would you to go through them and just do a rapid fire? All right, so Jennifer Cooper asks, "If faced with a choice, why should someone choose R over Python?"
- Hadley Wickham: So I think R, it's an obvious choice. If you've never programmed before, I think you can learn data science in R and then you could learn how to program in R. And I think the other reason to choose R, fantastic, fantastic community online, bunch of people really excited that you're learning R and happy to help you out.

- Kirill Eremenko: Fantastic. Great answer. Another one from Jennifer Cooper. What does the future look for coding languages like R given the rise of automated ML and drag and drop tools?
- Hadley Wickham: I think they're going to remain strong. I am pretty skeptical about drag and drop tools, because the hard part about programming is not typing. The hard part is not that you're typing words rather than dragging things and connecting with the lines. It's figuring out what those connections should be and programming languages just give you this fantastic set of tools for sharing and critiquing that you just cannot get with drag and drop tools.
- Kirill Eremenko: But wouldn't you say drag and drop is just a faster way to get insights?
- Hadley Wickham: I mean the other problem with drag and drop tools is you're fundamentally constrained by the author of that tool. You can only do the things that they want to be easy. Whereas with a programming language some things are easier, some things are harder, but you're never fundamentally stopped from doing something. So I think in any drag and drop, in any kind of [inaudible 00:36:40] tool, you'll always get to a point where you're like, "Oh, I'd really like to be able to do this thing that really makes sense for my analysis and there's not that widget." So you're stuck.
- Kirill Eremenko: That's true. I'm learning right now, very advanced level of Tableau and getting to that widget, you can get there, but do you have to know extremely advanced features and concepts whereas in programming, as

long as you know how to program it, you know you're going to get there eventually somehow.

Hadley Wickham: And, and then the other great thing about programming is once you've solved it for today's data sets, you can apply it to tomorrow's dataset just as easily.

Kirill Eremenko: Yeah. And then you can turn it into a package and then release ggplot2.

Hadley Wickham: Exactly.

Kirill Eremenko: Something like that. Okay. Another third one from Jennifer Cooper. What is your advice to someone learning R who may be overwhelmed by all the syntax libraries and modeling techniques? Any tips, tricks, shortcuts to remembering it all?

Hadley Wickham: I think some of it you just have to accept. You're not going to remember it all and that's fine. It's just like learning a new human language. It takes a while before you can become fluent and there's no way around that. It happens to everyone. So don't feel despondent. Don't blame yourself that you're too dumb to remember this. Absolutely everyone has to go through that. I think doing some structured practice can help. Doing sort of flashcards stuff where you just practice that one aspect of recall can be valuable. I think the other thing that can be really valuable is find some people to walk down the road with you, so you've got friends, you've got colleagues who are struggling the same as you who can commiserate you when things are going badly and celebrate you when things are great.

- Kirill Eremenko: Yeah. Yeah. I agree. It's a long journey, but it's worth getting there. All right question from Morgan Mendis, an advanced data scientist who's actually been on the podcast just recently, what is your preferred method of multidimensional analysis?
- Hadley Wickham: Oh, I don't know if I have a preferred method of multi-dimensional analysis. I mean, this is a little bit of a glib answer, but I always start with visualization just to get a sense of what is actually going on with this data, because I think if the first five discoveries of your data analysis project are not data quality problems, that just means there's data quality problems you have not discovered. So really figuring out what the heck is going on with the data first before you do any formal modeling.
- Kirill Eremenko: Okay, gotcha. You wouldn't to jump into dimensionality reduction before discovering it?
- Hadley Wickham: Yeah, absolutely not.
- Kirill Eremenko: What if you had so many dimensions it's just really hard to even guess where to start visualizing?
- Hadley Wickham: Yeah. I mean that's basically a problem that I do not have. So I do not have any good advice. I think in their case dimensionality reduction can be really useful. You just have to be a little skeptical. Take it iteratively. Do some reductions. Look for the weird points. Trace them back to their original variables. Have those variables make sense. Are they looking really weird because NA are being stored as -999? That's the sort of stuff you need to be thinking about very early on.

- Kirill Eremenko: Gotcha. Why did you say you don't usually have that problem?
- Hadley Wickham: Just because the type of problems I normally work with. I'm not a data scientist anymore really. I'm someone who developed tools for data science. So I'm mostly playing around with datasets that are interesting to me, which tend to have maybe 10 or 20 or a 100 variables, but not thousands or tens of thousands.
- Kirill Eremenko: Interesting. I was actually going to ask you that. It's a question that came to my mind. I was listening to a podcast with you and then reading interview with you and I just thought you just constantly coming up with these new things. Like now, you're working on dtplyr, so not just dplyr, but dtplyr, which sounds like a super exciting project. You have to have a different mindset. I don't know, I could maybe randomly come up with one of these ideas, but unless I see the programming language and look at R from a completely different perspective to what normally people see in it, I wouldn't be able to keep coming up and improving these ideas. And no wonder you got this prize and the wonder you're so recognized, is there a secret? How do you do this?
- Hadley Wickham: I don't know. I think a part of it is I do have a terrible... My long-term memory for things that I've done is quite surprisingly terrible. And so that means that I can attack the same problem with a fresh view because I've forgotten what the heck I did last time. Sometimes it's embarrassing, like I asked the same question again and again and again for a two year interval between it.

But I think somehow it's part of that and just trying to... I think one of the challenges is, how do you avoid becoming trapped by your success? How do you make sure... The things that you've done in the past that have made you successful, you can't just keep doing them again and again and again and I hope to continue to be successful.

Hadley Wickham: It's the sort of model retraining thing, right? You can't just fit a model and then expect that model to keep on working year after year, after year. It's somehow you've got to think what has changed in the world since I last tackled this problem? How can I come at it with a fresh mindset and maybe tackle it in a new way? But you know that's also very vague. I don't know if I have any [crosstalk 00:42:53]

Kirill Eremenko: No, that's really good. I love that answer. Now I have something to tell my girlfriend next time she says you keep forgetting everything. I'll say, well I'm just following Hadley's advice. Oh that's awesome. But I get your point. Would you say that is a useful skill to look at things from a fresh perspective even though it was going to put you behind in terms of how quickly you can address a problem? But do you think that would be useful for data scientists as well?

Hadley Wickham: Oh absolutely. I mean it's this balance. You don't want to be doing this all of the time. There's always this sort of balance I think of being successful in the short term and being successful in the long term. If you just optimize for being successful in the long term, doing the thing that your boss wants you to do by tomorrow, in the long run you're not successful. But if all you do

is think, well where do I want to be in five years' time? I want to focus on that. I just want to be learning the stuff that's not going to pay off for two or three years. If you do that, you fail in the short term because you've lost your job, because you've missed all these important deadlines. So getting that balance right. But I think it's really important to carve out time where you're not just solving today's problem. You're thinking about trying to take a step back and saying, "Well, how could I be doing everything that I'm doing more efficiently?"

Kirill Eremenko: Interesting. You speak of balance. Would you say you struck that balance or would you say you went in the extreme to the other side of the spectrum where you're just thinking about long-term problems all the time and that's what helps you stand out and really contribute to the world?

Hadley Wickham: I think I've always been fairly long term focused, but at the same time, I guess that was sort of one of my worries leaving academia. Where I think one of the nice things about teaching a class is every year you're teaching the same thing to a new bunch of students. So you get this reset button push every year and you've got to start from scratch again. You can't get lost in the clouds. I think [crosstalk 00:44:58]

Kirill Eremenko: In the clouds.

Hadley Wickham: Leaving academia into this environment where if I wanted to, all I could do is focus on, what's going to be really important in a year or twos time. But I think the thing that really pulls me back to earth now is

interacting with people on Twitter who are like... People tell me pretty frankly all the time. Well not all the time, but some of the time. When I create something that's too complicated that people don't understand, something that makes perfect sense to me because I'm embedded in it, all I'm doing, I'm thinking 60 hours a week about R, and how to express my ideas in R. I still get this feedback, it doesn't make sense to me. Maybe it makes me super powerful, but it doesn't make the average data scientist it just doesn't help because it's too abstract.

Kirill Eremenko: Too specific.

Hadley Wickham: It's too out there it's not concrete enough.

Kirill Eremenko: Interesting. You talk about R like it's a way of expressing yourself as like an art form for you. Is that how you see it?

Hadley Wickham: Yeah, I mean, I really the idea of ggplot2 as the grammar of graphics and I sometimes think about like, well what's... And then in some ways dplyr as the grammar of data analysis. So what builds on top of grammars? How do we get to the poetry of graphics or the poetry of data analysis. And I think that being able to express yourself in code, it's just such a powerful mindset, thinking about code as this medium of communication that I think that's a really powerful lens to look at it.

Kirill Eremenko: That's very cool. What's the thing painters have? They have a paintbrush and that thing they hold in their hand. I forget what it's called.

- Hadley Wickham: Palette?
- Kirill Eremenko: Palette. Yeah. It's like your palette. Okay.
- Hadley Wickham: Actually a while ago I read about someone, it was a masters and fine arts in programming.
- Kirill Eremenko: No way.
- Hadley Wickham: Which I just thought was sort of a fascinating... You study what the great masters have done and copy it and think of it. I don't know. That's going a little too far I think. But that's such a neat idea to think about code not just as a sort of mechanical telling what the [inaudible 00:47:21] computer but as a means of expressing yourself and creating emotions in other humans.
- Kirill Eremenko: Yeah. Have you heard of the ICCC or something like that? It's the international C coding... Something confuscated something C coding contest. Where who creates the most bizarre C code that actually works. For me, I learned about it maybe 12, 15 years ago and I was like wow, that is art and programming. Kind of like postmodernism like we have in normal art. You know when we have very strange looking things but they deliver a message. The same thing here, it's an art to code in a very obfuscated way so that still works, but people don't understand your code.
- Hadley Wickham: Yeah, absolutely.
- Kirill Eremenko: You should start something like that in art. It'd be fun. Okay. Here's an interesting question. I think we've touched on this from Arun, but maybe just to hit the nail on the head. "All I hear is R is for analysis and

Python is for production environment. Why can't we create a production environment based on R? How do you see this developing in the future?"

- Hadley Wickham: Yeah. So you absolutely can get a production environment in R. Lots of big companies do. So last year, at Rstudio con, Jacqueline Nellis gave this really great talk about how they're using R in production at T-Mobile. They're using it to score millions of events every day. There's plenty of people that are doing this. Again, I think sometimes as sort of pushback back, you hear this from data engineers that are primarily familiar with Python, they look at R, it looks really weird to them, makes them feel uncomfortable and they're just like, "No, I don't want to deal with it. Python's the only way you can write production."
- Kirill Eremenko: Yeah. And are there any advantages of writing production in R rather than Python?
- Hadley Wickham: So I think there's a huge advantage in using the same language for exploration and production, because we never... You've got to change languages, particularly if you're now changing people too. As soon as you've got to communicate, "Oh, this is how I did the analysis." And now me as an R user, have to explain it to you, a Python user, or a C++ user or whatever, and you've got to reimplement it. That human to human communication is so expensive.
- Kirill Eremenko: Agree with you. In one role I had to build a statistical model in SQL, which is already funny, right? But that was a constraint at the organization and then when I communicated to the production guys, to the IT

department to put it into production, they actually had their own procedures. And they're like, "No, we can't put it in the way you code it. We have to recode it." And just recording it from SQL back to SQL, but in their own way. That was a whole nightmare. There's so much potential for errors along the way.

Hadley Wickham: Yeah. So, you might enjoy the ModelDB package, which actually translates R codes modeling specifications into SQL. So it generates a sequel to do linear regression in the database and stuff, which is pretty cool.

Kirill Eremenko: Very cool. Very cool. And you mentioned the RStudio Conference. I know you've attended the UseR Conference... UseR or User Conference. What's the difference between the two, and which one would you recommend for our listeners to attend based on their journey in their career?

Hadley Wickham: I mean obviously I'm biased so I'm going to recommend RStudio Con. But I mean they are quite different conferences. So UseR comes from an academic heritage. So a lot of the people presenting are from academia. There seems to be a lot more parallel tracks. I don't know, six to 10 parallel tracks.

Kirill Eremenko: Wow. That's a lot.

Hadley Wickham: It's funny that you say it's a lot because my conferences, as statistician that I go to is the JSM and that has like 50 parallels.

Kirill Eremenko: How do you choose? So speaking of choice paralysis. How do you chose?



- Hadley Wickham: Ironically because there are so many choices, you just need to [inaudible 00:52:04] the things that you know are going to be good. So you never try anything new. [crosstalk 00:52:09] So UseR, it's more academic. It's cheaper, tends to be held in universities.
- Hadley Wickham: It's a little smaller, 800 to a 1000 people these days. Rstudio con, much bigger. We're aiming for 2200 people maybe this year, much bigger. We've gone up to four tracks this year from three tracks in the past. Trying to keep it smaller and more focused. It's more of an industry conference. I don't know, it's a little ritzy. The foods a little better. But, I still think either conference, the best thing about these conferences, isn't going to be true forever, but it's still true right now. The vast majority of people attending UseR or RStudio Con are the only person in their group or their company that's really excited about R. And so you go from being, I'm this weirdo that really likes R and no one else likes it around me to being surrounded by a thousand other weirdos just like you. That is really, really fantastic and really fun in both conferences.
- Kirill Eremenko: Yeah. That sense of belonging. Very important.
- Hadley Wickham: Exactly. Exactly.
- Kirill Eremenko: And speaking of belonging and actually community, there's a great question from Desmond. I think you'll like this one because you're highly invested into equality and helping minority groups. So Desmond Choy asks, "how can data scientists do pro bono work and give back to the community? Faced with unprecedented challenges such as climate change, fake news, growing

income inequality. Are there datasets which data scientists, both professional and amateurs, could data wrangle, do EDA, exploratory data analysis, and model onto shared insights and contribute to solutions?"

Hadley Wickham: Yeah, there's a number of really fantastic organizations that can help you if you're interested in doing this. DataKind is one, I think it's Data For Good is another that let you basically... They match up data scientists who want to give back in some way with organizations doing important work in the world who don't have the budgets to hire really expensive data scientists. I think that's a really fantastic way to give back as a data scientist is to find some organization, maybe it's a local organization, maybe it's smaller. I think that's awesome. Just find, help people, help these, these smaller groups, these non-profits. NGOs really important data and desperately need the help of data scientists.

Kirill Eremenko: But how do you even approach them? Do you send them an email saying, "Hey, I'm a data scientist, I'm willing to contribute three hours a week of my time. What can I do?"

Hadley Wickham: I mean that's how I normally... There's a couple of organizations where I have semi-regular calls with just a chat with their data scientists and answer any questions that they have. I mean that's how it worked for me. I'm not sure how that would generalize. I think the other thing you have to accept is that in most of these cases you would be the first data scientist. And the first data scientist genuinely can't do much data science. You're not going to be deploying the latest in

deep learning technology, but where you can really do provide value is to take those 3000 Excel spreadsheets they have and get it into one nice clean CSV file where you can start to turn data into insight. But I think reaching out to organizations directly or connect up with what DataKind or one of these big organizations that provide these matchmaking services I think is a great way to get started.

Kirill Eremenko: You do yourself, you do quite a lot of work to help communities. I really liked what you said in one interview that there's a lot of underrepresented groups in data science, and there is a way to help and help everybody feel comfortable in data science and pursue a career there. And one of the things you said was, to build a nucleus of people who know each other and who can network and support each other. Tell us a bit about that and what has your experience been with the specific maybe groups where you've helped and have you seen this approach make an improvement?

Hadley Wickham: Yeah, I don't know how much I have ever directly contributed to these things, but where possible when I'm seeing groups of people starting to create some little nucleus, anything I can do to help them I've tried to do. I think one of the biggest successes in the art community and I've had very, very little to do with it is R-Ladies, because that has gone from a group of five women that started various user groups to this worldwide phenomenal that tens of thousands of people are participating in.

Hadley Wickham: I think that finding that core, those few people who can... Starting anything new is tough and having those

sort of people around you that keep cheering you on is so important. And I think the other thing that I think that the R-Ladies have done that is I think really contributed to their success is the sort of focus on process. It's not just about how do we do a good meetup? It's how do we help people create a new meetup that's going to be good? How do we create a meetup in a box that new meet up organizers, how can we give them some process, some checklist to follow so that they can get started in a way that is most likely to lead to success? And I think that to me, thinking about process, thinking about workflow in every... I don't know. That's something I think about in every aspect of life.

Kirill Eremenko: Yeah. That's very cool. Speaking of R-Ladies, have you met Gabriela de Queiroz?

Hadley Wickham: Yes.

Kirill Eremenko: She's really cool. She's been on the podcast twice now and last time she was on... It's crazy. They've grown even between the two appearances on the podcast. I think they grew from something 60 or 70 chapters to 130 chapters around the world and from 30 or 20 countries to over 40 countries. They're doing huge progress. It is very inspiring, as you say, to observe what impact they're having and I guess you're right. It's this model that they provide to people to create these meetups is the key.

Hadley Wickham: Absolutely.

Kirill Eremenko: The tools. Okay. All right. Question. Desmond also had a bonus followup question. It's more of a technical

one. "What are your thoughts on the useful but under the radar, R functions and packages that you personally use quite a bit?"

- Hadley Wickham: Again I don't do that much in data analysis in R. Let me... I don't know. I think one of my favorite types of package now are these ggplot2 extensions. But one that I've loved for a long time is ggrepel. It makes it really, really easy to automatically labeled points on a scatterplot without all the labels glomming on on top of each other. It's a really, really useful package. Scf has really revolutionized piling spatial data and R just makes it so much easier than it used to be. What are some other ggplot2 extensions I was using recently? What was that?
- Kirill Eremenko: How does it feel that people are just developing these extensions for your original tool that you created a little bit back?
- Hadley Wickham: I mean I find it mind blowing. The other thing that just blows my mind is RStudio offers this tidyverse trainer certification and that just, sort of blows me away that not only people are learning and teaching my stuff, but now there's a mechanism by which you could be certified as a trainer that... Yeah just amazing.
- Kirill Eremenko: Wow. That's really cool. Did they get your approval to do that certification?
- Hadley Wickham: Yeah, I mean this mostly happened without me, but I did look through the exams. Greg Wilson who was involved in software carpentry was really instrumental in getting this set up as well. It's a [inaudible 01:01:52]. I think that's a really great combination of

pedagogy. How do you actually teach anything well? How do you teach programming well, plus the basics of the stuff that I really believe in and the tidyverse?

Kirill Eremenko: Yeah. Okay. Wow. Very cool. That's a huge [inaudible 01:02:09] one is a certification for something that you've created. For sure. For sure. Okay. Thank you for those. So here's a couple career questions.

Kirill Eremenko: So Alexander Perrine, I'm not sure how to pronounce this correctly. Sorry, Alex. "With data science being the current in demand career path and all companies starting to employ data scientists, how does someone like myself that doesn't meet the required qualifications on a job posting combat and break into this field?" And he specifies that "most required qualifications that I've seen are asking for 10+ years of experience and want someone to know just about every program under the sun."

Hadley Wickham: Yeah, I mean I think the first thing to remember is that when you're looking at job ads, they just have this laundry list of things that in an ideal world they'd love to have. And just because they ask for that, and you don't have it shouldn't stop you from applying, but if you don't have the experience, or the credentials that they're looking for, you have to figure out how to sell your skills in some other way. And I personally believe a really good way to do that is to think about building up a data science portfolio like a website where you can show off some of the projects you have tackled. And I think doing that sort of focus on, not I'm an amazing programmer, or I know all the latest and greatest deep learning techniques.



Hadley Wickham: But focusing on, I'm a problem solver. I hit the ground running, I can work with your data in whatever crazy format it lives in, I could do some analysis and then I can explain what I've done to people who are not experts. I think if you can build up that portfolio through a combination of writing up case studies of things that maybe you can't share all the details, but you can share the broad outline, analyzing freely available data sets that you're interested in. These days, the Tidy Tuesday project. It's a hashtag on Twitter. Fantastic way of getting a bunch of little datasets. Just show that you can take some data and do something useful with it. That's what most companies want at the end of the day is someone who can work with their data as it is and turn it into some useful insights for the company.

Kirill Eremenko: Yeah, I love that approach. That's something that I also recommend to students all the time, build a portfolio, and you don't even need to launch it on a website. It's gotten so easy. You can, which will take you half a day to put together, or you can even just put it on LinkedIn. I know plenty of data scientists who, and maybe that's even a better way because they're Randy Lao, Kristen Kehrner, Kate Strachnyi, Favio Vasquez, plenty of data scientists who just post their work not even revolutionary packages they build. They just post what they're actually learning themselves and they post it on LinkedIn, which has a blogging capability now, or on Medium and then other people get to read it. So not only you can show that off to employers, potential companies you want to work

for, but other people get to read it and other people get to learn. How great is that? Even if you don't get a job from it, you've helped five people learn the language as well.

Hadley Wickham: Absolutely, and even if no one, or hardly anyone reads it, the act of writing up what you've done, that helps you and improves your communication skills as well.

Kirill Eremenko: Totally agree.

Hadley Wickham: Yeah, really, really valuable.

Kirill Eremenko: And just to, to add about what you said about the laundry list, the list of qualifications. I would say that a lot of the time these employers unfortunately just don't know what they need, because data science has only been around for 10 years. It's not accounting that's been around for hundreds of years and it's very structured and you know exactly I need this accountant, I need a tax accountant. I need actuarial accountant or corporate accountant. Here it's like you're just shooting into the sky. So you might as well just write everything. So yeah, approaching it from that perspective. I agree with you. Just apply for the jobs anyway, have that portfolio, build it up and eventually you'll get something very, very good. Okay. Another career question from Elizabeth West. "How will programming literacy shape the future of workforce? Should everyone learn to code? How can we create pathways to efficiently translate across the space between those who code and those who don't?" Interesting question. Really touches on what we've talked about already today.

- Hadley Wickham: Yeah. So I pretty strongly believe that, not everyone's going to become a programmer, but I think everyone should be able to code, because it just unlocks so much value. There's just so many of these little things in my own life that I automate through code. Always doing little R scripts that take data out of Google sheets and do various things. Like send a bunch of emails with code. Just the ability to automate these mundane life tasks I think is so valuable that everyone should learn it.
- Kirill Eremenko: But wouldn't you say it's like asking for everyone to learn how to code is like asking everyone to learn how to dance? Maybe some people are just not inclined that way. They might do it, but it'll be much harder for someone to learn to code than somebody else.
- Hadley Wickham: Yeah. And I think that's fine. I don't know. I think everyone should have the opportunity and be encouraged to learn how to program. If you don't like it, you're not forced to do it. But I do think it is something that's accessible for the vast majority of people. It's not this thing that can only be done by the intellectual Titans. Anyone can code. Not everyone is going to become a great programmer, but everyone can learn a few little useful things that'll solve some problem in their life. And I think to me that's the key, the teaching programming to a wider audience, is just focusing on what are some useful tools that I can give people that it's not about data structures and algorithms and programming for the sake of programming, but what are some neat tools that I can give people. And I think data science is so great for

that, because everyone has some data they're interested in. Everyone has some website that they read all the time and they'd love to be able to scrape a bit of information off and aggregate it over time and see what's changing.

Kirill Eremenko: Okay. Okay, gotcha. Very, very interesting comments on that. Okay. And so to finish off, let's talk a bit about some future related questions. This is maybe three or four questions related to future. You go to talk about predictions? Do you have a crystal ball, Hadley?

Hadley Wickham: I do not, but I'll try to give predictions that will not be famously wrong.

Kirill Eremenko: Okay. Gotcha. So Morgan Mendis asks another question. "What will be the most challenging aspect of learning data science in the future?"

Hadley Wickham: I mean I think it's going to stay what it is today, which is wrangling crazy data formats into something that makes sense for you. I think it's always going to be hard.

Kirill Eremenko: Gotcha. Next one question is from Martin Kemka. "Given the rise of fake news, deep fakes, and the reduction of trust in data science will statistics be useless in the future and will we just rely on intuition or truthiness?"

Hadley Wickham: I don't think so. I mean I think that statistics and data science and thinking rationally, those are the key tools against the fight against fake news and Mimi things and listicles. It's hard. The brain just has so many shortcuts. Your brain always wants to do the

minimum amount of thinking to solve a problem. And I think statistics and data science, they're sort of part of training your brain to look a little deeper and to consider things a little more fully.

Kirill Eremenko: But on the other hand, there are also the tool of the perpetrators, right? The statistics and data science [crosstalk 01:10:51]

Hadley Wickham: There're lies, damn lies and then there's statistics. People have been saying this for the last 200 years. So I don't think that's anything new.

Kirill Eremenko: Okay, gotcha. Another one from Jennifer Cooper, "one thing we should all be doing to make sure we are ready for the future of data science and machine learning." What is that one thing in your perspective?

Hadley Wickham: I don't know. I'll give you two things. Learn to program and learn the idea of tidy data or normalized data. Just learn how to collect data in a form that can be easily analyzed later on.

Kirill Eremenko: Fantastic. And there's another question here about the future from Ashish. It's "how does the future look like for R", but we already spoke about that in a way. So I'll rephrase that to, what is the one thing that you're most excited for R in the coming future?

Hadley Wickham: I'll tell you what I'm most excited about right now, which is in the very near future, which is we're currently working on a big release of dplyr. So we're going to be releasing dplyr version 1.0 hopefully in March. There's a lot of really cool stuff that's happened behind the scenes. Understood, there's this crazy idea

that you can have a column of a data frame that is itself a data frame, which seems like a crazy idea and it kind of is, but as sort of a data structure it's unlocked a bunch of potential in dplyr. Making things a bunch more flexible so you can express more ideas more succinctly with fewer functions. I'm really excited about this release. It's going to be a big release. Hopefully gives you more power and it will be easier to use. Easier to learn in the long run, which is always the thing that makes me most excited.

Kirill Eremenko: Wow, that's really cool. This is a great a spoiler. I think this podcast will come out just before that then. Awesome. Okay. Well we're done with all the questions and thanks for staying on the show for a bit longer than our usual hour.

Hadley Wickham: You're welcome.

Kirill Eremenko: It's been really exciting. Hadley, huge, huge respect for everything you do and there's plenty of bands in our network and I'm personally a fan. I have your ggplot2 book. I learned a lot from you. So please keep doing what you're doing. You're a great contributor to the community. Amazing, amazing work. Thank you so much for everything.

Hadley Wickham: Thanks so much for having me.

Kirill Eremenko: And yeah, have a fantastic time in the US and we'll speak to you some other time.

Hadley Wickham: Yeah. Thanks.

Kirill Eremenko: Thank you everybody for being part of our conversation today with Hadley. I hope you enjoyed it

as much as I did and learned a lot of new things from Hadley. My personal favorite was the way that Hadley actually thinks about the language and his advice about looking at things from a fresh perspective, forgetting what you did in the past and looking with a new mindset at the same problems and coming up with different solutions. I think it's worked really well for him, and we can see the results, they're impacting all of us, impacting the world and that is something that we can all take away and apply in different areas of our careers and even lives. And on that note, make sure to follow Hadley on social media. You can follow him on LinkedIn and Twitter. On Twitter, he has almost a 100,000 followers. By the time you're listening to this it probably is a 100. If it's not, let's push it to 100,000 and of course check out RStudio.

Kirill Eremenko:

If you haven't yet, hopefully you're inspired to check RStudio and some of the different packages Hadley is working on. As usual, you can find all of the links and materials mentioned on the show in the show notes at superdatascience.com/337. There you will also find the full transcript for this episode. And on that note, if you know somebody who is interested in RStudio, who is a fan of RStudio, who likes Hadley's work, who is following Hadley, then give them the gift of sending this podcast, send them a link to this podcast so they can also listen and learn from Hadley. It's very easy to share. Just send them a link superdatascience.com/337, and once again, thank you so much for being here today. I look forward to



seeing you on the next episode and until then, happy analyzing.