

for many of us when we first get into data science learning coding is probably not the hardest part but learning math and understanding the mathematical machinery behind several analysis techniques and algorithms is often where we feel most overwhelmed and frustrated at least that was the case for me coming from an economics and computer science background feeling like a complete idiot sometimes because for example i had no idea what eigenvector was so today we'll be talking about the math we need to know for data science and i'll be sharing with you seven and a half tips to learn math without beating yourself up or feeling like a complete loser are you ready let's dive right

mathematics is very broad but the good news is there are only a few areas of mathematics that we most often use as data scientists or data analysts they are linear algebra calculus statistics and probability statistics is probably more under science than math because it relies on real world data but let's include it here for completeness i made a quick poll on this channel about a week ago to see which area you often find most challenging to learn interestingly one out of three of you say that calculus is the hardest probably because it's a bit more abstract than others so before i go into detail and bombard you with information i want to make a small note here it's important to know that the level of math knowledge you need to know very much depends on your goal if you want to do a phd in data science or go into machine learning research then it's necessary to study math very well then maybe you should grab the biggest books you can find on calculus and linear algebra and start grinding however i think many of us myself included are doing applied data science instead of doing research meaning we want to solve business problems through data science but we are not necessarily the people who study and come up with new machine learning algorithms for example so we might not need to know every little detail understand the most advanced math concepts it is okay to have some solid high level and basic understanding and that's probably enough just like when we drive a car we care about how to drive them properly and get to the destination rather than understanding how every small component in the car engine works you certainly can if you're a hobbyist but you don't need to that's why not everything we're going to talk about is applicable to you and you need to decide if you want to put them into your learning curriculum alright so the first

important kind of math used in data science is linear algebra it is a branch of mathematics that has everything to do with vectors and matrixes and the operations on them you might encounter linear algebra in several machine learning algorithms for example principal component analysis uses singular value decomposition to present your data in few dimensions linear algebra is also the backbone of the calculations behind all neural network algorithms so it is a very useful area of knowledge to have solid understanding about some main concepts to know in this area is for example dot product matrix multiplication matrix factorization eigenvectors and eigenvalues singular value decomposition and so on the second branch of mathematics that's extremely useful for data science is calculus it is a study of continuous change whether you laughed or hated it in college calculus pops up in several places in data science and machine learning if you've learned about ordinary least squares problem in linear regression or learned about the back propagation algorithm in neural networks you might have encountered a lot of calculus important topics in calculus are limits derivative of a function integrals partial derivatives and the chain rule i used to like calculus a lot back in high school time and i actually found it quite intuitive but the more you dig into it the more complex it can be and so i kind of stopped digging i also find that the calculus necessary for data science is actually usually not super advanced and often limited to the concepts we just mentioned and that's good news for many of us who find calculus challenging statistics and probability is the third important pillar in data science in fact many experts in the field consider classical machine learning nothing but statistical learning many of us are probably already familiar with a lot of basic statistics for example the mean mod quantile standard deviation variance covariance and correlation besides that we might also encounter a conditional probability for example when you learn about bayes theorem and we should also be familiar with topics such as probability distributions sampling and hypothesis testing again what you need to learn really depends on what you want to do honestly i can't remember the last time i used t-test or worked with t-distribution anymore i don't even do hypothesis testing in my work also many concepts for example central limit theorems are important in traditional statistics but are no longer important in modern data science so probably you

don't need to sweat about them you can also find in another video on my channel some detailed explanation of most of the statistical concepts you need to know for data analysis for some of us who might be curious about computational systems data structures and algorithms you might want to learn a bit discrete math but it's not strictly necessary in my opinion i only learned discrete mathematics in my computer science degree and it's quite interesting to learn about sets counting functions basic data structures like stacks queues graphs hash tables and the implementations in different programming languages for example if you work with social network analysis you might find it useful to know about graph data structure and the algorithms you can perform on graphs another very useful concept you'll learn in discrete math is the growth of functions and the big o notation when you have to make a choice of algorithms to use for your project it's really useful to understand the running time and the space requirement of an algorithm using big o notation but i feel like all these concepts are probably more useful further down the road for those of us who want to level up in our data science careers than for those of us who are just starting out okay so it seems like there are tons of things to learn now let's move on to some of the tips that could help you learn math for data science more effectively the first thing as i mentioned earlier is to keep your goal in mind so that you don't unnecessarily overwhelm yourself with things you don't really need to know for example think about where you want to position yourself in the data science field you can choose to be more in the direction of applied data science aka more business oriented kind of data science or you can choose to be more in the direction of data science research and from this you can determine for yourself how deep you want to go into a topic if at all for me i know i wouldn't want to become a data science researcher i'm more inclined towards being an implementation person and i like to create really tangible stuff so i usually just make sure to learn things well enough to be able to explain them in my own way to other people and implement them to solve a problem the second tip that i think is relevant for everyone is to get the fundamentals right there's really no shortcuts because everything is built on top of some fundamental concept if you like learning on coursera like me they have a few really helpful math courses for data science the first one is their science math

skills by duke university which covers the basic mathematics that you need to progress to more advanced math topics so if you feel like you need to brush up your knowledge on set theory functions and graphs logarithms and things like that this is the course you can start with the second course i think is very good is the mathematics for machine learning specialization by imperial college london this specialization contains three courses linear algebra multivariate calculus and pca so they cover pretty much most of the math you need to learn from machine learning it's a beauty that it doesn't cover statistics because well statistics is in a sense not math but for that there are many other resources as well in description below i put a link to some statistics books in r that i found really helpful so you can check it out if you know any good resources for statistics with python please share in the comment below it's often easy enough to press enroll button on an online course but i often find that to stick with it and to get the most out of it is a different story as someone who has self-studied a lot one of the biggest lessons i've learned is to actively take notes and organize them when you're studying and also keep the necessary references to related material you're using so that you can go back to them it's so important to be able to retrieve your knowledge when you need to yet so many times i want to beat myself up for not keeping notes of the things i've learnt because we do tend to get things very quickly after learning something if we don't recall and reinforce that knowledge often enough in the past i was very keen on keeping physical notes but now i usually keep digital notes on notion just so that i have everything in one system and i can look up things quickly you can definitely build your own system in many other ways starfrost one of the subscribers reached out to me a few weeks ago and shared with me an amazing system he built in one note he organized everything he learned in the mit applied data science program in this notebook it's really an amazing piece of system that makes sure he can retrieve anything he learned in just a matter of seconds this kind of system might take some time and effort to build at the start but you'll be really grateful later on and you'll feel so much more motivated and in control rather than finding yourself in the middle of a big overwhelming unmanageable mess during your study another tip i find really useful when learning math is to use divide and conquer methods and to be creative

with it for example when you encounter a gnarly formula for instance this Melbourne equation in reinforcement learning often find useful to chop the formulas into smaller parts and try to understand each smaller component and even say it out loud and use metaphors for it for example this seemingly complicated function can be broken down into a few parts and let's put this equation in the dating context so the left hand side basically means the value of action  $a$  in state  $s$  let's say it's the value of calling your date on Friday evening and this part on the right hand side basically means the total rewards including the rewards at the current moment plus all the discounted future rewards or in this case we can think of it as how much happiness your date will feel right on that Friday and all the following days because you caught her the gamma factor here is smaller than one meaning that every day this happiness effect that your date will experience will get smaller and smaller because people do forget things and this part basically means given your current state and action chosen by the action policy by so this whole equation in our case can mean finding an action policy for example calling your date every  $x$  days at a given time such that you can maximize the total rewards which is your date's happiness level now and in the future well if so you're very good at selfish date breaking things down into smaller parts also makes it easier for us to identify where and why exactly we don't understand something maybe it's because of some unfamiliar notation or symbol this will give you some direction to make progress with your understanding and makes the complicated math concepts feel a bit more manageable another way of learning I often find extremely useful is to actually go to the math formulas in R or Python in the deep learning specialization I took on Coursera a while ago I found it such a great course because you get to code the neural network algorithm from scratch you get to implement the backprop algorithm yourself which is probably the hardest part to understand in neural networks and step by step those complicated matrix multiplications and partial derivatives become much more transparent and intuitive because you can code it yourself and I guess this method can be applied to many different machine learning algorithms for example you can try for once coding an algorithm by hand like K-means for example instead of using the out of the box function in the scikit-learn library I promise

you'll get to understand it so much better if you watch some of my earlier videos you might have also noticed that i'm a big fan of visual learning if you're a visual learner then turning to visual ways of learning is a great way to learn math we have interactive books tools and also youtube channels like three blue one brown that focus on explaining things with animations and visualization and so all these abstract concepts become much easier to understand and digest one of the very common feelings that we face when learning math is the imposed feeling but it is what it is it's just a feeling one of the lessons learned for me is when you feel impostor you don't need to act on it just let it pass but when you feel the motivation act on it keep learning and realize that that feeling probably doesn't have much to do with your true ability and your knowledge and also another thing that i learned is that when you don't understand something for example if you read a book and it feels like this whole thing is so complicated and you feel like you want to give up and you feel like you're not smart enough to understand it actually it's probably not all your fault it could also well be that that book doesn't explain well enough that concept i would go on youtube on google to find a better explanation a better presentation of what i've tried to learn and most of the time there is a better explanation elsewhere on the internet i hope you enjoyed these tips and don't forget to check out this video about how much statistics you need to know for data analysis with that i'll see you next video bye