WEBVTT

1
00:00:00.110 --> 00:00:01.350
The notes, dog

2
00:00:33.080 --> 00:00:36.420
Ty Tuff, Ph.D.: Nate, could we add a question about how awake people are this morning?

3
00:00:36.690 --> 00:00:42.340
Nate Quarderer (Earth Lab/ ESIIL): How awake are people? Okay? Yes, I can. I can do that quick on the fly here.

4
00:00:44.410 --> 00:00:50.730
Nate Quarderer (Earth Lab/ ESIIL): What can also be kind of fun is like, if folks want to use like an emoji, you know. Maybe like

5
00:00:51.080 --> 00:01:00.550
Nate Quarderer (Earth Lab/ ESIIL): if you want to give us like a thumbs up, I'm super awake, or a thumbs down, or like a sleepy emoji, or kind of sleepy. I'll add this to the doc, though, too, for folks that don't like use Emoji

6
00:01:41.510 --> 00:01:47.010
Nate Quarderer (Earth Lab/ ESIIL): a little like scale there on the weakness. Yeah.

7
00:01:47.890 --> 00:01:57.090
Nate Quarderer (Earth Lab/ ESIIL): cool. Well, we've got 30 folks in here. Now, Rachel and Virginia, wait. And Ty, how are you all feeling? Do you want to go ahead and get rolling?

8
00:01:57.120 --> 00:02:05.750
Nate Quarderer (Earth Lab/ ESIIL): We have couple more minutes, Virginia. Seems like she's ready. Okay, cool. Rachel's kind of thumbs up via Thai. And you're our lead instructor here this morning. How are you feeling, my friend

9
00:02:06.240 --> 00:02:16.529
Ty Tuff, Ph.D.: feeling good. Thank you. Let's see. Let me send out a link real quick to the web page that we're going to refer to.

10
00:02:18.290 --> 00:02:23.780
Ty Tuff, Ph.D.: So this is on the Pre Hackathon training sessions, and I will

11
00:02:24.010 --> 00:02:27.899
Ty Tuff, Ph.D.: share my screen

12
00:02:29.700 --> 00:02:30.420
Rachel King: oops

13
00:02:34.010 --> 00:02:35.310
Ty Tuff, Ph.D.: alright.

14
00:02:36.410 --> 00:02:53.120
Ty Tuff, Ph.D.: So you can go to the. This is just a website that scrolls through and has all the stuff we're going to talk about. So you can go to this yourself, and that way you can make it really big and easy to see and walk through.

15
00:02:54.590 --> 00:03:05.879
Ty Tuff, Ph.D.: My discovery environment is broken this morning, so if I need to do any code with you, I'll bring in my little our studio and show you something. Live

16
00:03:06.210 --> 00:03:13.409
Ty Tuff, Ph.D.: while we're working on that in the background. But, I don't think we need to do any of that, because most of it is us just trying to

17
00:03:13.710 --> 00:03:18.189
Sodiq Jinad: get you to change the way you're thinking about this process a little bit.

18
00:03:19.890 --> 00:03:21.429
Ty Tuff, Ph.D.: Let me change my.

19
00:03:22.910 --> 00:03:28.020
Ty Tuff, Ph.D.: with the viewer over here. Okay. so

20
00:03:30.080 --> 00:03:32.869
Ty Tuff, Ph.D.: try to make a second. See as many people as possible.

21
00:03:34.390 --> 00:03:48.310
Ty Tuff, Ph.D.: Good. Most people have their cameras off understandably. It's early in the morning. okay. So what we're going to talk about is sort of how to get yourself set up for doing analyses.

22
00:03:48.510 --> 00:03:56.409
Ty Tuff, Ph.D.: So on the big narrative arc of what we've been talking about. So far. we have shown you how to get into the discovery environment.

23
00:03:56.760 --> 00:04:16.369
Ty Tuff, Ph.D.: Now with that discovery environment which is your big computer. We want to learn how to get a bunch of data and bring it in. Set that up in a way that's convenient for analysis, and then you can do your analysis. And so hopefully, you, when we get to the hackathon, have a good grasp of all 3 of those steps, and can easily link them together and start doing really cool stuff.

24
00:04:17.380 --> 00:04:24.229
Ty Tuff, Ph.D.: Now, my job is to sort of give you a new way of thinking about moving data around. And

25
00:04:24.400 --> 00:04:38.699
Ty Tuff, Ph.D.: you know, as a Ph d. We sort of have to start all of our lessons with some pontification about how it philosophically matters. So that's what you're getting now is that I need you to think differently about your data. And

26
00:04:39.030 --> 00:05:02.149
Ty Tuff, Ph.D.: that's a hard thing to do, especially for people who have been working with data for a really long time at a really high level. But the way most of us were trained to think about data was something like a sculptor like you bring in this huge chunk of data. Then you chip away at all the things that aren't what you want, and you're left with this beautiful sculpture that you're that is the thing that you want.

27
00:05:02.390 --> 00:05:12.250

Ty Tuff, Ph.D.: and that involves getting that rock to you, and that is big and expensive and limits, how much sculpting you can do.

28
00:05:12.570 --> 00:05:20.670
Ty Tuff, Ph.D.: And the big data came around. It was originally defined as data that was too big to fit on a single machine.

29
00:05:20.820 --> 00:05:27.450
Ty Tuff, Ph.D.: And so just that pure definition means that none of the methods that existed worked anymore.

30
00:05:27.670 --> 00:05:37.810
Ty Tuff, Ph.D.: And that's why big data became a buzz buzz word is because we now needed all these new techniques to be able to deal with data that was distributed across multiple machines

31
00:05:37.870 --> 00:05:40.369
that was too big to fit in one place.

32
00:05:40.580 --> 00:05:52.220
Ty Tuff, Ph.D.: And so when you're thinking about analyzing data, I need you to. Now switch over to that multi machine thinking, you just think there's no way I can do this in one machine.

33
00:05:52.510 --> 00:05:55.840
Ty Tuff, Ph.D.: so you can't just build stuff

34
00:05:56.100 --> 00:06:05.650
Ty Tuff, Ph.D.: from scratch like you would a house or a sculpture. You can't. You need to think of. How do I, instead of bringing all of this stuff to me?

35
00:06:05.920 --> 00:06:29.589
Ty Tuff, Ph.D.: How do I just patch into all these streams of information? So if you think about those old operator boards that they used to have in the White House, where sort of operators had to sit and plug the phone line into a bunch of different places. Right? There's just information trying to come in through every portal on that wall, and the President sitting there just patching into which person are they talking to here or there?

36

00:06:29.730 --> 00:06:32.940
Ty Tuff, Ph.D.: And oh, I love the phone, Emoji. Thanks, Nate.

37
00:06:33.300 --> 00:06:49.659
Ty Tuff, Ph.D.: So I that's the thing we're gonna really try to get to right now is thinking differently about our data, thinking differently about how data should flow in and out of our analysis and think differently about how we're gonna organize those data. Okay?

38
00:06:49.680 --> 00:06:54.750
Ty Tuff, Ph.D.: excuse me, I had a little bit of a cough, so I'll try to not blow out your speakers.

39
00:06:55.020 --> 00:06:57.610
Ty Tuff, Ph.D.: So

40
00:06:58.040 --> 00:07:05.860
Ty Tuff, Ph.D.: let's give ourselves a little analogy to to grasp onto. So we can start to think about these tools and think about the way to conceptualize this.

41
00:07:05.920 --> 00:07:14.759
Ty Tuff, Ph.D.: Hmm one of my favorite photographers is David Yaro, and he loves to engage in this debate on whether you make or you take photographs.

42
00:07:15.160 --> 00:07:31.460
Ty Tuff, Ph.D.: And what I love about this debate is that you know, particularly, nature photographers often think like I'm gonna go and find the perfect spot. And I'm gonna find the perfect moment, and I'm gonna find the perfect angle. And then when that thing happens in front of me, I'm gonna take the picture.

43
00:07:31.700 --> 00:07:46.759
Ty Tuff, Ph.D.: But as you look at this picture right in front of you. There's no way this would happen naturally. Right? That is a wild animal, and David Yarro in particular, is known, for normally you would take pictures of animals through a long telephoto lens, and that loses a lot of the detail and the richness and sort of

44
00:07:46.900 --> 00:08:00.260
Ty Tuff, Ph.D.: the panic that you get from being close to something.

And so instead, he puts himself in these metal cages and gets a really short lens and gets really close to these things. So you get the intensity.

45
00:08:00.440 --> 00:08:03.539
Ty Tuff, Ph.D.: But every single thing about this is staged.

46
00:08:03.730 --> 00:08:18.150
Ty Tuff, Ph.D.: And so what happened here is David Jarro in his head, visualize the scene, and then went and figured out a way to create it. And that is what we need to do when we are making quote unquote, making a data queue, I want you to.

47
00:08:19.120 --> 00:08:23.570
Ty Tuff, Ph.D.: Good. Somebody posted some more images in the slack in the chat. So

48
00:08:25.840 --> 00:08:27.859
Ty Tuff, Ph.D.: oh, nice good gifts.

49
00:08:28.980 --> 00:08:34.309
Ty Tuff, Ph.D.: But okay? so the

50
00:08:34.539 --> 00:08:42.809
Ty Tuff, Ph.D.: the idea here, when you're thinking about data is you're creating this picture. That data cube you're making is this layered image.

51
00:08:43.140 --> 00:08:48.060
Ty Tuff, Ph.D.: And you have to go out and figure out the stuff that comes into that and makes it creative.

52
00:08:48.830 --> 00:09:04.989
Ty Tuff, Ph.D.: This one is another David Yarrow, one that's absolutely amazing. Right. It's just buffalo, but this is a made image. It feels taken. But to get those things charging right at you and get yourself placed right in the path of them charging with the short frame, short lens. Camera.

53
00:09:05.050 --> 00:09:12.179
Ty Tuff, Ph.D.: You know he's like in a protective machine right now

to keep it from getting trampled from those buffalo.

54
00:09:13.410 --> 00:09:15.410
Ty Tuff, Ph.D.: Alright. So

55
00:09:15.940 --> 00:09:26.759
Ty Tuff, Ph.D.: we're gonna talk about the technology of capturing this with our other super favorite photographer, Ansel Adams. If you went into a house in the nineties you definitely saw an an' Adams picture on the wall.

56
00:09:26.890 --> 00:09:34.969
Ty Tuff, Ph.D.: And he was amazing. But this is how he had to take pictures right. He drove this car around, and he set up this huge machine on top of the car.

57
00:09:35.100 --> 00:09:36.370
Ty Tuff, Ph.D.: and

58
00:09:36.490 --> 00:09:59.590
Ty Tuff, Ph.D.: first he would have to say, What do I want to capture, and so he'd have to aim his camera at the scene at the landscape, and then he'd have to say, How do I want to capture it, and for that it was like, take a piece of glass, pour chemicals that are reactive to red, let them dry poor chemicals that are reactive to blue. Let them dry, pour chemicals that are reactive to something else. Let them dry to make your own film.

59
00:09:59.590 --> 00:10:23.319
Ty Tuff, Ph.D.: So he's literally having to envision what the impact product is gonna look like when he's pouring chemicals onto a sheet of glass before he even opens up the frame. So when we are searching through data in the cloud. Here's the scene. I want you to have right the landscape. This is the unlimited landscape of data that you could go and retrieve from the cloud on anything that you could possibly imagine.

60
00:10:23.480 --> 00:10:35.750
Ty Tuff, Ph.D.: You just have to point your camera at it. But then that information started to flow towards you. Right, the lights hitting those trees and those mountaintops and everything. And it's coming at you. You now need to set up your camera

61
00:10:35.780 --> 00:10:44.599
Ty Tuff, Ph.D.: to capture how much you want the mood you want, you need to modify those data. And ultimately you're trying to get them imprinted on this piece of film.

62
00:10:44.900 --> 00:10:48.810
Ty Tuff, Ph.D.: Our analogy here is that that final piece of film is our data cube.

63
00:10:49.100 --> 00:10:58.809
Ty Tuff, Ph.D.: that data cube that you're building is an expression of what you want to say with data. It's an expression of the questions you want to ask with data. It's an expression of

64
00:10:58.900 --> 00:11:02.439
Ty Tuff, Ph.D.: sort of the style in which you want to do those things.

65
00:11:02.670 --> 00:11:11.330
Ty Tuff, Ph.D.: And so you have to think about this whole system when we're thinking about moving data in and out of our analyses on the cloud.

66
00:11:12.450 --> 00:11:15.170
Ty Tuff, Ph.D.: Okay? So

67
00:11:15.270 --> 00:11:20.490
Ty Tuff, Ph.D.: here we are with all of the packages that I have loaded that you don't need to worry about.

68
00:11:20.610 --> 00:11:22.920
Ty Tuff, Ph.D.: But we're gonna switch analogies real quick.

69
00:11:24.450 --> 00:11:36.120
Ty Tuff, Ph.D.: We're gonna bounce back and forth between these 2 analogies. But this is to get us a little more specific. So right now we would be thinking about the light through the lens and the old analogy sort of how much light do you let in? Or do you let out?

70
00:11:36.560 --> 00:11:49.230

Ty Tuff, Ph.D.: But when we think about moving data in the cloud, we have 3 really big problems that we want to solve right off the bat. So switching analogies just to talk real quick about the 3 big problems we want to solve

71
00:11:49.490 --> 00:11:52.799
Ty Tuff, Ph.D.: first is called the Rat through the snake problem.

72
00:11:53.220 --> 00:11:58.719
Ty Tuff, Ph.D.: So here could I take a quick poll in the chat for what people think

73
00:11:58.760 --> 00:12:01.599
Ty Tuff, Ph.D.: is stuck in this snake.

74
00:12:06.380 --> 00:12:10.230
Ty Tuff, Ph.D.: I have cell phone phone, phone, phone, jewelry box.

75
00:12:10.860 --> 00:12:23.609
Ty Tuff, Ph.D.: cell phone cell phone, a keyboard. Oh, that'd be a really tiny keyboard, but I like it. It's a mouse trap. So there was a mouse stuck in the mouse trap, and it ate the mouse and the mouse trap.

76
00:12:24.540 --> 00:12:35.370
Ty Tuff, Ph.D.: Yeah, if you remember the lawsuit between Samsung and Apple this would have to have curved edges, I think, in order for it to be a cell phone above.

77
00:12:35.400 --> 00:12:36.460
Ty Tuff, Ph.D.: And

78
00:12:36.580 --> 00:12:49.319
Ty Tuff, Ph.D.: yeah, here, as an example of, I mean, there's a natural problem that snakes have of like, how big of a thing can they eat and get it through their system? But here, when you combine 2 data, sets a snake and them a mouse and a mouse trap.

79
00:12:49.540 --> 00:12:52.240
Ty Tuff, Ph.D.: you now have a really big problem getting those things through.

80
00:12:52.630 --> 00:13:09.130
Ty Tuff, Ph.D.: Now, what is the cheapest, easiest fix we can give you for fixing this snake through rat through the snake problem is a bigger computer. Okay? And so this is why we sent you to the discovery environment. Discovery environment lets you go bigger. It just lets you be a bigger snake.

81
00:13:09.280 --> 00:13:17.420
Ty Tuff, Ph.D.: so that rat just doesn't feel as big aim. But as you can imagine, we're going to move to Problem number 2.

82
00:13:19.530 --> 00:13:25.559
That only fixes some of the problems because we now move to the antelope through the python problem.

83
00:13:25.970 --> 00:13:29.850
Ty Tuff, Ph.D.: So you have scaled up to be a much bigger snake.

84
00:13:29.950 --> 00:13:41.739
Ty Tuff, Ph.D.: and your appetite gets much bigger. And now, all of a sudden, you've still gone way way bigger, and you've eaten something that's too big for you. And now you have any a little stuck inside of you? So

85
00:13:41.770 --> 00:13:58.210
Ty Tuff, Ph.D.: how do we fix this one? This one is to mount data. So if we fix the first problem by giving ourselves a really big snake, but once the snake is really big. Now, we need to figure out some ways to pre digest our food. We need to break up that food. We need to find a ways way to feed

86
00:13:58.250 --> 00:14:01.789
Ty Tuff, Ph.D.: the food to us in smaller, more remote bits.

87
00:14:03.500 --> 00:14:07.539
Ty Tuff, Ph.D.: Once we get those we move on to the really high class problem

88
00:14:07.760 --> 00:14:16.239
Ty Tuff, Ph.D.: of trying to drink from a fire hose. Right? You have

calibrated the speed and size of input. So the data can come at you really fast.

89
00:14:16.770 --> 00:14:22.130
Ty Tuff, Ph.D.: You've calibrated your computer to be as big as possible. So you have room to receive those data.

90
00:14:22.440 --> 00:14:31.739
Ty Tuff, Ph.D.: But now those data are going to be flowing at you so quickly that you're it's going to be impossible to make good inference from those data you just like can't run New

91
00:14:32.010 --> 00:14:41.840
Ty Tuff, Ph.D.: Random Forest models every day, or something to help you keep track of that. You now have this really really big analytics problem for keeping up with that speed of input.

92
00:14:42.380 --> 00:14:50.879
Ty Tuff, Ph.D.: so we're gonna go through some solutions for all 3 of those problems. Okay? So we already know the solution problem one, get on the discovery environment. Get a bigger computer.

93
00:14:52.810 --> 00:14:56.470
Ty Tuff, Ph.D.: Number 2 is the mounting. We're going to talk about that right off the bat.

94
00:14:56.550 --> 00:15:07.750
Ty Tuff, Ph.D.: and then, we're going to move on to how to use AI and Ml. And the second half of this lesson, not me. But the next instructor is going to talk about sort of how to use a an AI and Ml.

95
00:15:07.980 --> 00:15:12.179
Ty Tuff, Ph.D.: to sort of help you keep up with this deluge of data that might be coming at you.

96
00:15:12.800 --> 00:15:15.870
Ty Tuff, Ph.D.: Okay, so mounting data.

97
00:15:16.550 --> 00:15:17.270
okay.

98
00:15:18.190 --> 00:15:32.270
Ty Tuff, Ph.D.: why do I say mounting instead of downloading? Well, this is, we're plugging into that wall. So that we have that operator wall, it has unlimited amounts of data that we could plug into. And we say, Ping, I want these particular data.

99
00:15:32.340 --> 00:15:37.029
Ty Tuff, Ph.D.: Now let's go look at what we're gonna find. Okay, these are just data on the web. This is

100
00:15:37.210 --> 00:15:58.200
Ty Tuff, Ph.D.: a website called hydro sheds. This is a large repository of river specific data. And we're gonna go through one of the pages that just shows you all of the different data that are in this website. We're gonna do that in a few minutes. But first let's just talk about this one little element of mounting data before we get into the specifics of a bunch of types of data.

101
00:15:59.580 --> 00:16:12.769
Ty Tuff, Ph.D.: Okay? And that's good. And one of these types of data is called void fill. DM, this is a special type of DM, where somebody has gone through the effort of filling in all of the little gaps.

102
00:16:13.170 --> 00:16:30.299
Ty Tuff, Ph.D.: And specifically, this is a hydrologic website. And so they're really interested in water flowing down. So these are all the places in your aster where water might artificially get stuck because there's just avoiding data. But the dem is gonna read it as a low spot in their terrain or something.

103
00:16:30.630 --> 00:16:32.729
Ty Tuff, Ph.D.: And so you've had actual

104
00:16:32.740 --> 00:16:39.519
Ty Tuff, Ph.D.: real, you know, nice professional scientists go through and correct this in a in a highly professional way.

105
00:16:40.270 --> 00:16:50.459
Ty Tuff, Ph.D.: Now, when you look at the data types here, they have a really high resolution, 1, 3 s, and they have a slightly lower resolution, one and a slightly lower resolution one.

106
00:16:51.320 --> 00:17:01.639
Ty Tuff, Ph.D.: They don't allow remote downloads for the highest of resolution. So if your question requires going to the highest of resolutions. The 3 s.

107
00:17:01.700 --> 00:17:07.510
Ty Tuff, Ph.D.: then you're gonna have to do the old way of downloading the data and importing it like you would expect.

108
00:17:08.050 --> 00:17:11.300
Ty Tuff, Ph.D.: But if you can deal with 15 s

109
00:17:11.480 --> 00:17:15.950
Ty Tuff, Ph.D.: data, this is a geographic seconds, not time, seconds.

110
00:17:16.030 --> 00:17:19.820
Ty Tuff, Ph.D.: Then those allow mounting.

111
00:17:20.030 --> 00:17:23.910
Ty Tuff, Ph.D.: and that allows us to utilize this technology that I'm about to show you

112
00:17:28.680 --> 00:17:29.670
Ty Tuff, Ph.D.: find me.

113
00:17:32.840 --> 00:17:35.069
Ty Tuff, Ph.D.: Okay. So in here.

114
00:17:35.810 --> 00:17:40.330
Ty Tuff, Ph.D.: let me make this a little bigger. This is again our code.

115
00:17:41.090 --> 00:17:42.220
Ty Tuff, Ph.D.: And

116
00:17:43.880 --> 00:17:49.130
Ty Tuff, Ph.D.: this right here is the address that I would have found

117

00:17:49.320 --> 00:17:58.129
Ty Tuff, Ph.D.: on this website. So here, if I just go here and I say 15. Second for North America.

118
00:17:59.370 --> 00:18:04.860
Ty Tuff, Ph.D.: America, North and Central America. Right here I left click

119
00:18:05.970 --> 00:18:07.769
Ty Tuff, Ph.D.: and copy link

120
00:18:07.980 --> 00:18:12.979
Ty Tuff, Ph.D.: Bing. So this is just like any generic link that you find on the Internet

121
00:18:13.010 --> 00:18:16.480
Ty Tuff, Ph.D.: where you want to download the thing, you're just getting the copy link.

122
00:18:16.570 --> 00:18:28.029
Ty Tuff, Ph.D.: So I'm showing you a specific example. But this is very generic, right? I have just put that link in there. Now that Link tells me a couple of things. One. It tells me that this is a zipped file.

123
00:18:28.180 --> 00:18:40.210
Ty Tuff, Ph.D.: So a lot of you that have downloaded files would be familiar with a zipped file. You have to download this thing and unzip it and so. And then inside that zip file is going to be

124
00:18:40.390 --> 00:18:42.619
Ty Tuff, Ph.D.: a Tif file that we want to download.

125
00:18:43.680 --> 00:18:52.350
Ty Tuff, Ph.D.: And so I have just put all 3 of those I've put those sorry I haven't talked about the first one yet. I put those 2 things in a string

126
00:18:52.400 --> 00:19:01.790
Ty Tuff, Ph.D.: so Glue here is going to pretend like they were continuous, and they don't have them separated by a comma. Just I did

that just so I could see, show you the 3 parts.

127
00:19:02.170 --> 00:19:05.880
Ty Tuff, Ph.D.: Okay, so that we have the link that I downloaded

128
00:19:06.890 --> 00:19:25.260
Ty Tuff, Ph.D.: or the link that connects to the download. And this could be any link you find anywhere on the Internet. Here is the folder that you would find inside of that. or a generic version of that. Sometimes you have to download one of these Zip files and open it to know what this file path structure is to fill this in.

129
00:19:26.040 --> 00:19:29.360
Ty Tuff, Ph.D.: But that lets you just set it up. So you never have to download anymore.

130
00:19:30.130 --> 00:19:37.470
Ty Tuff, Ph.D.: And then here is the magic sauce. Okay, vsi is Gdall's virtual file system.

131
00:19:37.830 --> 00:19:44.729
Ty Tuff, Ph.D.: And it allows us to, instead of downloading this, to just plug into them. This is that plug.

132
00:19:44.890 --> 00:19:52.560
Ty Tuff, Ph.D.: So why do I have 2 commands here? Well, vsi, curl, this is the just read things on the Internet. So if this wasn't zipped

133
00:19:53.700 --> 00:20:01.780
Ty Tuff, Ph.D.: and it was just a file that was available. I could just use Vsi curl. And you'll see some examples of that later. When I go through specific data.

134
00:20:01.850 --> 00:20:09.569
Ty Tuff, Ph.D.: they're just open. They're not locked. They're not zipped. You just go and Bsa curl, and you just have plugged into them. And it's as if they're in your computer.

135
00:20:10.010 --> 00:20:21.570
Ty Tuff, Ph.D.: this one, because it's zipped. I have to add Vs izip, which all it lets us do is peek inside that zipped folder. So we don't

have to unzip anything. We're just peeking inside. Still

136
00:20:21.780 --> 00:20:31.259
Ty Tuff, Ph.D.: in in the Server in the cloud. We're not bringing any of this into our machine. And then I have added, this is a piping function. So

137
00:20:31.360 --> 00:20:41.849
Ty Tuff, Ph.D.: after I have glued together this address of the Vsi Zip vsi, curl the address to download, and where I want it to go inside.

138
00:20:43.660 --> 00:20:50.109
Ty Tuff, Ph.D.: I add this, which says, Okay, and convert it to a raster. And I do that. So I.

139
00:20:50.210 --> 00:20:54.489
Ty Tuff, Ph.D.: This is for a DM. At 15 s for all of North America.

140
00:20:55.590 --> 00:20:58.299
Ty Tuff, Ph.D.: Here it takes 7 s.

141
00:20:59.300 --> 00:21:01.270
Ty Tuff, Ph.D.: and I have a raster.

142
00:21:02.410 --> 00:21:15.669
Ty Tuff, Ph.D.: So for those of you who have in the past downloaded it can often take 7 s to unzip the folder that you've downloaded. It can be right. It could take an hour or 2 to download.

143
00:21:17.390 --> 00:21:24.879
Ty Tuff, Ph.D.: I'm gonna Tyler asked. How do we know and find out which are accessible? By Gdahl? And I'm gonna go over that a little bit.

144
00:21:25.040 --> 00:21:37.999
Ty Tuff, Ph.D.: Usually you can assume that they are, and you try it, and if it doesn't work, it pops up with this warning that says this this website does not allow remote, remote download or remote access.

145

00:21:38.450 --> 00:21:50.200
Ty Tuff, Ph.D.: And so it's like, it's sort of a lock that people can put on stuff that you can't necessarily tell ahead of time. But when you ping the website it'll just send you that message saying, essentially, somebody's locked this and won't let you.

146
00:21:50.320 --> 00:21:54.100
And that is because it's costing them money. I'm going to show you

147
00:21:54.410 --> 00:22:01.809
Ty Tuff, Ph.D.: further down. We can do all of our analyses on their side before they give us any of this information.

148
00:22:02.000 --> 00:22:05.369
Ty Tuff, Ph.D.: So we can do really complicated

149
00:22:05.720 --> 00:22:19.780
Ty Tuff, Ph.D.: all kinds of complicated functions operations on these data. And the server has to do that. And then they give you just the end result. And if you do really big computations that can cost them a lot of money. And so in things like the hydro sheds.

150
00:22:19.810 --> 00:22:26.650
Ty Tuff, Ph.D.: They're just turning off the highest resolution ones, because it was probably costing them too much money to service those data for everybody.

151
00:22:27.480 --> 00:22:40.309
Ty Tuff, Ph.D.: But I'll show you some more of those examples. Okay. So we pulled those data in 7 s. Now, I want to show you that we can easily just

152
00:22:41.110 --> 00:22:43.439
let me see if this is done. I

153
00:22:43.660 --> 00:22:49.170
Ty Tuff, Ph.D.: it's doing a final render with one. This function name is incorrect. But

154
00:22:49.500 --> 00:22:56.630
Ty Tuff, Ph.D.: There's as soon as this finished inventory. We'll just

refresh this page, and those will be fixed. Okay,

155
00:22:57.820 --> 00:23:09.569
Ty Tuff, Ph.D.: So here is another example of pass behavior. You might go. I'm going to download a dem, and I'm going to download a slope layer and I'm going to download an aspect layer

156
00:23:10.260 --> 00:23:14.000
Ty Tuff, Ph.D.: instead. Here, I just grab those

157
00:23:14.020 --> 00:23:23.319
Ty Tuff, Ph.D.: I pull in the one DM. Which I haven't even pulled in my machine I've just plugged into, so it never! I never have to download it. I'm just reading it directly from the server.

158
00:23:23.510 --> 00:23:30.880
Ty Tuff, Ph.D.: But then I can take that, and I can calculate the slope from that memory, and I can click, calculate the aspect directly from those memories.

159
00:23:32.820 --> 00:23:41.840
Ty Tuff, Ph.D.: Do zipped tifs incur. The question is, do Tif Zifs incur a performance penalty in the case of

160
00:23:42.330 --> 00:23:44.300
Ty Tuff, Ph.D.: I'm guessing. And

161
00:23:44.310 --> 00:23:50.280
Ty Tuff, Ph.D.: can you be a little more specific cause. I was talking about other things I'm are you saying in the remote?

162
00:23:50.520 --> 00:23:52.100
Ty Tuff, Ph.D.: This should be faster.

163
00:23:52.830 --> 00:24:18.140
Ian Breckheimer: But I need you to click, Ian. If you could clarify your question a tiny bit. In which case you you mean, yeah. Sorry about that. What I was curious about was why hydro sheds had decided to mount zipped files there, instead of just having tiffs be available as as cogs on in an uncompressed format that was a little bit unusual choice, and I wasn't sure what was driving it.

164
00:24:18.170 --> 00:24:22.169
Ty Tuff, Ph.D.: Totally. I have no idea. My guess is something

165
00:24:22.420 --> 00:24:27.539
Ty Tuff, Ph.D.: just in their mass production system, you know, they just do it because it's easy.

166
00:24:27.640 --> 00:24:30.950
Ty Tuff, Ph.D.: but but I don't know but the nice thing is once you

167
00:24:31.220 --> 00:24:42.479
Ty Tuff, Ph.D.: if you use the Vsi tif function, it's just peeking inside of there, and it can move things easily in and out and doesn't get any penalty for for unzipping them per se

168
00:24:43.840 --> 00:24:45.310
Ty Tuff, Ph.D.: should be just as fast.

169
00:24:47.350 --> 00:24:49.190
Elsa Culler: Thanks. It's time to bring up the flow.

170
00:24:49.580 --> 00:25:02.180
Elsa Culler: Tyler.

171
00:25:03.690 --> 00:25:10.710
Ty Tuff, Ph.D.: Yeah. Let give me an I. It's down like another 3 steps in this in the presentation

172
00:25:12.400 --> 00:25:13.270
Elsa Culler: cool.

173
00:25:14.180 --> 00:25:16.770
Ty Tuff, Ph.D.: Okay. So

174
00:25:18.530 --> 00:25:32.890
Ty Tuff, Ph.D.: for those of you who do a lot of geography, these will feel like just normal steps. I just wanted to throw them in here for people who well, we need the information, for later in the code, they

also wanted to just highlight it for people who don't do this all the time.

175
00:25:33.250 --> 00:25:44.490
Ty Tuff, Ph.D.: Geographic data are projected in a particular way, and this is because you can never do a perfect translation between a spherical planet and a flat representation.

176
00:25:44.550 --> 00:25:59.889
Ty Tuff, Ph.D.: And those different transformations have different codes. This 1, 43, 26 is a really really standard code, probably most of the time. This is what people are going to ask for a request in.

177
00:26:00.840 --> 00:26:09.140
Ty Tuff, Ph.D.: but they sometimes will return data in a different projection. So when we

178
00:26:09.280 --> 00:26:10.770
Ty Tuff, Ph.D.: go, and

179
00:26:10.970 --> 00:26:22.409
Ty Tuff, Ph.D.: this is coming up. I'm getting ahead of myself a little bit. But when we make a call for the data we are going to do it, using a spatial object, and that spatial objects need to usually needs to be in this projection.

180
00:26:22.640 --> 00:26:28.690
Ty Tuff, Ph.D.: And so we often are going to do come up with 2 projections of our area of interest.

181
00:26:28.760 --> 00:26:33.850
Ty Tuff, Ph.D.: one. To compare the Pre stuff, and one to compare with the stuff we've received.

182
00:26:36.010 --> 00:26:39.259
Ty Tuff, Ph.D.: The question about what language this is in. This is an R.

183
00:26:39.480 --> 00:26:50.360
Ty Tuff, Ph.D.: And all of these things work in Python nicely. Also, the code obviously is a little different, but these systems are meant

to be agnostic to the particular coding language you're using at the end.

184
00:26:51.860 --> 00:27:00.520
Ty Tuff, Ph.D.: okay, so this is so I've transformed into 2 different projections. That I'm gonna see later.

185
00:27:01.110 --> 00:27:25.420
Ty Tuff, Ph.D.: The question is that if do I have a python version to? I don't yet. We're trying to get that developed. I don't know if we'll have it done by the hackathon. But we're certainly have it done really soon. But a lot of this code, because it's so specific. If you take this code and put it into Chat Gpt, and ask for python translation. It'll give you a fully working python translation of the thing because you're giving it really specific code to start with. So that would be my first step.

186
00:27:25.420 --> 00:27:32.020
Ty Tuff, Ph.D.: and we'll try to get stuff written up for you. But if you need stuff right now, just take this and copy and paste it into chat and ask for a python translation.

187
00:27:33.210 --> 00:27:41.329
Ty Tuff, Ph.D.: Okay, here I. So here, we're just generally talking about area of interest. So

188
00:27:41.680 --> 00:27:42.590
Ty Tuff, Ph.D.: that

189
00:27:42.950 --> 00:27:53.709
Ty Tuff, Ph.D.: specific version of this system that I'm talking about today is the one that was sort of designed for spatial analyses specifically. And so the thought there is

190
00:27:53.760 --> 00:28:07.859
Ty Tuff, Ph.D.: because it's a spatial question. You're gonna have an area of interest. And that is essentially going to be your searching query. So you're gonna ask the system for to send you data. And one of the pieces of information that it's gonna want is sort of what is the geographic

191
00:28:08.110 --> 00:28:20.029

Ty Tuff, Ph.D.: limits, the extent of that that you want. And so right here, when I'm asking for bounding boxes. Right? So first I have. This is the name of that dem that I already pulled in.

192
00:28:20.540 --> 00:28:27.279
Ty Tuff, Ph.D.: I then transformed it into 2 different things, and then I said, What is the bounding box around that.

193
00:28:27.850 --> 00:28:29.339
Ty Tuff, Ph.D.: and

194
00:28:29.940 --> 00:28:38.430
Ty Tuff, Ph.D.: that founding box I can then use as my request. So if I'm I say, here are the. Here's the extent that I want to

195
00:28:38.540 --> 00:28:39.690
Ty Tuff, Ph.D.: to search

196
00:28:39.860 --> 00:28:52.450
Ty Tuff, Ph.D.: this one for the nothing is all of konis. So it's all of North America, even which is really big. And I made that request, and it returned like a hundred 10 GB of information

197
00:28:53.010 --> 00:29:06.949
Ty Tuff, Ph.D.: which was too big for almost everybody's computer. So instead, I did one with just Boulder County. So in the discovery environment. If you've made your discovery environment big enough, you could start doing really big stuff again. This is how big do you want your snake to be.

198
00:29:07.110 --> 00:29:10.439
Ty Tuff, Ph.D.: But right here you're feeding your snake a very, very big rat.

199
00:29:10.540 --> 00:29:26.020
Ty Tuff, Ph.D.: and here it's a much, much smaller rat. And so, just for demonstration. I wanted to break out and do a much smaller rat so that we can get through this thing a little bit better. So all I so get bounding box. This is a function through Openstreetmap.

200
00:29:26.340 --> 00:29:36.539

Ty Tuff, Ph.D.: I will show you a whole page that we have to aid you through getting openstream map data, and it's really easy. So right here, I've just given it boulder County

201
00:29:36.650 --> 00:29:52.309
Ty Tuff, Ph.D.: Boulder, Colorado gives you the whole county, and I've asked for it as a polygon, and it just gives me a straight polygon, and then I can transform it into the 2 projections I need, and get the bounding box of those new projections. I just need those values to make queries.

202
00:29:53.090 --> 00:29:55.999
Ty Tuff, Ph.D.: Here is, oh, I

203
00:29:56.460 --> 00:29:59.410
Ty Tuff, Ph.D.: sorry these should be next to each other. But,

204
00:29:59.820 --> 00:30:06.450
Ty Tuff, Ph.D.: This extent I essentially just did the same thing for the United States. I asked for a bounding box or a polygon of the United States.

205
00:30:06.470 --> 00:30:10.239
Ty Tuff, Ph.D.: and then cropped it down to the same size as that.

206
00:30:10.420 --> 00:30:12.750
Ty Tuff, Ph.D.: DEM,

207
00:30:14.360 --> 00:30:20.570
Ty Tuff, Ph.D.: okay, now, we're doing the good stuff. Okay? So stack. So the answer.

208
00:30:24.800 --> 00:30:34.870
Ty Tuff, Ph.D.: okay, I have a question about, is there easy way to keep track of memory usage and file size while working on the virtual machine. That's a really good question, Kelly, and I don't.

209
00:30:38.150 --> 00:30:40.859
Ty Tuff, Ph.D.: Yes, it's on the dashboard.

210

00:30:40.990 --> 00:31:05.220
Ty Tuff, Ph.D.: sort of in your discovery environment. I'm not gonna go to it right now. But in on the front page of your discovery environment. You can. You get a usage. One of the bars on the left is like performance. So you don't do it from within this, the discovery environment. Necessarily, you'd have to open up the second window and go to your opening dashboard in the discovery environment. And it it will tell you how much RAM how many resources you're using? I think?

211
00:31:05.860 --> 00:31:15.899
Ty Tuff, Ph.D.: But we should make that easier to use, especially with big data sets, because they can catch you by surprise cause on my laptop. I have to open my activity, monitor, and just watch the RAM go.

212
00:31:16.000 --> 00:31:21.179
you know, up and up, and up, and up and up as those things come in, because it doesn't even indicate it really well inside the code.

213
00:31:21.360 --> 00:31:23.300
Ty Tuff, Ph.D.: So really good question.

214
00:31:24.030 --> 00:31:28.190
Ty Tuff, Ph.D.: I wish I had a better answer for you. I can only give you that sort of sort of answer.

215
00:31:29.250 --> 00:31:30.069
Ty Tuff, Ph.D.: Not bad.

216
00:31:30.640 --> 00:31:43.069
Ty Tuff, Ph.D.: Okay, stack. This is the spatial temporal asset catalog. So this is essentially supposed to be a Dewey decimal system for spatial data sets.

217
00:31:43.340 --> 00:31:44.710
Ty Tuff, Ph.D.: And

218
00:31:45.830 --> 00:31:52.220
Ty Tuff, Ph.D.: it's it's not as good as a Dewey decimal system because there's a lot more chaos in all the data sets. But

219

00:31:52.260 --> 00:32:06.060
Ty Tuff, Ph.D.: it's really it's really good, and at least has a system. So first you are going to go to what's called a stack catalog. Here is just one stack catalog element 84, element 84 is a particularly nice

220
00:32:06.480 --> 00:32:14.439
Ty Tuff, Ph.D.: it's all on all these data, physically on aws which makes them cloud based and fast.

221
00:32:14.500 --> 00:32:25.969
Ty Tuff, Ph.D.: They're focused on Earth, their Earth data sciences. That websites, there is a V 0 that has more low level data, and that has slightly higher level data.

222
00:32:26.030 --> 00:32:30.190
Ty Tuff, Ph.D.: And we can go on to there and

223
00:32:30.370 --> 00:32:38.750
Ty Tuff, Ph.D.: request to look at the catalog. So the collection formats. Here are the thing. This is just within this one catalog. So you could potentially go and find

224
00:32:38.900 --> 00:32:49.719
Ty Tuff, Ph.D.: any other stack catalog you wanted and put in the address here and do the same command. And it'll list the the data sets that are available.

225
00:32:50.070 --> 00:33:03.830
Ty Tuff, Ph.D.: hey? So just within this one catalog, which is a particularly popular and open source. One. It has thing. So we have the chirps satellite, which is precipitation data, more precipitation data.

226
00:33:03.930 --> 00:33:14.440
Ty Tuff, Ph.D.: We have soil moisture. We have another soy moisture product. Here is

227
00:33:14.770 --> 00:33:19.989
Ty Tuff, Ph.D.: precipitation again, different different satellite precipitation.

228
00:33:20.140 --> 00:33:22.190
Ty Tuff, Ph.D.: Here is Landsat.

229
00:33:23.530 --> 00:33:36.709
Ty Tuff, Ph.D.: here's Landsat. Here's Maxar. here is the aqua or terra products. So they have broken these, though up into this one, is specifically the surface reflectance.

230
00:33:36.970 --> 00:33:49.689
Ty Tuff, Ph.D.: This is the modus snow cover. This is the modus Ls land service temperature. This is different bands of the land surface surface temperature.

231
00:33:50.750 --> 00:34:03.820
Ty Tuff, Ph.D.: Let's cruise down Ndvi from Modis. Here is plant scope and sentinel to data. We're going to deal with this quite a bit. In our examples.

232
00:34:04.580 --> 00:34:11.730
Ty Tuff, Ph.D.: They here's the for you fire folks. Here's the modus fire product. So all of these are available

233
00:34:12.909 --> 00:34:17.119
Ty Tuff, Ph.D.: house on aws in this one catalog for free. Okay?

234
00:34:17.320 --> 00:34:20.700
Ty Tuff, Ph.D.: So now we're back to framing our landscape.

235
00:34:23.600 --> 00:34:38.989
Ty Tuff, Ph.D.: So again, that catalog is the thing that we could point our camera at. But when you look at this picture right notice, there are trees that are telling one bit of story. There is a river that's telling a bit of story. There are mountains that are telling a bit of story. There's sky that's telling a bit of story.

236
00:34:39.020 --> 00:34:44.649
Ty Tuff, Ph.D.: Every part of this is coming in to help you tell the story that you want to tell with your data.

237
00:34:48.340 --> 00:34:51.790

Ty Tuff, Ph.D.: Eric? Just to know Eric's giving a

238
00:34:52.639 --> 00:34:55.410
Ty Tuff, Ph.D.: Eric, do you wanna hop in real quick and

239
00:34:56.080 --> 00:34:59.019
Ty Tuff, Ph.D.: talk about this finding out the memory limit?

240
00:34:59.860 --> 00:35:04.970
Erick Verleye: Oh, yeah, for those that are comfortable with using the terminal

241
00:35:05.650 --> 00:35:09.810
if you've used top or h top before

242
00:35:10.230 --> 00:35:18.739
Erick Verleye: but if you wanna open up a tab in your discovery environment and click on the little terminal application.

243
00:35:19.190 --> 00:35:24.169
Erick Verleye: All you have to do is run the command, tap all over, case.

244
00:35:25.550 --> 00:35:37.300
Erick Verleye: and then you will see you're running processes as well as very top. There's an MIB mem, line.

245
00:35:38.010 --> 00:35:40.929
Erick Verleye: It's gonna show you how much total RAM you have.

246
00:35:41.490 --> 00:35:47.739
Erick Verleye: and it's close to megabyte camera. It's like, maybe bytes or something like that.

247
00:35:48.600 --> 00:35:55.090
Erick Verleye: But that's going to show you how much total RAM you have and how much you're currently using.

248
00:35:55.470 --> 00:36:04.789

Erick Verleye: So some useful to look at. Why, you have processes running to see if you're quickly running out of RAM or something like that.

249
00:36:09.410 --> 00:36:16.849
Ty Tuff, Ph.D.: Thanks, Eric, appreciate it. For those of you who are following along. The new render is finally done. So if you just hit refresh.

250
00:36:17.140 --> 00:36:19.189
Ty Tuff, Ph.D.: it'll pop up the

251
00:36:19.840 --> 00:36:32.550
Ty Tuff, Ph.D.: the one I was hoping we were going through today. And the main thing that you're gonna see changed is that the Dms are gonna show up a little better. So here's the DM, here's the slope.

252
00:36:33.200 --> 00:36:41.360
Ty Tuff, Ph.D.: Here is aspect. So let me go back just a second to show you cause these weren't working first. So

253
00:36:42.650 --> 00:36:54.120
Ty Tuff, Ph.D.: this one went fast, or even 4.6 s it created the entire dem for North America. So again, that's mind-blowing here.

254
00:36:54.630 --> 00:37:03.299
Ty Tuff, Ph.D.: From that we then calculated, and it took a lot longer. It took 53 s to just calculate the slope. But then we had the slope.

255
00:37:03.980 --> 00:37:12.800
Ty Tuff, Ph.D.: and then we calculated the aspect, and these again are just for one function. This is in the terror package, and it's called terrain. And you just ask for the slope.

256
00:37:13.470 --> 00:37:17.509
Ty Tuff, Ph.D.: Now. boom, we can create our very first

257
00:37:19.610 --> 00:37:31.460
Ty Tuff, Ph.D.: data queue out of those 3. So we just concatenate those 3. We take the dem and the slope and the aspect. And we have

created our first data queue just out of those 3 things.

258
00:37:31.800 --> 00:37:34.340
Ty Tuff, Ph.D.: So that's a pretty simple data cube.

259
00:37:34.500 --> 00:37:44.789
Ty Tuff, Ph.D.: We don't need to save this right if you save this. This is really big, and it's probably slower moving it back from your memory back from your drive

260
00:37:44.950 --> 00:37:59.610
Ty Tuff, Ph.D.: back into your program. It's probably faster to just leave it as these 3 plugins that just re pull this thing each time you're about to use them. That's not always the case, but especially for these really big things. It could be a lot faster to just mount them.

261
00:37:59.830 --> 00:38:14.010
Ty Tuff, Ph.D.: Notice where I talk about the sources of the 3 layers. The first one is an actual tif. This is the one that we plugged into. So it's saying that first layer we made from tiff, but then that third, that second layer, the slope one we made from memory

262
00:38:14.140 --> 00:38:17.870
Ty Tuff, Ph.D.: and the aspect we built from memory. So it's it's

263
00:38:18.010 --> 00:38:21.230
Ty Tuff, Ph.D.: so we had to actualize those under our computer.

264
00:38:21.250 --> 00:38:23.290
Ty Tuff, Ph.D.: But then they built each other really fast.

265
00:38:24.820 --> 00:38:28.790
Ty Tuff, Ph.D.: Okay, now, let me catch back up with where I was at down here.

266
00:38:29.080 --> 00:38:38.529
Ty Tuff, Ph.D.: Okay, this has some of the description I was hoping was in there, in the first place, to talk about stack and talk about element 84 s. Or search on stack

267
00:38:39.160 --> 00:38:56.490
Ty Tuff, Ph.D.: And this is going back to Tyler's question is like, How do we know which things are set up for this? Well, Stack is is supposed to be the catalog of things that exist. You can go outside of stack people and find all these files, and that's the one that I showed you in the first one

268
00:38:56.560 --> 00:39:16.569
Ty Tuff, Ph.D.: that was one that was not part of a stack catalog, but you can go and just download it like anything else, and just see if they air out. You can also look at lists of vsi compatible. So a lot of places. If you go to their metadata they'll have a a thing saying, vsi just a column saying Dsi and yes or no, on whether it's compatible all the way down.

269
00:39:17.670 --> 00:39:34.439
Ty Tuff, Ph.D.: These are those collections that we look through. And we were here talking about deciding which data to bring in. And that is what. So the actualizing, this looking at a scene right and Ansel Adams. It was just pointing it towards the scene.

270
00:39:34.540 --> 00:39:38.240
Ty Tuff, Ph.D.: We have to code this in. So we are saying.

271
00:39:38.310 --> 00:39:46.710
Ty Tuff, Ph.D.: Okay, here is the address I want to pull from. Notice that I switch from V one in the early example to V. 0 in this example

272
00:39:46.920 --> 00:39:51.799
Ty Tuff, Ph.D.: it doesn't really matter. I'm just pulling from a slightly different catalog which would. And

273
00:39:52.180 --> 00:39:59.809
Ty Tuff, Ph.D.: if you look in the metadata. This has 22 million items. So it's sentinel to cogs, sentnel to data.

274
00:39:59.980 --> 00:40:11.350
Ty Tuff, Ph.D.: And there are 22 million items indicating way, way, way, way too big for our computer. There's just absolutely no way you could download this whole thing.

275

00:40:11.630 --> 00:40:22.499
Ty Tuff, Ph.D.: So the first thing we do is just make a virtual collection of the things that we might want to download. And this is what I'm saying is framing in your scene that you want to take a picture of.

276
00:40:22.910 --> 00:40:24.799
Ty Tuff, Ph.D.: So we're saying, Okay.

277
00:40:25.160 --> 00:40:28.520
Ty Tuff, Ph.D.: first, take S, which is my connection

278
00:40:28.690 --> 00:40:32.439
Ty Tuff, Ph.D.: to the stack. So here I've I've plugged in.

279
00:40:32.690 --> 00:40:39.790
Ty Tuff, Ph.D.: I've just sort of linked to the website, the stack website. I haven't done anything yet. I've just like, sort of pointed my computer to that.

280
00:40:40.450 --> 00:40:55.320
Ty Tuff, Ph.D.: This is the piping, this little piping function just says, Stack all these functions after it. So I plug in. And then I say, Okay, I want you to search everything in. I want you search within that catalog

281
00:40:55.460 --> 00:41:06.620
Ty Tuff, Ph.D.: for the collection that I want. I decided I wanted sentinel to and these are the L 2, a cogs. So

282
00:41:07.390 --> 00:41:32.260
Ty Tuff, Ph.D.: these, when a satellite collects its raw data. That's l. 0. Then that is sort of synthesized into a higher level abstraction, which is L one higher level, which is L 2 higher level, which is l. 3. So the higher this number, the more analysis has had has happened on those data before you're getting them. So L. 2 is pretty raw. You're just getting sort of. They've been spatially projected.

283
00:41:32.550 --> 00:41:33.710
Ty Tuff, Ph.D.: That's good. Nope.

284
00:41:33.990 --> 00:41:42.869
Ty Tuff, Ph.D.: here I made these bounding boxes earlier, saying, What is my area of interest. That's when we were having the area of interest discussion.

285
00:41:42.930 --> 00:41:56.890
Ty Tuff, Ph.D.: And so if you look through, I'm just giving the 4 different quadrants the top bottom, left and right of what that area of interest is. So what is my collection? I want to look for? Where is my area of interest? And what is my

286
00:41:57.370 --> 00:42:04.880
Ty Tuff, Ph.D.: my time span here? I'm just going for one day from the fifteenth of May to the sixteenth of May in 2,001,

287
00:42:05.460 --> 00:42:06.790
Ty Tuff, Ph.D.: and

288
00:42:06.870 --> 00:42:11.000
Ty Tuff, Ph.D.: post request is the send it off and see what they get.

289
00:42:11.370 --> 00:42:20.989
Ty Tuff, Ph.D.: And then this gives me a progress bar, so I can see how fast things are going. Okay. So I did that. I built this search

290
00:42:22.640 --> 00:42:27.550
Ty Tuff, Ph.D.: from that search. I then am going to assemble a collection.

291
00:42:34.510 --> 00:42:40.479
Ty Tuff, Ph.D.: The question is about, how do we get that list? It is a function that you call right here

292
00:42:43.390 --> 00:42:46.660
Ty Tuff, Ph.D.: called collection formats. So once I

293
00:42:47.100 --> 00:42:53.189
Ty Tuff, Ph.D.: put in this, get request. So I first did this line of code and that plugged in and got

294
00:42:53.220 --> 00:42:54.510
Ty Tuff, Ph.D.: got the stuff.

295
00:42:55.050 --> 00:42:59.570
Ty Tuff, Ph.D.: And then I ran this function to read that file that came in.

296
00:42:59.620 --> 00:43:06.289
Ty Tuff, Ph.D.: and that file tells me all of the possible things. So it's this collection, underscore formats

297
00:43:06.340 --> 00:43:07.770
Ty Tuff, Ph.D.: and

298
00:43:08.230 --> 00:43:17.670
Ty Tuff, Ph.D.: python. The python function is just a slightly different command here. But you're going to get the same thing. But again, we just said, Okay, I have this collection out there in the world.

299
00:43:18.550 --> 00:43:24.100
Ty Tuff, Ph.D.: Tell me what's in it. And then this is the list it returns from those

300
00:43:24.190 --> 00:43:28.770
Ty Tuff, Ph.D.: here is that sentinel. But no, here's the sentinel. 2 a

301
00:43:29.940 --> 00:43:30.940
Ty Tuff, Ph.D.: right there.

302
00:43:39.110 --> 00:43:46.160
Ty Tuff, Ph.D.: Okay, so back to here. First we searched. So this was, okay, what's in that catalog?

303
00:43:47.790 --> 00:43:51.639
Ty Tuff, Ph.D.: Second, we build our collection. So see this

304
00:43:51.840 --> 00:43:56.009

Ty Tuff, Ph.D.: stack image catalog. This is saying, okay, from

305
00:43:56.220 --> 00:44:02.300
Ty Tuff, Ph.D.: inside sentinel. And you have to go look at the metadata to find out all the things that you

306
00:44:02.550 --> 00:44:04.640
might want to pull from here.

307
00:44:05.020 --> 00:44:10.270
Ty Tuff, Ph.D.: You're saying, okay, the assets that are in sentinel. 2. 2. A.

308
00:44:10.560 --> 00:44:12.619
Ty Tuff, Ph.D.: It has a bunch of spectra.

309
00:44:13.480 --> 00:44:26.270
Ty Tuff, Ph.D.: So these are different bands of spectra from the multi spectral sensor on set sentinel to each of these are a different frequency, and they're just giving you a different reading on that band.

310
00:44:26.440 --> 00:44:32.090
Ty Tuff, Ph.D.: And then this is sort of a data quality metadata

311
00:44:32.260 --> 00:44:34.810
Ty Tuff, Ph.D.: field. And we're gonna use that later.

312
00:44:38.070 --> 00:44:43.550
Ty Tuff, Ph.D.: Thank you. Sibeli is correcting me send No. L 2. A is the surface reflectance data.

313
00:44:46.090 --> 00:45:08.500
Ty Tuff, Ph.D.: Okay. So here, we not correcting just adding actual surface reflectance.

314
00:45:08.680 --> 00:45:20.060
Cibele Amaral: because the levels that you you describe it before rightly. So I'm just like adding that information. Great. Thank you, Sabella. I appreciate it.

315
00:45:25.100 --> 00:45:34.830
Ty Tuff, Ph.D.: Sorry I muted myself. Okay, now, we're gonna actually build the collection. So this. So we search to find out what we what was available.

316
00:45:34.980 --> 00:45:42.189
Ty Tuff, Ph.D.: When that returned, we came and said, Okay, we want these assets. We're gonna build them into a collection.

317
00:45:42.370 --> 00:45:56.080
Ty Tuff, Ph.D.: And one thing that's cool about when you build your collection is right. Now is your first time to run big functions like actually do operations on that collection before we move forward.

318
00:45:56.170 --> 00:46:03.970
Ty Tuff, Ph.D.: So here, what I've done is so in my collection I first have taken item features

319
00:46:04.180 --> 00:46:16.959
Ty Tuff, Ph.D.: which are the things that were returned here. So item, I called that items. And so the things that were returned in my search. I'm going to give my search results to that collection.

320
00:46:17.350 --> 00:46:22.270
Ty Tuff, Ph.D.: hey? I'm saying, Okay, I'll here's my entire Google histories. There, my, go, Google, search for that thing.

321
00:46:22.820 --> 00:46:29.130
Ty Tuff, Ph.D.: I'm saying, okay, from those search results. Give me these assets that I've listed here

322
00:46:30.050 --> 00:46:44.999
Ty Tuff, Ph.D.: and then run this function. So I'm going to do a filter on those. So this is like, if you imagine a Google search, you've made a Google search. You've gotten a bunch of results. You then subset those results for your assets, and then you can subset again.

323
00:46:45.390 --> 00:46:51.190
Ty Tuff, Ph.D.: And so the sub setting we're doing here is saying, Oh,

man, there is a whole separate

324
00:46:51.360 --> 00:47:03.740
Ty Tuff, Ph.D.: cloud cover data set associated with this that you're not seeing in the assets, but it just each individual day is tagged with how much cloud cover was in that picture.

325
00:47:03.790 --> 00:47:19.699
Ty Tuff, Ph.D.: And so you can come and say, Well, only give me pictures, or any get rid of any pictures with. Oh, no. Only keep pictures with less than 20% cloud cover. So there's more than 20% cloud cover. Just throw the things away and don't even return them.

326
00:47:19.720 --> 00:47:42.469
Ty Tuff, Ph.D.: Okay, so we've given the search results. Subset. It subseted, and we bundle that all into our collection. And you can think about this now as the scene of landscape that we framed in. We haven't taken a picture. We have just decided what we want to look at what time we wanna date, what day we want to be there, what we're trying to capture. We've sort of built our scene.

327
00:47:43.510 --> 00:48:02.469
Ty Tuff, Ph.D.: and that can be Comp. That could be a long and complicated process in and of itself, as we saw from the David Yarro photo, right? Just getting the scene set up and deciding what you want to go in can take you a very long time. So give yourself patience. But actually running. This code only takes half a second.

328
00:48:06.810 --> 00:48:18.130
Ty Tuff, Ph.D.: Okay, so I have built that collection. And this is what it looks like. So this is an object. Now, in my code, I have this thing that I can pass to other things and need to pass it to other things.

329
00:48:18.810 --> 00:48:21.530
Ty Tuff, Ph.D.: And in there it just has

330
00:48:21.590 --> 00:48:24.829
Ty Tuff, Ph.D.: sort of this list of search results

331
00:48:24.910 --> 00:48:32.589
Ty Tuff, Ph.D.: that I have. I have found the scene that I'm looking

at now, we have to get our camera ready.

332
00:48:33.090 --> 00:48:37.440
Ty Tuff, Ph.D.: Okay, so here's a back to our Anzl Adams camera. Here is his film.

333
00:48:37.890 --> 00:48:46.239
Ty Tuff, Ph.D.: right? This. He is pouring chemicals on glass in the morning to get them to build the film. That is your end goal.

334
00:48:46.330 --> 00:48:58.999
Ty Tuff, Ph.D.: you as the scientist, you're trying to make a data cube at the end that can be flat like a piece of film, or that could be multiple layers. But the artistry is what hits the film. What's the very endpoint

335
00:48:59.240 --> 00:49:03.189
Ty Tuff, Ph.D.: the camera. This is our tool for modifying those data.

336
00:49:03.250 --> 00:49:18.240
Ty Tuff, Ph.D.: So in route coming towards us. We actually can change anything we want. Right? And a camera, you can put filters. You can put different lenses. You can run light around corners and in the camera. You can do all kinds of crazy things through mirrors.

337
00:49:18.300 --> 00:49:23.580
Ty Tuff, Ph.D.: And you can manipulate the data in lots and lots of ways before it hits your phone.

338
00:49:24.130 --> 00:49:33.589
Ty Tuff, Ph.D.: The old way of doing this would require you getting like lots and lots of film into your computer ahead of time. We're gonna try to just modify it while it's coming in. Okay.

339
00:49:33.760 --> 00:49:37.720
Ty Tuff, Ph.D.: this. So the Vsi is from Gdall.

340
00:49:38.060 --> 00:49:40.159
Ty Tuff, Ph.D.: G. Doll is

341

00:49:41.130 --> 00:49:50.099
Ty Tuff, Ph.D.: a computer package that has been around for very long time that runs almost anything you can imagine that you've ever done in spatial analysis.

342
00:49:50.330 --> 00:49:56.789
Ty Tuff, Ph.D.: So anytime you've loaded any spatial data or done anything like a union, or an extraction, or anything. This is Geo.

343
00:49:57.280 --> 00:50:08.670
Ty Tuff, Ph.D.: Now, Gdol has had this problem in the past, where you know the data would come into your hard drive, and then you go up into Python and you go through Gdall in Python.

344
00:50:08.690 --> 00:50:12.789
Ty Tuff, Ph.D.: And so again, you have to have all of that data coming in through the machine.

345
00:50:12.860 --> 00:50:25.289
Ty Tuff, Ph.D.: Which slows things down, and a lot of people, a lot of us spent a lot of frustrating hours trying to get to eat all the work in cloud environments in that environment, in that system. And it was really difficult.

346
00:50:25.540 --> 00:50:38.129
Ty Tuff, Ph.D.: And so Gdol, in response actually changed the way their software functions, and so they could put it on the server side. And so then, when I say, you can do anything that you have historically historically been able to do

347
00:50:38.230 --> 00:50:44.550
Ty Tuff, Ph.D.: in a Gis you can now do in route. That's because Gdl has essentially made a camera.

348
00:50:44.750 --> 00:50:56.860
Ty Tuff, Ph.D.: So when you're looking at this, think of this as a Gdb camera. It takes any information from that scene and modifies it in any way that you can imagine the Gdb would normally modify anything.

349
00:50:56.900 --> 00:51:01.709
Ty Tuff, Ph.D.: So this is, reproject it. This is change the extent. Now.

350
00:51:02.080 --> 00:51:13.560
Ty Tuff, Ph.D.: the collection, which is the scene we're looking at needs to be larger slightly than the picture you're going to take. If you can't see it. You can't take a picture of it

351
00:51:13.740 --> 00:51:14.900
Ty Tuff, Ph.D.: so

352
00:51:15.070 --> 00:51:43.929
Ty Tuff, Ph.D.: often an error that people find is that they will make their collection for one day, and then they will try to set up their camera to do a 2 day picture, and it'll just fry out and say, well, you didn't build your collection big enough to capture that picture. Okay, so this is just the conceptual rule is your landscape, which is that collection that you made from the cloud and assembled from your search results that needs to be larger than the I, the picture you're gonna take.

353
00:51:44.920 --> 00:51:45.830
Ty Tuff, Ph.D.: Okay?

354
00:51:46.540 --> 00:52:01.070
Ty Tuff, Ph.D.: So we set up. It's called a view window, a cube view window. and we set up a few things. Here's the spatial projection. So this is, I was telling you. There was some confusion with spatial projection here.

355
00:52:01.290 --> 00:52:10.969
Ty Tuff, Ph.D.: I gave it a bounding box with that really really popular spatial projection I was telling you about. but it returned those results

356
00:52:11.740 --> 00:52:13.690
Ty Tuff, Ph.D.: in a different

357
00:52:13.820 --> 00:52:18.910
Ty Tuff, Ph.D.: spatial resolution. And I just have this note that this is harder than expected.

358
00:52:19.200 --> 00:52:29.720

Ty Tuff, Ph.D.: because you don't usually know what projection they're giving you back. And so this can be a little bit of trial and error. A little bit of

359
00:52:30.140 --> 00:52:33.609
Ty Tuff, Ph.D.: yeah sort of getting things and seeing the projection they're in

360
00:52:33.700 --> 00:52:39.479
Ty Tuff, Ph.D.: but this, this can be a little tougher than you than expected. I would try to just use the code that I have

361
00:52:39.510 --> 00:52:53.880
Ty Tuff, Ph.D.: and see if it works. Okay. But again, we are just, we're setting out what our picture. We're setting the settings on our camera. So this is the projection would be? What is the exact morph of the film that that final maps eventually gonna be on?

362
00:52:54.370 --> 00:53:05.890
Ty Tuff, Ph.D.: How big is a pixel? So here I have 100 by 100. I can. You can go down to one by one or 10 by 10, or you can have them mismatched, and they don't have to match the original data set.

363
00:53:05.960 --> 00:53:14.330
Ty Tuff, Ph.D.: And this is again one of the steps that one of the sort of most magical steps here. because we can set this on the camera.

364
00:53:14.830 --> 00:53:19.070
Ty Tuff, Ph.D.: Now you go. Take a picture of 4 different data sets.

365
00:53:19.910 --> 00:53:29.410
Ty Tuff, Ph.D.: bring that information in and modify it. They're all standardized at the end. So a lot of us if we think back to the bad old days.

366
00:53:29.880 --> 00:53:44.349
Ty Tuff, Ph.D.: you sort of had to bring all 3 data sets in, transform them all in your machine. So they're on the same projection, and then they could go together. I have this question about, where do you find the Crs from? And

367
00:53:45.650 --> 00:54:06.929
Ty Tuff, Ph.D.: you essentially ask to use the Crs function, and it tells you what something is projected in. Usually here you have to fiddle around a little bit. These I found in the documentation. So when I say this is harder than expected, it's because there's not an easy way to just look this up from the data themselves. So

368
00:54:07.890 --> 00:54:25.439
Ty Tuff, Ph.D.: I think this took me several hours to sort of work through what the actual right one was so normally. If you have the thing in hand. It's pretty easy to find a Crs here. It can be a little trying, looking through documentation to try to find it. So let me try to find a better answer than that. All I can tell you right now is.

369
00:54:25.700 --> 00:54:30.370
Ty Tuff, Ph.D.: it's a little harder than you want it to be. but I'll I'll find. Try to find a better answer then.

370
00:54:31.490 --> 00:54:36.299
Ty Tuff, Ph.D.: So here's the size of the pixel. You can make this tiny, but

371
00:54:38.750 --> 00:54:51.629
Ty Tuff, Ph.D.: that can make this really big. So I guess this is again where the art comes from is like you have to set up your camera to be proportional to the film you're producing. So if I make this.

372
00:54:53.710 --> 00:55:03.900
Ty Tuff, Ph.D.: Okay, I have the this question about masking. We're not masking yet. Masking is one of the things that we can do in the operation. A mask would be something like

373
00:55:04.290 --> 00:55:18.679
Ty Tuff, Ph.D.: may maybe putting a tiny piece of tape across your lens so like can't get through that. So maybe if you have coastal data, and you want to mask out everything that's ocean. So it's just like an na rather than something else.

374
00:55:19.590 --> 00:55:23.839
Ty Tuff, Ph.D.: these are so you can have

375

00:55:24.550 --> 00:55:34.690
Ty Tuff, Ph.D.: let me pull up the documentation for this because you can specify the the question here in the chat was, What is Dx and DY.

376
00:55:34.700 --> 00:55:42.489
Ty Tuff, Ph.D.: This is essentially the as specified here. It's specifying how big, how many meters the pixel is.

377
00:55:42.570 --> 00:55:47.519
Ty Tuff, Ph.D.: But you can have it being dividing or specifying the exact amount.

378
00:55:47.610 --> 00:55:53.140
Ty Tuff, Ph.D.: me pull up the documentation.

379
00:56:06.780 --> 00:56:17.520
Ty Tuff, Ph.D.: Okay? So the function is called cube. So the Gdcubes is the library and queue view as the function. As we go down here.

380
00:56:21.280 --> 00:56:29.830
Ty Tuff, Ph.D.: you can use Nx as the number of pixels so based on the extent, how many pixels do you divide it up into.

381
00:56:30.080 --> 00:56:34.220
Ty Tuff, Ph.D.: or you can do the Dx to save the size of the pixels.

382
00:56:34.860 --> 00:56:36.390
And

383
00:56:37.440 --> 00:56:44.300
Ty Tuff, Ph.D.: I say so. The but the size quote unquote depends on the projection. So if you are in

384
00:56:44.570 --> 00:56:55.530
Ty Tuff, Ph.D.: meters, then this would be like size and meters. If you're in degrees, then it would be size and degrees. And so the projection sort of tells you a little bit about what your size is.

385
00:56:55.750 --> 00:56:58.080
Ty Tuff, Ph.D.: Okay? So

386
00:56:58.990 --> 00:57:02.890
Ty Tuff, Ph.D.: you're gonna want to probably spend some time familiarizing yourself with

387
00:57:03.870 --> 00:57:08.810
Ty Tuff, Ph.D.: these few settings and how they change your data.

388
00:57:09.130 --> 00:57:20.040
Ty Tuff, Ph.D.: But I think the key thing would be here is that once you set your cue view. Just have it the same for every picture you take, and then all of your outputs are automatically standardized.

389
00:57:20.310 --> 00:57:29.570
Ty Tuff, Ph.D.: So this can take a while to set up. This can be a little laborious, but once you get it, it's pretty nice because it standardizes all of your your outputs.

390
00:57:30.310 --> 00:57:36.009
Ty Tuff, Ph.D.: Okay? So this again takes a fraction of a second to build this cause. You're not

391
00:57:36.190 --> 00:57:42.410
Ty Tuff, Ph.D.: getting any information from the cloud. You're not doing any processing. You're just going and

392
00:57:42.690 --> 00:57:45.020
Ty Tuff, Ph.D.: building your little view finder camera.

393
00:57:46.690 --> 00:57:54.130
Ty Tuff, Ph.D.: There are 2 settings in here that are really important. It's how do you want to aggregate your space?

394
00:57:54.600 --> 00:57:55.650
Ty Tuff, Ph.D.: So

395
00:57:56.110 --> 00:58:05.390
Ty Tuff, Ph.D.: if this pixel sizes in your collection are different than your view. There, it's automatically going to reproject those to something different.

396
00:58:05.500 --> 00:58:13.990
Ty Tuff, Ph.D.: If it needs to combine multiple cells, multiple pixels into one. Pixel. What mathematical operation would you like them to do

397
00:58:14.410 --> 00:58:27.139
Ty Tuff, Ph.D.: the same with time? If you need the time to be shortened? Maybe you build a collection out of a day and a half, and then you only ask for a day, and there's some averaging on the borders, or you say, give me a monthly average

398
00:58:27.540 --> 00:58:29.170
Ty Tuff, Ph.D.: then.

399
00:58:29.350 --> 00:58:39.199
Ty Tuff, Ph.D.: And so here, the time that I've given they have some time codes. This is the monthly average. So this is the P. One is the average and across the month.

400
00:58:40.310 --> 00:58:44.460
Ty Tuff, Ph.D.: And so you're saying, Okay, how do you want to to deal with those?

401
00:58:52.110 --> 00:58:54.729
Elsa was making a

402
00:58:54.910 --> 00:58:57.009
Ty Tuff, Ph.D.: yeah, Elsa, do you want to make your point real quick?

403
00:58:58.120 --> 00:59:19.010
Elsa Culler: Yeah. Sure, I think that. Where I am. Students often run into trouble with that aggregation function is when you're dealing with something like land cover classes. That's categorical. So we have, like Class 4 and Class 13. And if we average those, then that's not gonna mean anything anymore. So

404
00:59:19.080 --> 00:59:21.950
Elsa Culler: you'll want to have

405
00:59:22.390 --> 00:59:32.309

Elsa Culler: an operation that gives you a whole number in that case, like the median. So yeah, just

406
00:59:32.380 --> 00:59:34.610
Elsa Culler: watch out for that with categorical data.

407
00:59:35.640 --> 00:59:38.869
Ty Tuff, Ph.D.: Yeah, the average of A and C is not too

408
00:59:40.930 --> 00:59:41.830
Elsa Culler: right?

409
00:59:42.570 --> 00:59:48.339
Ty Tuff, Ph.D.: Okay, perfect. Really, really. Good point. Okay? And now it's time to take the picture. Okay. So

410
00:59:48.410 --> 00:59:59.270
Ty Tuff, Ph.D.: we have done the hard parts. We have done, the setting our frame and setting up our collection. We have built our camera with the standardized output that we want in the end.

411
00:59:59.280 --> 01:00:04.700
Ty Tuff, Ph.D.: And now we take the picture. So what do we have to do? We have to give it our collection.

412
01:00:05.710 --> 01:00:20.730
Ty Tuff, Ph.D.: and then we are going to use this function called raster cube and give it our view. So our the V here is the camera that we built. So it's saying, take this collection and build a raster cube using that view finder.

413
01:00:21.130 --> 01:00:23.610
Ty Tuff, Ph.D.: And it takes the picture.

414
01:00:23.980 --> 01:00:28.320
Ty Tuff, Ph.D.: Okay. Now coming in. We have not hit the film yet

415
01:00:28.800 --> 01:00:35.889
Ty Tuff, Ph.D.: because of the beauty of the way the system works. You don't actualize until you do something like write or plot.

416
01:00:35.930 --> 01:00:58.419
Ty Tuff, Ph.D.: Okay? So with the remote connection, we take the collection, we build a virtual data cube. So we've now built the cube. This is the information coming through our camera. And now we can do operations on it before it gets to us. So this is where you get a ton of power of like getting a bunch of computation done before it hits your RAM. And so here I've calculated Ndvi.

417
01:00:58.480 --> 01:01:07.730
Ty Tuff, Ph.D.: So by selecting these 2 bands, sticking those bands into a function and giving the column a name. This is all happening while the light is coming through the camera.

418
01:01:08.110 --> 01:01:15.580
Ty Tuff, Ph.D.: Then, finally, I do not have to write this this was to save. So now that we want us. If we want to save this.

419
01:01:15.670 --> 01:01:17.440
Ty Tuff, Ph.D.: we can make it into tiffs.

420
01:01:17.860 --> 01:01:24.049
Ty Tuff, Ph.D.: Here. Oh, this is just me showing you specifically how to make a raster stack. So we're gonna convert.

421
01:01:24.130 --> 01:01:36.829
Ty Tuff, Ph.D.: We're gonna so those things that light is now actualized on the film as Tif's. and we stack those into a raster stack, and we get a raster stack out.

422
01:01:36.850 --> 01:01:47.689
Ty Tuff, Ph.D.: and that whole process there, right? So this is for the for Boulder County for the entire county. It took. Nope, well, did we do counting here?

423
01:01:47.800 --> 01:01:49.880
Ty Tuff, Ph.D.: Here? We do. Let me. So let me check.

424
01:01:55.200 --> 01:01:58.039
Ty Tuff, Ph.D.: Yeah, we did Boulder. Yep. So for the county

425
01:01:58.310 --> 01:02:06.879
Ty Tuff, Ph.D.: it took 4 min. So that was 4 min to pull all of the hypers, all of the spectral data. the outline of reflectance data

426
01:02:06.930 --> 01:02:11.879
Ty Tuff, Ph.D.: and calculate ndvi and write it to tiff and stack. It

427
01:02:12.170 --> 01:02:22.010
Ty Tuff, Ph.D.: all took 4 min. Okay, that it. So there are 2 different formats. You can save as there is raster style format. So this is.

428
01:02:22.360 --> 01:02:29.780
Ty Tuff, Ph.D.: this requires 3 dimensionality. This requires it to be a cube shape. So you're going to have grid of data stacked.

429
01:02:29.880 --> 01:02:34.319
Ty Tuff, Ph.D.: and those gridded data are going to be in Tif format.

430
01:02:36.000 --> 01:02:45.359
Ty Tuff, Ph.D.: But you've also there's a more flexible version called Stars in R, this, I think, is Tsar in python.

431
01:02:45.510 --> 01:02:55.319
Ty Tuff, Ph.D.: And this allows you to do different types of data. So somebody asked last week about combining vector point deck Beta, vector, vector, data and raster data.

432
01:02:55.360 --> 01:02:57.150
Ty Tuff, Ph.D.: You cannot do those

433
01:02:57.440 --> 01:03:10.609
Ty Tuff, Ph.D.: as raster without actually converting all of those different formats to a raster. But stars and Czar. Let you have more flexibility. So you can actually put you can have more than 3 dimensions.

434
01:03:10.640 --> 01:03:14.440
Ty Tuff, Ph.D.: And you can have different types of data stacked on top of each other.

435
01:03:15.800 --> 01:03:23.779
Ty Tuff, Ph.D.: Okay, let me show you, because I'm running out of time. Let me make sure we get through the last couple points, extracting data. So from that same collection.

436
01:03:23.930 --> 01:03:34.220
Ty Tuff, Ph.D.: from that same view I select bands. This is all virtual, just saying, like, out of that cube that was created. These are the bands that I want.

437
01:03:34.490 --> 01:03:45.459
Ty Tuff, Ph.D.: And now I can do this extract. GM, so this is I'm gonna take that polygon of Boulder County and put it over. And I'm gonna extract the values out of every single pixel

438
01:03:46.930 --> 01:03:56.160
Ty Tuff, Ph.D.: here. It's just renaming those because they are spectra. These were the spectra, the band names. And I wanted to put what the actual frequency was.

439
01:03:57.450 --> 01:04:03.460
Ty Tuff, Ph.D.: Okay, else is correcting me. The X-ray can do the A bunch of informats sources.

440
01:04:03.650 --> 01:04:29.399
Ty Tuff, Ph.D.: okay, it was okay. So I extracted those. I only print the very top. But this created a 2 dimensional data frame. So this is really, really long. This is a list of every single pixel what time we collected, and then what the measurement was for every single pixel. And so now you can go and plant print those bands.

441
01:04:29.580 --> 01:04:37.060
Ty Tuff, Ph.D.: This only took 1.7 min because I didn't write it right this one up here where it took 4 min.

442
01:04:38.610 --> 01:04:55.720
Ty Tuff, Ph.D.: That's because we decided to write it to our hard drive, and as soon as you write it to your hard drive now you were slowing the whole process down, so it took 4 min because I decided to write it down here where I don't write it. It only takes 1.7 min, so I can make it twice as fast. If I just

443
01:04:55.920 --> 01:04:58.710
Ty Tuff, Ph.D.: don't need to actually write this thing to disk all the time.

444
01:05:00.230 --> 01:05:13.079
Ty Tuff, Ph.D.: Here we can do a whole time series. So the last one I showed you was just one day here. I want a long time series. So it's from 2,020 to 2,022.

445
01:05:14.380 --> 01:05:29.320
Ty Tuff, Ph.D.: There is a function that I don't have in here, which is called reduced time, which is another one you can add in here. If you want to do more complicated time reductions than I've done. I just I still have that one month average thing built into the camera view

446
01:05:29.680 --> 01:05:41.079
Ty Tuff, Ph.D.: just down here. The PP. One M. This is the in my camera view. I'm just specifying what the timelist is, but even after you've taken the picture, if you need to reduce it more. You can do that.

447
01:05:41.520 --> 01:05:43.730
Ty Tuff, Ph.D.: So here, let's just

448
01:05:43.860 --> 01:05:48.689
Ty Tuff, Ph.D.: take another picture exactly like I did before. Here's my raster cube function.

449
01:05:48.770 --> 01:05:56.129
Ty Tuff, Ph.D.: Select those bands click. I calculate the ndvi exactly like you had before. It's set. Now let's animate it

450
01:05:56.530 --> 01:05:57.770
Ty Tuff, Ph.D.: as a gif.

451
01:05:58.650 --> 01:06:04.460
Ty Tuff, Ph.D.: And so here, let's see how long. This took real quick before we go look at it. So this took almost 5 min.

452

01:06:04.600 --> 01:06:11.309
Ty Tuff, Ph.D.: but that was writing a plot for every month. So we get the ndvi

453
01:06:12.810 --> 01:06:16.370
Ty Tuff, Ph.D.: animated over the course of 2 full years.

454
01:06:16.860 --> 01:06:27.309
Ty Tuff, Ph.D.: That gray in the beginning is right when the sensor came on, so I found the very first instance of the sensor coming across. And it only that month came across that little bottom corner wedge.

455
01:06:31.610 --> 01:06:37.480
Ty Tuff, Ph.D.: Okay, when you want to write when you do want to finally save something

456
01:06:37.490 --> 01:06:42.359
Ty Tuff, Ph.D.: you can save in 2 different formats. net cdf.

457
01:06:42.400 --> 01:06:48.759
Ty Tuff, Ph.D.: or that tiff. So I showed you above where it was. Write tiff, and then another way to do it is write net. Cdf.

458
01:06:49.190 --> 01:06:59.070
Ty Tuff, Ph.D.: they also have different compression levels that you can choose. If you're having trouble finding the memory, finding the disk space to actually save these huge things that you're sucking in

459
01:06:59.940 --> 01:07:10.750
Ty Tuff, Ph.D.: and then this was yeah. it's pretty easy to. So here I've done one collection for 2020,

460
01:07:11.340 --> 01:07:23.359
Ty Tuff, Ph.D.: right? I'm again searched, built the collection of sentinel data with the bounding box with a time period, and asked for that data and got some data, got some items back that were just 2020,

461
01:07:23.720 --> 01:07:30.389
Ty Tuff, Ph.D.: just everyone for 2021 ran both of those processes in parallel that you saw before.

462
01:07:31.900 --> 01:07:38.070
Ty Tuff, Ph.D.: Calculate and dvi, just like I did before. And now you can just subtract them.

463
01:07:39.540 --> 01:07:49.810
Ty Tuff, Ph.D.: So within the Stars Library they just. They have things like raster calculators where you can just subtract out. Here's the actual sorry down here.

464
01:07:50.350 --> 01:07:54.289
Ty Tuff, Ph.D.: So you just subtract one versus the other. And now you can get Max.

465
01:07:54.590 --> 01:07:59.189
Ty Tuff, Ph.D.: difference in ndvi between the years, but these objects, again, are completely.

466
01:07:59.240 --> 01:08:06.750
Ty Tuff, Ph.D.: virtually plugged in, but you can bring them into all kinds of functions that would you normally use and use them as if they already exist on your computer.

467
01:08:07.940 --> 01:08:09.600
Ty Tuff, Ph.D.: Okay, so that

468
01:08:10.930 --> 01:08:15.079
Ty Tuff, Ph.D.: has burned up almost all my time. I have 5 min left.

469
01:08:15.150 --> 01:08:20.859
Ty Tuff, Ph.D.: And with that time I'm going to show you some of the data that I've made available to you with some code.

470
01:08:21.160 --> 01:08:31.280
Ty Tuff, Ph.D.: the ones right here on the sidebar, you can see, are really flood specific. And most of these only have our code. Right now I'm going to go through those in just a second.

471
01:08:31.470 --> 01:08:36.190
Ty Tuff, Ph.D.: let me put another link up, though this is our data

library

472
01:08:36.560 --> 01:08:38.509
Ty Tuff, Ph.D.: from the summit.

473
01:08:45.109 --> 01:08:51.060
Ty Tuff, Ph.D.: and then II see, Tyler. I see your question. Let me go through a couple of data sources, and I'll try to answer your question on my way out the door.

474
01:08:51.330 --> 01:08:57.440
Ty Tuff, Ph.D.: Okay? So first, if you go to this on the sidebar. You're gonna see a lot of

475
01:08:57.630 --> 01:09:12.930
Ty Tuff, Ph.D.: different data sets organized in groups related. The organization doesn't make a lot of sense for this context. But don't worry about it right now. But in here you can find some. This is an example. Native lands. Digital.

476
01:09:13.460 --> 01:09:15.180
Ty Tuff, Ph.D.: This is how to get

477
01:09:15.899 --> 01:09:27.389
Ty Tuff, Ph.D.: polygons of all of the native tribal land around the globe really easily. And the reason I want to show you this is most of these. We did Python, and our

478
01:09:27.470 --> 01:09:31.790
Ty Tuff, Ph.D.: not all of them. But a lot of these have Python and our solutions in them.

479
01:09:31.979 --> 01:09:42.879
Ty Tuff, Ph.D.: so feel free to look through these in terms of giving you lots of different data sets that you could potentially put in your landscape and ways to download them.

480
01:09:43.310 --> 01:09:47.510
Ty Tuff, Ph.D.: As we go back to the data sets specifically for the hackathon.

481
01:09:47.590 --> 01:09:49.410
Ty Tuff, Ph.D.: One is

482
01:09:49.750 --> 01:09:55.970
Ty Tuff, Ph.D.: a flood inventory. So this is just a long list of all the floods that have happened

483
01:09:56.040 --> 01:10:05.000
Ty Tuff, Ph.D.: where they have happened and what their cause was. So here is point data where those data, those floods happened. and

484
01:10:08.020 --> 01:10:15.240
Ty Tuff, Ph.D.: here they have how many dad died or were displaced. And so I did some plots on just the relatedness of

485
01:10:15.290 --> 01:10:29.690
Ty Tuff, Ph.D.: dead and displaced. As you got bigger. you know, something like we have a lot of low frequency floods only a few high severity, and not very many super high severity. But when you get to really high severity, you get a little bit more higher, higher levels of displacement.

486
01:10:30.130 --> 01:10:35.230
Ty Tuff, Ph.D.: So the other thing I show you here is how to make a time series.

487
01:10:35.270 --> 01:10:47.020
Ty Tuff, Ph.D.: So things like breaking down the composite time series of flood data. Trying to D trend them and find out. Oh, well, there really seems to be a pretty strong seasonality to when floods happen.

488
01:10:47.370 --> 01:10:50.450
Ty Tuff, Ph.D.: Here is future projections of floods.

489
01:10:51.680 --> 01:11:03.990
Ty Tuff, Ph.D.: Okay. flood event polygons. So this is way to actually get the polygon of different flood areas or different severities or different numbers of that are displaced.

490
01:11:04.700 --> 01:11:13.969

Ty Tuff, Ph.D.: If you need river geography. So you actually want to pull just the vector layer for a particular river. You're looking at. You can do that from Openstreetmap here at Lakes.

491
01:11:14.050 --> 01:11:16.589
Ty Tuff, Ph.D.: and then you can combine the rivers and lakes together.

492
01:11:17.890 --> 01:11:26.640
Ty Tuff, Ph.D.: Hydro basins. The original data set. I the original website. I showed you at the very beginning of the lesson. It has a lot of great data on.

493
01:11:27.590 --> 01:11:28.850
Ty Tuff, Ph.D.: So

494
01:11:30.940 --> 01:11:32.330
Ty Tuff, Ph.D.: see.

495
01:11:33.100 --> 01:11:46.389
Ty Tuff, Ph.D.: this is like. What the order is like. How high is it in the tributary in the, in the watershed? So here our headwaters all the way down to tail waters.

496
01:11:49.580 --> 01:11:52.230
Ty Tuff, Ph.D.: The DM. We looked at before.

497
01:11:54.230 --> 01:11:58.000
Ty Tuff, Ph.D.: here is one where.

498
01:11:58.330 --> 01:12:08.249
Ty Tuff, Ph.D.: So I showed you that some of those original data could only come in at 15 to seconds. If you want to do higher, you want to do those 3 s ones.

499
01:12:08.310 --> 01:12:17.189
Ty Tuff, Ph.D.: You can do it through a slightly different system. You just have to tile them together. And so this is me showing you how to pull the individual tiles if you need the higher resolution staff.

500

01:12:23.500 --> 01:12:37.199
Ty Tuff, Ph.D.: again, these are showing you different information about the basins. Different information about. So this is not necessarily flood data. This is, these are great river data where water would flow, data, how to calculate.

501
01:12:37.460 --> 01:12:40.119
Ty Tuff, Ph.D.: How bad a flood is those sorts of things.

502
01:12:40.340 --> 01:12:53.360
Ty Tuff, Ph.D.: Okay, neon has a bunch of great lake datasets. The thing I want to show you here. And so to get back to Tyson's question about

503
01:12:53.640 --> 01:12:57.670
Ty Tuff, Ph.D.: what is sort of the ideal structure of a data queue. Okay.

504
01:12:58.670 --> 01:13:05.710
Ty Tuff, Ph.D.: this sort of this again depends on the question you're answering. And that picture you want to take.

505
01:13:05.720 --> 01:13:08.000
Ty Tuff, Ph.D.: So here we have neon data.

506
01:13:08.020 --> 01:13:15.510
Ty Tuff, Ph.D.: You see, there. They only go back to 2,015, but they cover a bunch of stuff and they cover a bunch of sites.

507
01:13:15.530 --> 01:13:17.279
Ty Tuff, Ph.D.: And that's wonderful.

508
01:13:17.420 --> 01:13:26.410
Ty Tuff, Ph.D.: Now, if you combine that with Ltr data here. Ltr data, they only cover a couple of places, but they do it for a really long time.

509
01:13:26.700 --> 01:13:31.700
Ty Tuff, Ph.D.: Neon data cover a bunch of places with really high fidelity, but for only for a little bit of time.

510
01:13:33.020 --> 01:13:39.169
Ty Tuff, Ph.D.: And so fitting those 2 together into an ideal data. Queue

511
01:13:39.240 --> 01:13:49.139
Ty Tuff, Ph.D.: actually has a bunch of philosophical questions, not structural questions. It's sort of well, do I want to make the queue this whole area and have all this blank?

512
01:13:49.460 --> 01:13:54.309
Ty Tuff, Ph.D.: Or do I want to just look at this part and have just compare the data here.

513
01:13:54.390 --> 01:14:01.959
Ty Tuff, Ph.D.: Sort of how, how philosophically do I compare those 2 data types into something that's a queue.

514
01:14:02.040 --> 01:14:04.090
Ty Tuff, Ph.D.: Now, what is the ideal cube

515
01:14:05.440 --> 01:14:11.129
Ty Tuff, Ph.D.: computer? AI, the thing that you're gonna plug in to look at this once

516
01:14:11.310 --> 01:14:14.660
Ty Tuff, Ph.D.: a completely uniform, completely filled in structure.

517
01:14:15.050 --> 01:14:23.699
Ty Tuff, Ph.D.: So anything, any parts of that queue, that sort of jet off into some new direction are hard to interpret on. Why, that would be.

518
01:14:23.800 --> 01:14:34.270
Ty Tuff, Ph.D.: It's best for an AI to be able to look at this cube and say, Okay, as I scan it, different directions, I see different things changing in this way, and I can gain different inference from that.

519
01:14:34.440 --> 01:14:40.149
Ty Tuff, Ph.D.: So ideally. You want a queue queue.

520
01:14:40.190 --> 01:14:49.659
Ty Tuff, Ph.D.: and that is our goal. But you can see how hard that is. There's so many decisions, so many compromises that have to go into creating

521
01:14:49.690 --> 01:14:51.939
Ty Tuff, Ph.D.: that cube shape thing at the end.

522
01:14:52.350 --> 01:15:12.250
Ty Tuff, Ph.D.: okay, there was a question about environmental data to predict water quality. Yeah, there are a couple of those. So in the neon river ones, they have sensors above and below a bunch of their sites. And so you can get things like dissolved oxygen and dissolved carbon and things like that.

523
01:15:12.450 --> 01:15:28.010
Ty Tuff, Ph.D.: From the neon sites. The EPA. Here's a water, a water quality data portal. So this way you can get things like how much ammonia was there. And this is one where there is our and Python code, that to get this

524
01:15:28.490 --> 01:15:36.130
Ty Tuff, Ph.D.: Usgs water services, these are the places to go generally to find out what the flow was, how much flow was there over time? And when were the peaks?

525
01:15:37.080 --> 01:15:42.170
Ty Tuff, Ph.D.: This one's broken? I'll fix that

526
01:15:42.200 --> 01:15:48.190
Ty Tuff, Ph.D.: The species occurring uses I, naturalist. There's a package that helps you pull species occurrence data

527
01:15:48.930 --> 01:15:59.769
Ty Tuff, Ph.D.: neon this goes to an external link, but they show you how to get the Lidar data from neon. So if you wanted to look at sort of physical structure beyond what the DM can give you.

528
01:16:00.380 --> 01:16:06.609
Ty Tuff, Ph.D.: and neon biochemistry also has sort of these are the

terrestrial

529
01:16:06.680 --> 01:16:27.419
Ty Tuff, Ph.D.: ones to match the river once. So if you wanted to say, Okay, we see a ton of carbon, we see a huge nitrogen pulse in this river on this date. You could then compare it to the terrestrial measurements before that to say, Okay, yeah. In this flood it swept a bunch of that nitrogen off of the surface and pushed it into water.

530
01:16:27.840 --> 01:16:34.740
Ty Tuff, Ph.D.: Openstreetmap. We already play with that a tiny bit. This just shows you how to get those layers for open stream. Map

531
01:16:34.850 --> 01:16:38.010
Ty Tuff, Ph.D.: us census, if you want to find out.

532
01:16:38.150 --> 01:16:40.249
Ty Tuff, Ph.D.: You know

533
01:16:41.090 --> 01:16:50.110
Ty Tuff, Ph.D.: sort of poverty estimates or demographics on who lives there, or how long they live there, how many grocery stores they have?

534
01:16:50.450 --> 01:17:00.060
Ty Tuff, Ph.D.: And then the remote sensing one is the sentinel stuff that I showed you in the main lesson. This is just a longer, more involved version of those sentinel data.

535
01:17:00.220 --> 01:17:03.600
Ty Tuff, Ph.D.: So I think that I have burned through all of my time.

536
01:17:05.190 --> 01:17:15.310
Ty Tuff, Ph.D.: I'd like to give people like 2 or 3 min to ask questions before I pop out of the way. But I also want to reassure everybody that I know. We cruise through this really quickly.

537
01:17:15.310 --> 01:17:35.339
Ty Tuff, Ph.D.: just trying to give you the philosophy of what we're doing. Give you a little bit of tools to point you in the right direction. But then, have you go and sort of play with these throughout the hackathon, and know that we'll be around to help you

get them set up and help work with things when you have a problem. So let me open up for questions for about 5 min, and then I need to pass off the torch to our next fearless leader.

538
01:17:55.140 --> 01:18:16.839
Ty Tuff, Ph.D.: Hey, Tyler is asking about the difference between Zeros and Nas, and Na is filling the space so and usually na's are preferable if, unless the zeros are confirmed. So if most data scientists, when they see a 0, think that you have confirmed. That is 0. That 0 is the actual value of that data.

539
01:18:16.840 --> 01:18:27.540
Ty Tuff, Ph.D.: not of that datum. Not that those data are missing, so don't put a 0 unless it's confirmed as the 0, and otherwise put an na, because that completely fills the space.

540
01:18:35.990 --> 01:18:43.380
Ty Tuff, Ph.D.: Alright. Let's go take a break. Let's give everybody a bio. Break 5 min. Let me just pass it off to Nate and let him run the

541
01:18:43.560 --> 01:18:46.450
Ty Tuff, Ph.D.: run the breaking. But thanks, everybody. I'm around.

542
01:18:46.470 --> 01:18:52.770
Ty Tuff, Ph.D.: We'll see at the hackathon, and try to answer all the questions, and deal with all your panic as much as we can then.

543
01:18:53.040 --> 01:19:05.249
Nate Quarderer (Earth Lab/ ESIIL): Yes, yes, thank you, Ty, give it up for Ty, our very own friendly neighborhood data scientist. We are lucky to have you, my friend. Yes, thank you. The great job.

544
01:19:05.530 --> 01:19:16.610
Nate Quarderer (Earth Lab/ ESIIL): Remember, this is all recorded. And so you can go back and watch that stuff, and we're gonna be here to help you during the hackathon, too. So don't think like you have to have absorbed all of that information that Ty just gave you.

545
01:19:16.730 --> 01:19:30.780
Nate Quarderer (Earth Lab/ ESIIL): because we've got resources for you. So give Ty a big shout out, why don't we do this? It's 1023 mountain time on my clock. What do you think? Rachel and Virginia go

until 103-01-0350n a break. How do you feel?

546
01:19:31.230 --> 01:19:34.990
Nate Quarderer (Earth Lab/ ESIIL): Make sure we want to give enough time?

547
01:19:35.860 --> 01:19:37.169
Rachel Lieber: 1033,

548
01:19:37.420 --> 01:19:45.099
Nate Quarderer (Earth Lab/ ESIIL): 1033. I love it. 1033 mountain time we'll come back and we'll get started with the next piece.

549
01:19:45.450 --> 01:19:46.679
Nate Quarderer (Earth Lab/ ESIIL): Thanks again, Ty.

550
01:19:48.060 --> 01:19:50.699
Ayoub Ghriss: so the first thing we would do

551
01:19:53.020 --> 01:19:59.059
Ayoub Ghriss: I will send my guitar repo in the chat, and you will have to go and loan it.

552
01:20:02.810 --> 01:20:04.240
Ayoub Ghriss: Okay.

553
01:20:04.320 --> 01:20:05.910
Ayoub Ghriss: let me start here.

554
01:20:28.890 --> 01:20:29.970
Ayoub Ghriss: Great

555
01:20:40.870 --> 01:20:42.020
Ayoub Ghriss: a

556
01:20:43.980 --> 01:20:46.490
Ayoub Ghriss: 1 s. Here. I just want to grab it.

557

01:20:52.680 --> 01:20:55.529
Ayoub Ghriss: The window is still all right. Is it better this way?

558
01:20:58.410 --> 01:20:59.150
Ayoub Ghriss: Okay.

559
01:21:07.770 --> 01:21:10.629
Ayoub Ghriss: so I sent a link on the chat, and

560
01:21:10.750 --> 01:21:18.339
Ayoub Ghriss: you just have to follow my the steps I'm doing now. I guess you might be really familiar with that, just in case.

561
01:21:50.230 --> 01:21:52.520
Ayoub Ghriss: So the first thing we want to do

562
01:21:53.780 --> 01:21:57.280
Ayoub Ghriss: while I'm doing, the presentation

563
01:21:57.820 --> 01:22:03.860
Ayoub Ghriss: is to execute the script, repair environments. So this is going to install the packages that you are going to use.

564
01:22:03.920 --> 01:22:10.520
Ayoub Ghriss: So just do bash repair underscore environments.

565
01:22:16.370 --> 01:22:21.439
Ayoub Ghriss: It's gonna take some time, but we don't have to wait for it. Once it starts again, we'll open.

566
01:22:34.450 --> 01:22:37.310
Ayoub Ghriss: You have questions, feel free to send them on the chat.

567
01:22:38.370 --> 01:22:40.140
Ayoub Ghriss: yeah.

568
01:22:41.680 --> 01:22:44.359
Ayoub Ghriss: I'll try to answer. If there are

569
01:22:44.680 --> 01:22:47.980
Ayoub Ghriss: irrelevant to what I am doing. If not, I'm going to answer them later.

570
01:22:50.250 --> 01:22:57.470
Elsa Culler: I just A quick question is the intention that people be following along with this or watching

571
01:22:57.740 --> 01:23:00.140
Elsa Culler: you, you know later?

572
01:23:00.780 --> 01:23:03.029
Ayoub Ghriss: Yeah, they can do them

573
01:23:03.200 --> 01:23:06.430
Ayoub Ghriss: who wishes keep up or just watch. Seems likely will keep up.

574
01:23:07.440 --> 01:23:20.800
Elsa Culler: Okay. So if if folks are following along, then you can use the github extension over on the left hand side to clone that repository. If you didn't catch the command.

575
01:23:20.890 --> 01:23:25.930
Elsa Culler: It's like a little Github symbol all the way on the left. Yeah, or in the get menu.

576
01:23:26.690 --> 01:23:27.390
Ayoub Ghriss: Okay?

577
01:23:30.590 --> 01:23:37.490
Ayoub Ghriss: Alright. So I'm going to start while this is done. and I'll go ahead.

578
01:23:41.170 --> 01:23:46.500
Ayoub Ghriss: Okay. So I'm just gonna do. Yes, here I'll let it do its thing.

579
01:24:12.940 --> 01:24:13.600

Ayoub Ghriss: It

580
01:24:16.060 --> 01:24:17.659
Ayoub Ghriss: so this is going to be

581
01:24:17.690 --> 01:24:23.609
Ayoub Ghriss: a dense but quick presentation, just to give you a flavor of what machine learning is.

582
01:24:23.690 --> 01:24:26.840
Ayoub Ghriss: And at the end it's gonna be more like a

583
01:24:32.160 --> 01:24:35.400
Ayoub Ghriss: discovered environment. Should we be using?

584
01:24:43.090 --> 01:24:44.529
Ayoub Ghriss: You see that again?

585
01:24:48.190 --> 01:24:50.880
Ayoub Ghriss: We want to keep an opportunity. It's

586
01:24:51.620 --> 01:24:55.860
Ayoub Ghriss: yeah, just gonna be Jupiter one. But we're not. We're not using it for now.

587
01:24:59.280 --> 01:25:00.040
Ayoub Ghriss: Okay.

588
01:25:03.360 --> 01:25:07.950
Ayoub Ghriss: we've already started that we are going to resume that later, when the packages are being installed.

589
01:25:09.200 --> 01:25:10.320
Ayoub Ghriss: So the

590
01:25:16.890 --> 01:25:19.120
Ayoub Ghriss: yep. So.

591

01:25:19.760 --> 01:25:28.440
Ayoub Ghriss: The theme was more like artificial intelligence. But we are going to talk about a subset of artificial intelligence. The main difference is that

592
01:25:28.750 --> 01:25:31.650
Ayoub Ghriss: artificial intelligence is a more.

593
01:25:34.950 --> 01:25:37.000
Ayoub Ghriss: The black screen is just me.

594
01:25:37.670 --> 01:25:38.930
Elsa Culler: Yeah, it's not.

595
01:25:39.200 --> 01:25:45.209
Elsa Culler: Yeah. I also just see a blank screen. Maybe you didn't yet select the

596
01:25:46.020 --> 01:25:47.700
Elsa Culler: window you were sharing.

597
01:25:48.130 --> 01:25:50.019
Ayoub Ghriss: I'm doing share screen.

598
01:25:59.480 --> 01:26:01.040
Ayoub Ghriss: Jay, 1 s.

599
01:26:55.490 --> 01:26:58.940
Ayoub Ghriss: Okay, sorry for that. Let me just move to something else.

600
01:27:50.370 --> 01:27:52.810
Ayoub Ghriss: Hey? I'm just gonna use my.

601
01:27:54.260 --> 01:27:56.069
Ayoub Ghriss: it's good up for everything, right?

602
01:28:00.110 --> 01:28:02.159
Ayoub Ghriss: So in the meantime.

603
01:28:02.350 --> 01:28:09.659
Virginia Iglesias: Elsa, would you show us how to clone the repo one more time so that everybody can follow.

604
01:28:10.980 --> 01:28:20.309
Elsa Culler: Yeah, yeah, 100%. So we're going to head to the discovery environment here.

605
01:28:20.560 --> 01:28:30.230
Elsa Culler: let's see, I've already got this analysis. But if you're on the discovery environment which

606
01:28:30.320 --> 01:28:33.450
Elsa Culler: I'll put this link in the chat in case you've

607
01:28:33.670 --> 01:28:35.340
Elsa Culler: forgotten

608
01:28:36.850 --> 01:28:42.250
Elsa Culler: de cybers.org you're gonna log in

609
01:28:42.440 --> 01:28:50.729
Elsa Culler: and you're gonna go to the apps page. And probably any of these Jupiter labs will work. But we're gonna go with this Jupiter lab Earth lab

610
01:28:54.010 --> 01:28:57.499
Elsa Culler: and click through

611
01:28:57.840 --> 01:29:07.120
Elsa Culler: here. and then once you have click through, you will have, like I do an analysis running here.

612
01:29:08.580 --> 01:29:11.209
Elsa Culler: So I'm going to go to that analysis.

613
01:29:18.190 --> 01:29:24.250
Elsa Culler: And here's my Jupiter lab. So we can head over to this

614
01:29:25.460 --> 01:29:31.200
Elsa Culler: link on the side or on like I was showing. You can go to the gift. Tab

615
01:29:31.480 --> 01:29:36.470
Elsa Culler: my mouse isn't great. It's not always clicking on things when I say

616
01:29:36.560 --> 01:29:40.030
Elsa Culler: and then we're gonna go ahead and clone

617
01:29:40.220 --> 01:29:45.880
Elsa Culler: repository. This link is in the chat, but I'm gonna

618
01:29:46.790 --> 01:29:48.789
Elsa Culler: copy it and put it down

619
01:29:48.840 --> 01:29:51.469
Elsa Culler: and the message here again.

620
01:29:52.630 --> 01:29:56.790
Elsa Culler: This is an https link. So

621
01:29:58.350 --> 01:30:00.930
Elsa Culler: you will not need

622
01:30:01.920 --> 01:30:06.809
Elsa Culler: to set up the authentication yet. Here.

623
01:30:09.310 --> 01:30:17.380
Elsa Culler: okay, so I'm copying and pasting this easel. AI Link and I can go ahead and clone.

624
01:30:18.740 --> 01:30:21.809
Elsa Culler: And now it shows up in the main folder.

625
01:30:28.220 --> 01:30:30.660

Elsa Culler: and we can see all of these

626
01:30:30.740 --> 01:30:33.540
Elsa Culler: notebooks, and then I believe.

627
01:30:33.810 --> 01:30:39.360
Elsa Culler: I was running this prepare environment.sh!

628
01:30:41.710 --> 01:30:42.760
Elsa Culler: So

629
01:30:45.050 --> 01:30:56.989
Elsa Culler: you'll go here into the terminal. You'll notice that. The path in my terminal is the same as the path here in the file, browser, and so

630
01:30:57.410 --> 01:31:01.020
Elsa Culler: And so I can go ahead and

631
01:31:01.910 --> 01:31:04.590
Elsa Culler: run this command.

632
01:31:05.680 --> 01:31:15.899
Elsa Culler: There's a couple of ways to run shell scripts. One is to use the source command. Sometimes dot slash, prepare environment.sh would work, too.

633
01:31:16.870 --> 01:31:23.039
Elsa Culler: But here, this is going. And that's gonna take a minute. So I are you? How are you? How are you doing?

634
01:31:23.180 --> 01:31:25.639
Ayoub Ghriss: Yeah, let's go. Let's go. Okay.

635
01:31:37.750 --> 01:31:40.190
Elsa Culler: perfect. Yeah, I can see that. Now.

636
01:31:45.500 --> 01:31:48.560
Ayoub Ghriss: K, okay, so let's get started.

637
01:31:50.840 --> 01:31:51.730
Ayoub Ghriss: Thank you.

638
01:31:56.180 --> 01:31:57.619
Ayoub Ghriss: So read it on that.

639
01:31:58.690 --> 01:32:06.699
Ayoub Ghriss: So, as I was saying, the term artificial intelligence is more general. artificial intelligence is about developing software

640
01:32:06.730 --> 01:32:13.009
Ayoub Ghriss: that is intelligent in a way that can reason learn through new complex stacks.

641
01:32:13.200 --> 01:32:17.290
Ayoub Ghriss: the term intelligence is more like a philosophical thing.

642
01:32:17.810 --> 01:32:24.769
Ayoub Ghriss: But there are 2 basic approaches to define intelligence. First, one is human means that

643
01:32:24.880 --> 01:32:28.009
Ayoub Ghriss: software error machine is intelligent. If it can

644
01:32:28.310 --> 01:32:35.599
Ayoub Ghriss: make decisions similar to comparable to a human. You might be familiar familiar with the Turing test.

645
01:32:36.010 --> 01:32:42.810
Ayoub Ghriss: and the second one is the rational approach, which means that the algorithm should make the most rational decision

646
01:32:43.440 --> 01:32:52.689
Ayoub Ghriss: and is one method of developing such such algorithms.

647
01:32:53.520 --> 01:33:00.700
Ayoub Ghriss: There are 3 methods in machine learning supervised,

unsupervised and reinforcement learning.

648
01:33:00.850 --> 01:33:06.439
Ayoub Ghriss: I'm just going to cover, supervised and unsupervised in the repo.

649
01:33:06.450 --> 01:33:19.810
Ayoub Ghriss: You're going to find a folder called Rl, so that one, if you only take it later to get familiar with the notion of reinforcement learning. But it's a bit too technical to cover in today's talk

650
01:33:22.540 --> 01:33:23.220
Ayoub Ghriss: here.

651
01:33:23.730 --> 01:33:31.360
Ayoub Ghriss: This is like a sketch of the schedule, but I'm not sure we can have to follow that already last like 15 min, so I'll say

652
01:33:34.000 --> 01:33:43.879
Ayoub Ghriss: so in talking about supervised. The supervision is coming from what we call the labels. So imagine we have data sets. So a data set is just a

653
01:33:45.180 --> 01:33:47.339
Ayoub Ghriss: some number of of Tuples

654
01:33:47.700 --> 01:33:56.049
Ayoub Ghriss: that we are going to denote by x one y. One until Xn. YN. So these are the the samples.

655
01:33:56.420 --> 01:34:05.420
Ayoub Ghriss: and x one here is going to be the features. Then Y here is going to be the labels. So the goal in reinforcement learning is.

656
01:34:05.720 --> 01:34:12.630
Ayoub Ghriss: we assumed it is a function that would map the features to the labels. We don't know what that function is.

657
01:34:12.710 --> 01:34:18.010

Ayoub Ghriss: We don't even know if it exists. Why? Because sometimes you can have

658
01:34:18.470 --> 01:34:20.480
Ayoub Ghriss: in the dataset, you might have

659
01:34:20.700 --> 01:34:36.729
Ayoub Ghriss: the same features, but with different labels, usually machine learning, we remove that assumption. So it's basically we make assumption that the data set is kind of consistent, so the same features cannot have the same table.

660
01:34:37.440 --> 01:34:50.979
Ayoub Ghriss: and the goal here is to approximate these or find the best approximation for for this function. and the approximation quality is evaluated, based on some loss function.

661
01:34:51.380 --> 01:34:58.070
Ayoub Ghriss: Okay? So the first choice that we do in supervised learning is that we have to choose

662
01:34:58.090 --> 01:35:02.420
Ayoub Ghriss: the set of functions that we are going to

663
01:35:02.930 --> 01:35:09.280
Ayoub Ghriss: find the best approximation among it. And the second thing is, we had to choose the loss function.

664
01:35:09.980 --> 01:35:15.659
Ayoub Ghriss: And this loss function is what gonna tell us whether the approximation is good or not.

665
01:35:17.300 --> 01:35:19.589
Ayoub Ghriss: And even when you define the same

666
01:35:19.790 --> 01:35:28.139
Ayoub Ghriss: classical functions and the same loss. Then you have so many algorithms that you can use. Each one of them have its own guarantees.

667

01:35:31.840 --> 01:35:35.300
Ayoub Ghriss: So one of the simplest tasks is, or the

668
01:35:35.520 --> 01:35:42.029
Ayoub Ghriss: common one is the classification. So in that case you're given a set of features in this case going to be the cat images.

669
01:35:42.050 --> 01:35:55.429
Ayoub Ghriss: and you are trying to classify the image. And the second one is, instead of just classifying, you're also looking at. You're trying to find the region. the minimal region that

670
01:35:56.240 --> 01:35:59.710
Ayoub Ghriss: that makes that label relevant to that image.

671
01:36:00.530 --> 01:36:18.130
Ayoub Ghriss: Then you have also object extra detection. In that case it's not just one label per per feature or per image. You have so many labels. and in that case the label will be the region of the image and the object identity in that region.

672
01:36:18.190 --> 01:36:23.600
Ayoub Ghriss: You also have the image segmentation. So

673
01:36:23.890 --> 01:36:46.029
Ayoub Ghriss: here in classification you have a discrete. You have a discrete label in localization. You have a discrete one, and the continuous one, which is the coordinates. and vice versa. So it's not. It's not like a binary dichotomy of of the nature of the data sets we're working with. You can have all types of labels in the data.

674
01:36:48.210 --> 01:37:00.299
Ayoub Ghriss: You also have natural language processing. And this is what we call the sequence to sequence. prediction. In that case you have a speech

675
01:37:01.200 --> 01:37:03.000
Ayoub Ghriss: which is

676
01:37:03.110 --> 01:37:17.249
Ayoub Ghriss: converted to other features. Usually this speech is very

long means that you have an audio file, and then you have certain thousands of frames per second. So what happens there is that you first do the feature extraction.

677
01:37:17.460 --> 01:37:30.589
Ayoub Ghriss: and then you use the some machine machine learning model on top to match it to the text. So in this case the label is the sentence that are being uttered in the, in the voice.

678
01:37:34.990 --> 01:37:40.160
Ayoub Ghriss: Instead of taking the speech as input, you can also take

679
01:37:41.020 --> 01:37:46.749
Ayoub Ghriss: you can also take the other sentences as input in this case, you're doing translation.

680
01:37:46.760 --> 01:37:51.190
Ayoub Ghriss: So here you're translating an English sentence to

681
01:37:51.800 --> 01:37:54.410
Ayoub Ghriss: reference like to French

682
01:38:01.120 --> 01:38:09.080
Ayoub Ghriss: in all these examples that I've given. There's a common theme to all of them, and that's why we call the bias variance trade-off.

683
01:38:09.280 --> 01:38:15.989
Ayoub Ghriss: So if we're given this like set of data here, and we're trying to find the best regression.

684
01:38:16.530 --> 01:38:20.040
Ayoub Ghriss: There's also there's always this problem whether to find

685
01:38:20.240 --> 01:38:27.080
Ayoub Ghriss: high varying, high bias which called underfitting. In this case, you're not learning the

686
01:38:27.160 --> 01:38:29.070
Ayoub Ghriss: the true structure of the data

687
01:38:29.970 --> 01:38:37.699
Ayoub Ghriss: optimal one is what we're looking for. But the overfitting is when you are using 2

688
01:38:37.840 --> 01:38:40.969
Ayoub Ghriss: too much complex model, that kind of

689
01:38:41.020 --> 01:38:58.149
Ayoub Ghriss: fit, the entire training data. But it does not capture the interesting structure of the data. So if you add, like another point around this region here, you're going to to miss it. So the one on the left here we call it. We say it has a high bias

690
01:38:58.390 --> 01:39:04.410
Ayoub Ghriss: and the one on the right. We say it has a high variance, and you can show analytically that

691
01:39:05.520 --> 01:39:14.630
Ayoub Ghriss: when you optimize the bias. the variance becomes worse and vice versa. So there's always this trade-off

692
01:39:18.250 --> 01:39:20.370
Ayoub Ghriss: in classification is the same thing.

693
01:39:20.410 --> 01:39:29.989
Ayoub Ghriss: except here, that you can see that the under the overfitting is about finding 2 like a complex boundary between the 2,

694
01:39:30.050 --> 01:39:31.940
Ayoub Ghriss: the 2 different classes.

695
01:39:40.540 --> 01:39:43.520
Ayoub Ghriss: perhaps the most simple example.

696
01:39:44.670 --> 01:39:47.890
Ayoub Ghriss: Someone. Second, let me just check the chat.

697
01:39:51.070 --> 01:40:00.840

Ayoub Ghriss: Okay? Perhaps the simple, most simple example. Here is Classifier called decision trees, where you you can have different type of features.

698
01:40:00.970 --> 01:40:02.550
Ayoub Ghriss: Okay? So

699
01:40:03.200 --> 01:40:11.659
Ayoub Ghriss: you can have continuous features. You can have discrete features in this case the continuous features are the age and the weight.

700
01:40:12.070 --> 01:40:25.380
Ayoub Ghriss: Actually, you can argue, they are discrete. But let's just assume they are continuous, and then the discrete one is whether the person is smoking or not, and the label would be, whether it's low where they have low or high risk.

701
01:40:26.350 --> 01:40:27.899
Ayoub Ghriss: and the same thing here.

702
01:40:27.940 --> 01:40:30.219
Ayoub Ghriss: You might have

703
01:40:31.390 --> 01:40:34.059
Ayoub Ghriss: certain threshold at which

704
01:40:34.320 --> 01:40:39.669
Ayoub Ghriss: the model can over fit in a way that if the tree goes too deep.

705
01:40:39.900 --> 01:40:44.750
Ayoub Ghriss: You might have like a kind of a binary tree that will

706
01:40:44.870 --> 01:40:51.810
Ayoub Ghriss: end up with one person at each final node. but it does not mean that the model is doing is doing well.

707
01:40:54.930 --> 01:40:59.099
Ayoub Ghriss: The second one is perceptron, so the name might be kind

of

708
01:40:59.420 --> 01:41:15.509
Ayoub Ghriss: scary. But it's not so what happens here is just you take the features and you multiply them by a matrix. So you just take the features multiplied by certain weights. In this case we have 4 features for each feature we have the corresponding weight, and then we add the bias.

709
01:41:16.070 --> 01:41:24.330
Ayoub Ghriss: and in the perceptron we do in a classification. So if this value here is positive, we say it's one.

710
01:41:24.590 --> 01:41:30.420
Ayoub Ghriss: In this case, let's say it's high risk and 0. It means it's low risk.

711
01:41:40.470 --> 01:41:41.580
Ayoub Ghriss: Sorry I

712
01:41:41.930 --> 01:41:44.309
Ayoub Ghriss: and jump too many things.

713
01:41:44.940 --> 01:41:46.200
Ayoub Ghriss: we were.

714
01:41:46.850 --> 01:41:48.529
Ayoub Ghriss: We're here. Okay.

715
01:41:49.280 --> 01:41:49.940
Ayoub Ghriss: sure.

716
01:41:50.570 --> 01:41:58.210
I'm just gonna introduce the data set that what we're going to do work with. It's gonna be a simple image classification of cats and dogs.

717
01:41:58.570 --> 01:42:02.500
Ayoub Ghriss: And there is a plot twist that we're going to see later.

718

01:42:03.780 --> 01:42:19.140
Ayoub Ghriss: But the question here, how do we deal with images? I chose this. I chosen this type of data because I soon you are more familiar with spatial data. Just give you an example of how to work with images.

719
01:42:20.480 --> 01:42:31.660
Ayoub Ghriss: So with images, we have this concept of what we call convolutions. So it's kind of an extension of the matrix multiplication to spatial spatial data.

720
01:42:31.770 --> 01:42:41.590
Ayoub Ghriss: So here you can assume that these are the pixel values of the image that we are going to work with. And here what we call the kernel, or you can call it filter, depending on

721
01:42:42.820 --> 01:42:50.479
Ayoub Ghriss: the nonsense you want to follow, or the Python library that we only use. So in this case we have already initialized the weights

722
01:42:51.600 --> 01:42:54.990
Ayoub Ghriss: in the way we compute the convolution

723
01:42:55.880 --> 01:42:59.710
Ayoub Ghriss: is, we start with the upper left corner.

724
01:43:00.460 --> 01:43:07.979
Ayoub Ghriss: and we do an element-wise multiplication of the of these. So initially, we start from all the top left.

725
01:43:08.340 --> 01:43:11.479
Ayoub Ghriss: But assume that a certain level we reach this

726
01:43:11.630 --> 01:43:14.360
Ayoub Ghriss: region here where we're going to do the multiplication.

727
01:43:14.390 --> 01:43:21.560
Ayoub Ghriss: So the result of this convolution at this point here is just going to be 2 multiplied by one.

728
01:43:21.590 --> 01:43:26.789
Ayoub Ghriss: 2 multiplied by 5, it's going to be 10, and then we do the same thing for all of them. And we sum

729
01:43:28.330 --> 01:43:35.629
Ayoub Ghriss: so one thing here to note is that the input image and the output image have the same shape.

730
01:43:35.860 --> 01:43:40.669
Ayoub Ghriss: And we are good going to see why. So here, if you want to do the math.

731
01:43:42.540 --> 01:43:43.740
Ayoub Ghriss: just simple

732
01:43:44.440 --> 01:43:56.060
Ayoub Ghriss: multiplication addition. So in this case, the output of this convolutional operation between the input and the and the filter is going to is going to be 1 98.

733
01:43:56.390 --> 01:43:58.760
Ayoub Ghriss: You can check later. If the method is correct.

734
01:44:01.290 --> 01:44:04.760
Ayoub Ghriss: however, the output can also

735
01:44:04.960 --> 01:44:11.599
Ayoub Ghriss: have different shape than the input. So the difference here is that we have what we call a stripe.

736
01:44:11.690 --> 01:44:16.669
Ayoub Ghriss: So as right here is that it means that we start with the convolution at the top left.

737
01:44:17.460 --> 01:44:26.530
Ayoub Ghriss: But then this right here is 2. It means that we are going to jump 2 steps. So basically 2 columns and then compute the second convolution.

738

01:44:26.720 --> 01:44:27.490
Ayoub Ghriss: Okay?

739
01:44:27.610 --> 01:44:33.070
Ayoub Ghriss: And the same thing vertically and horizontally. So the the we have, the sorry.

740
01:44:34.970 --> 01:44:36.409
Ayoub Ghriss: We have the violets.

741
01:44:40.150 --> 01:44:52.799
Ayoub Ghriss: we have the violet convolution followed by the blue, so the third one is going to be the green, since we reach the the left, the right bound, and then we do the stride 2 steps to to the bottom.

742
01:44:53.720 --> 01:45:05.790
Ayoub Ghriss: and the same logic as before. So these are the element-wise multiplications. And then we just have to add everything here. So in this case it's going to be 2 here, it's going to be 5.

743
01:45:05.810 --> 01:45:09.450
Ayoub Ghriss: And yeah. So you see here that we

744
01:45:10.460 --> 01:45:17.819
Ayoub Ghriss: almost cut the the, we almost reduced the size of the input by half.

745
01:45:17.890 --> 01:45:25.340
Ayoub Ghriss: And the reason here, or the factor here, the stride is one previously the stride. Sorry this right here is 2.

746
01:45:25.590 --> 01:45:26.959
Ayoub Ghriss: Previously the

747
01:45:28.250 --> 01:45:36.350
Ayoub Ghriss: previously the stride was one. Okay, so that's why we keep the size. When you increase the stride, you reduce the size of the output of the output.

748
01:45:40.490 --> 01:45:54.500

Ayoub Ghriss: So now we imagine that we take the. we take the image, we apply the convolution, and then we have the output. And in this case, in classification with neural network. The neural network is going to.

749
01:45:56.330 --> 01:46:00.030
Ayoub Ghriss: Yeah, when you have a very large image you want to use.

750
01:46:00.140 --> 01:46:12.149
Ayoub Ghriss: stride larger than one to reduce the size, but sometimes, if you use a stride, one, you you take in 2 min, 2 min to small steps, and you might be learning redundant information.

751
01:46:12.590 --> 01:46:20.509
Ayoub Ghriss: So imagine I have a high resolution landscape picture stride. One is just one pixel. You're not actually moving anywhere. So

752
01:46:21.100 --> 01:46:25.249
Ayoub Ghriss: it's more like a parameter that you want to tune, to find

753
01:46:25.810 --> 01:46:27.240
Ayoub Ghriss: at each steps.

754
01:46:27.750 --> 01:46:35.639
Ayoub Ghriss: If you jump from one region to another, using a stride, you're going to have more different features than the previous the previous region.

755
01:46:35.910 --> 01:46:39.230
Ayoub Ghriss: Okay, going back to modeling the problem.

756
01:46:39.350 --> 01:46:55.439
Ayoub Ghriss: The output of the neural network in this case is gonna be probabilities. It's gonna be outputting the probability that the image is at cat, and the probability of the image of being a dog. So here imagine that the label 0, if it's a cat and and the labels one, if it's a dog

757
01:46:55.460 --> 01:46:58.230
Ayoub Ghriss: and the loss function. What we call the cross entropy

758
01:46:58.660 --> 01:47:03.760
Ayoub Ghriss: is this is the more general formula. But you can just easily

759
01:47:03.810 --> 01:47:15.310
Ayoub Ghriss: just write it down more simple. So if the label is 0, then the loss is minus log ecat, and if it's one, then the loss is minus log E dog.

760
01:47:15.460 --> 01:47:25.960
Ayoub Ghriss: And we are trying to minimize the loss. Okay. so it means here. Since log is an increasing function, we're trying to maximize the probability.

761
01:47:26.410 --> 01:47:34.869
Ayoub Ghriss: So it's as simple as that. If the label, if they labels cat, we're trying to maximize the pcat. If the label is dark. We're trying to maximize. Pdf.

762
01:47:39.130 --> 01:48:01.289
Ayoub Ghriss: the question here is how to transform the output of a neural network to probabilities. And that's where we have the notion of activation functions. So after each time you apply the convolution, you apply the activation function. So in this case, you want to apply an activation function that will transform whatever the output is to the range of 0 1.

763
01:48:01.940 --> 01:48:19.369
Ayoub Ghriss: The most relevant one is sigmoid. Okay? So the plot here is, it's always between 0 one. It never crosses these these limits, but you also have it, have other types of activation functions that might be relevant to whatever problem you have.

764
01:48:19.850 --> 01:48:21.870
Ayoub Ghriss: Usually we

765
01:48:22.060 --> 01:48:33.150
Ayoub Ghriss: always add an activation function after each convolution or after each multiplication layer, because these functions add non-linearity to the model.

766
01:48:33.650 --> 01:48:49.340
Ayoub Ghriss: These are basically the most famous one. So the tangent hip. But hyperbolic is just going to be mapping whatever you give it to, minus one and one. The relu is going to map it to positive range. So it means that at the output of a certain

767
01:48:49.390 --> 01:48:55.439
Ayoub Ghriss: element over the convolution is minus one or minus 2, then just going to be zeroed out.

768
01:48:57.010 --> 01:49:01.160
Ayoub Ghriss: the real can have a problem here. It means that if you 0 out everything.

769
01:49:06.240 --> 01:49:07.649
Ayoub Ghriss: yeah, we're we're gonna see that.

770
01:49:09.320 --> 01:49:23.000
Ayoub Ghriss: So we said that the activations help by adding non-linearity to the model. There's a second way of adding the non-linearity, and then what we call the pooling. So the pooling here. It means I'm applying

771
01:49:23.050 --> 01:49:32.240
Ayoub Ghriss: some nonlinear operations. So in this case we call it Max Bulling means that in this case it's a 2 by 2 Max bullying. So I'm taking each

772
01:49:32.490 --> 01:49:37.240
Ayoub Ghriss: 2 by 2 region, and try to find the Max. So the Max here is 20,

773
01:49:37.290 --> 01:49:39.080
Ayoub Ghriss: the Max here is 30,

774
01:49:39.120 --> 01:49:42.719
Ayoub Ghriss: and vice versa. So Max pooling is interesting because

775
01:49:42.740 --> 01:49:53.480
Ayoub Ghriss: there's no, there's no linear. There's no convolution

that is equivalent to Max pulling. It means that there is no filter that will allow you to get Max pulling

776
01:49:53.840 --> 01:50:12.070
Ayoub Ghriss: average pulling is easy. So in this case you just take a filter that is one quarter everywhere. So basically just summing everything and divide in by 4. But for the Max pooling you can prove that there is no convolution that will give you the Max pulling. That's why Max pooling is more is more popular.

777
01:50:16.560 --> 01:50:26.249
Ayoub Ghriss: So in neural network, the question there is, okay. So how do we train the neural network. And we have seen that the convolution there is the parameter.

778
01:50:26.280 --> 01:50:34.280
Ayoub Ghriss: And as in, for example, regression is going to be the weight matrix. So in the neural network. We are using gradient descent.

779
01:50:34.520 --> 01:50:41.870
Ayoub Ghriss: This is a very easy and nice function. but the

780
01:50:42.150 --> 01:51:02.579
Ayoub Ghriss: the neural networks are not really this nice. So you just use some batches to estimate some local gradient, and then you just follow the gradient where wherever you're going. It's not convex. It means that you're never guaranteed to reach the global optimum. So it's always some local optimum that that you are reaching.

781
01:51:07.070 --> 01:51:13.990
Ayoub Ghriss: Okay? So we're going to use. Now move to the Github. Sorry the notebooks.

782
01:51:18.840 --> 01:51:22.200
Ayoub Ghriss: So I'm going to move to the environment.

783
01:51:23.910 --> 01:51:31.829
Ayoub Ghriss: Assume at this level you have everything here. and that you have already executed the script and install the packages.

784

01:51:34.600 --> 01:51:37.169
Ayoub Ghriss: Yes or no. Any answers in chat.

785
01:51:41.050 --> 01:51:43.080
Ayoub Ghriss: Yes. yeah.

786
01:51:47.120 --> 01:51:49.079
Ayoub Ghriss: So we started with the supervised path.

787
01:51:50.090 --> 01:51:53.729
Ayoub Ghriss: Just hope you're gonna have enough time to do some interesting things. Okay?

788
01:51:57.890 --> 01:52:11.769
Ayoub Ghriss: Numpy is a package in Python, where you can do all the linear algebra thing, matrix multiplication, even random probabilities, like generating random variables. Multiplot by plots is doing some plots.

789
01:52:13.410 --> 01:52:19.230
Ayoub Ghriss: Yep. And then. yeah. the source. And then, if you have some

790
01:52:19.480 --> 01:52:22.880
Ayoub Ghriss: question, is just type. Yes, to install the packages. Okay.

791
01:52:23.800 --> 01:52:25.520
Ayoub Ghriss: scikit-learn is

792
01:52:25.810 --> 01:52:36.470
Ayoub Ghriss: a package that contains most, if not all, the classical machine learning algorithms. You can also do deep learning. But it's not really the speciality of psychic learn. Okay.

793
01:52:38.050 --> 01:52:43.350
Ayoub Ghriss: so going to use it here for some utilities and to do some model selection.

794
01:52:44.430 --> 01:52:46.319

Ayoub Ghriss: So I'm going ahead and start.

795
01:52:51.440 --> 01:52:58.590
Ayoub Ghriss: okay. So I forgot to. One thing, which is I have to do. Source download data

796
01:53:00.290 --> 01:53:02.610
Ayoub Ghriss: should be really fast. So that's not

797
01:53:08.630 --> 01:53:13.870
Ayoub Ghriss: so you do that. You're going to have a data set here that has the data that we're all going to use.

798
01:53:14.620 --> 01:53:17.400
Ayoub Ghriss: So let me go ahead. Oops.

799
01:53:17.710 --> 01:53:22.260
Ayoub Ghriss: no, it is. It is there.

800
01:53:22.480 --> 01:53:23.530
Ayoub Ghriss: So this is.

801
01:53:23.820 --> 01:53:25.400
Ayoub Ghriss: yeah, it is there.

802
01:53:26.880 --> 01:53:31.530
Ayoub Ghriss: Okay, let me restart the notebook, I guess. Start the kernel.

803
01:53:39.490 --> 01:53:40.250
Ayoub Ghriss: Okay?

804
01:53:43.650 --> 01:53:53.050
Ayoub Ghriss: So the images are 2, 56 by 2, 56 of some nice cats and dogs. Let me just decrease.

805
01:53:55.470 --> 01:54:01.879
Ayoub Ghriss: Okay. this is a random shuffle, so you might get different images depending

806
01:54:02.600 --> 01:54:04.140
Ayoub Ghriss: on the server.

807
01:54:07.340 --> 01:54:09.819
Ayoub Ghriss: It's quite a balanced data set.

808
01:54:09.910 --> 01:54:13.559
Ayoub Ghriss: So the problem of label balance is not a problem.

809
01:54:15.510 --> 01:54:23.639
Ayoub Ghriss: We're going to use keras tensorflow. One of the 2 most popular deep learning frameworks. Then the other one is pythorch.

810
01:54:24.800 --> 01:54:30.619
Ayoub Ghriss: One thing you can do in when you have images is what we call augmentation.

811
01:54:30.810 --> 01:54:50.580
Ayoub Ghriss: So we know that if we flip the image or rotate it or do some random rotation, it's still going to be a dog still gonna be at the cat. So we can exploit these augmentations to kind of superficially, artificially, increase the size of the data of the data sets

812
01:54:50.830 --> 01:54:54.979
Ayoub Ghriss: right here. It's the same image. But we can generate up to 9

813
01:54:55.070 --> 01:54:57.460
Ayoub Ghriss: variations of of the image.

814
01:54:58.350 --> 01:55:00.479
Ayoub Ghriss: Why we do this.

815
01:55:01.120 --> 01:55:07.950
Ayoub Ghriss: it's basically why? A way to avoid the overfitting problem that we have seen before. Okay.

816

01:55:08.410 --> 01:55:23.880
Ayoub Ghriss: so in this case, you're saying, I'm not just looking at a certain point you can. This is very similar to the 2D plot that we have. It's like moving the the dots slightly to the left or slightly to the right, to force a model to learn something that is not very tight

817
01:55:29.910 --> 01:55:30.580
Ayoub Ghriss: it.

818
01:55:31.440 --> 01:55:34.270
Ayoub Ghriss: So what happening here is that

819
01:55:34.400 --> 01:55:40.280
Ayoub Ghriss: I'm just creating what we call an input layer in keras or an input

820
01:55:40.290 --> 01:55:46.360
Ayoub Ghriss: and I'm trying to augment the data. So exactly, what do I have what I have done here?

821
01:55:47.650 --> 01:55:50.340
Ayoub Ghriss: And then I'm doing the rescaling, the rescaling.

822
01:55:50.750 --> 01:55:57.779
Ayoub Ghriss: So the images are RGB pixels. So the values are going to be between 0 and 2 55.

823
01:55:58.210 --> 01:56:00.100
Ayoub Ghriss: So divide by 2, 55

824
01:56:00.350 --> 01:56:03.379
Ayoub Ghriss: to make sure that it's scaled down to 0 1

825
01:56:05.250 --> 01:56:06.469
Ayoub Ghriss: why do we do that?

826
01:56:07.150 --> 01:56:13.450
Ayoub Ghriss: Because if the pixel values are too high, then the model is going to be very sensitive to

827
01:56:13.460 --> 01:56:15.830
Ayoub Ghriss: to the input.

828
01:56:16.330 --> 01:56:17.410
Ayoub Ghriss: Okay.

829
01:56:18.120 --> 01:56:33.759
Ayoub Ghriss: so this is where you're gonna start a plane with the with the sequential model happening here. So I'm telling the I'm creating sequential model. Which means that it's stack of different layers. The input layer. It tells it tells the

830
01:56:33.930 --> 01:56:37.410
Ayoub Ghriss: It tells the chorus that we expect an image.

831
01:56:40.750 --> 01:56:54.799
Ayoub Ghriss: So the input layer here is telling the chaos here, we expect an input, that is 2, 56 by 2, 56. Then we doing their scaling. And then we add in some augmentation to artificially increase the size of of the data.

832
01:57:01.920 --> 01:57:06.680
Ayoub Ghriss: Okay. this is where we start. So

833
01:57:06.990 --> 01:57:11.779
Ayoub Ghriss: this basically, the same type of convolutions that we have that we have shown

834
01:57:11.870 --> 01:57:19.190
Ayoub Ghriss: 32. It means that I'm using 32 convolutions. Okay? So the input has 3

835
01:57:19.470 --> 01:57:20.690
Ayoub Ghriss: layers.

836
01:57:20.710 --> 01:57:30.740
Ayoub Ghriss: And here I'm saying that I'm going to have 32 out in the output. I'm going to have 32 layers in in the input

837
01:57:31.850 --> 01:57:38.890
Ayoub Ghriss: so what happens here is that instead of just doing the convolution on the red

838
01:57:38.900 --> 01:57:47.389
Ayoub Ghriss: channel of the image, I'm also doing it on the green channel. I'm also doing it on the blue channel. And I'm summing all of those. Okay.

839
01:57:47.480 --> 01:57:55.829
Ayoub Ghriss: So when I'm sum, when I sum all of those that's basically one filter. And I do the same thing 32 times.

840
01:57:56.260 --> 01:58:01.750
Ayoub Ghriss: And this way I move from 3 channels in the input to 32 channels.

841
01:58:02.450 --> 01:58:07.239
Ayoub Ghriss: 8. Here is the size of the filter. In the last example we have used. My

842
01:58:09.230 --> 01:58:22.289
Ayoub Ghriss: So in example, I showed there, we just use the filter. This is 4 by 4. In this case it's 8 by 8. And the stride is 4. Which means that each time we're going to jump 4 pixels.

843
01:58:25.040 --> 01:58:31.029
Ayoub Ghriss: Okay? So the input of this layer is going to be 63, 63 by 32.

844
01:58:31.580 --> 01:58:32.380
Ayoub Ghriss: Okay.

845
01:58:32.930 --> 01:58:45.760
Ayoub Ghriss: And then we flatten these. It means that we take in the image that is 2 dimensional. And we're just creating one, vector, that has all the pixels, just one dimensional. Vector so it's going to be

846
01:58:45.790 --> 01:58:52.520
Ayoub Ghriss: as 63 by 63 by 32. Okay, that is what the flatten is

doing. And then

847
01:58:52.530 --> 01:58:56.399
Ayoub Ghriss: here we're doing a linear multiplication. So when it's multiplying

848
01:58:56.940 --> 01:59:03.450
Ayoub Ghriss: vector that has the product of these numbers as the dimension. And then we're multiplying by by the weights

849
01:59:03.680 --> 01:59:22.189
Ayoub Ghriss: and the end we're doing. Softmax softmax is a variation of the sigmoid. But when we are doing binary classification, they are pretty much similar. The same goal is to scale the output of the layer to a 0 one region, and where the probabilities. Sum to one.

850
01:59:23.070 --> 01:59:23.900
Ayoub Ghriss: Okay.

851
01:59:24.760 --> 01:59:33.580
Ayoub Ghriss: so the task for you be to add more convolutions. Change the parameters. you can also

852
01:59:33.680 --> 01:59:42.119
Ayoub Ghriss: use Max pulling. So here, instead of showing the answer, you can also go to just Google Keras, Max, pulling

853
01:59:45.660 --> 01:59:49.069
Ayoub Ghriss: there. Come. So this is the maximum layer.

854
01:59:49.490 --> 01:59:57.770
Ayoub Ghriss: And, as we said, the default. One is 2 by 2. So if I want to add a Max pulling layer, I would just come here undo

855
01:59:57.990 --> 02:00:01.160
Ayoub Ghriss: Kira's layers. Max.

856
02:00:02.200 --> 02:00:03.270
Ayoub Ghriss: fooling

857
02:00:03.300 --> 02:00:07.949
Ayoub Ghriss: to DI can change instead of 2 by 2 can do like 4 by 4,

858
02:00:08.360 --> 02:00:10.730
Ayoub Ghriss: and everything is is quite the same.

859
02:00:12.680 --> 02:00:18.060
Ayoub Ghriss: So once I do that hmm latest.

860
02:00:18.370 --> 02:00:25.030
Ayoub Ghriss: once I do that I can do the summary thing, and it shows me the layers and the number of weights.

861
02:00:25.050 --> 02:00:32.619
Ayoub Ghriss: Since the augmentation I have no parameter here. It's going to just be 0 parameter. And the convolution is going to be the weights of the filter plus the bias.

862
02:00:32.640 --> 02:00:33.740
Ayoub Ghriss: And yeah.

863
02:00:34.340 --> 02:00:42.089
Ayoub Ghriss: so at the end. I have almost 1 million parameter in my neural network. And that's I'm going to use.

864
02:00:42.300 --> 02:00:46.190
Ayoub Ghriss: How do you usually choose the number? That's the usually you just do

865
02:00:46.620 --> 02:00:49.710
Ayoub Ghriss: what we are going to do now, which is

866
02:00:50.450 --> 02:00:52.380
Ayoub Ghriss: we're going to split the data.

867
02:00:53.870 --> 02:01:07.059
Ayoub Ghriss: We already done this. So we in the model selection, module on socket learn, we have the training data. And I'm going to split it to a training data and validation data. Okay, so basically,

you're holding

868
02:01:07.170 --> 02:01:09.969
Ayoub Ghriss: part of the data to the site.

869
02:01:10.500 --> 02:01:13.040
Ayoub Ghriss: And what happens here is that we train in the model.

870
02:01:13.510 --> 02:01:24.040
Ayoub Ghriss: First, we do the compilation. This is the optimizer that's going to do the stochastic gradient descent. You can choose whatever learning rate you want. So that's the step size of the update

871
02:01:24.140 --> 02:01:29.009
Ayoub Ghriss: at the Cross entropy loss. That's what I have defined before in

872
02:01:29.100 --> 02:01:30.590
Ayoub Ghriss: in

873
02:01:31.490 --> 02:01:38.280
Ayoub Ghriss: in the slides, and the metric that we are going to monitor is going to be the accuracy. Of course, we have the loss

874
02:01:38.330 --> 02:01:41.499
Ayoub Ghriss: cross entropy, loss by default. And then we're just looking at the accuracy.

875
02:01:42.990 --> 02:01:43.820
Ayoub Ghriss: Okay?

876
02:01:44.350 --> 02:01:58.480
Ayoub Ghriss: So in the training, you specify the training data, the number of the batch sizes. So the small chunks of the data you're going to use at each time, because sometimes the data set is pretty large. In this case it's not just have 1,000

877
02:01:58.630 --> 02:02:14.609
Ayoub Ghriss: samples. It's not too large. But if the data set is too large, you have, like 10,000. 20,000. 30,000. You can do the updates

all at once. It's too expensive computationally. So we do. The stochastic one means that we're taking 16 samples at a time doing the optimization

878
02:02:15.580 --> 02:02:32.890
Ayoub Ghriss: and the validation data is what we are going to monitor. So the validation data is not used for the training. This is more used to just see, how the model does on the data that it hasn't seen. And the epox is how many times I'm going through the entire training data. Okay.

879
02:02:36.910 --> 02:02:40.490
Ayoub Ghriss: okay, I'm not gonna wait for all of these.

880
02:02:40.510 --> 02:02:44.299
Ayoub Ghriss: but you can try to toy with those. And let's see

881
02:02:46.020 --> 02:02:47.790
Ayoub Ghriss: who does the best.

882
02:02:50.670 --> 02:02:54.359
Ayoub Ghriss: Okay. So one thing here that a lot of people

883
02:02:55.160 --> 02:03:02.440
Ayoub Ghriss: like beginning in machine learning people beginning machine learning is that you just say, Okay, I'm going to train on training on the data.

884
02:03:02.640 --> 02:03:12.740
Ayoub Ghriss: And I'm going to monitor the validation data. And then I'm just going to choose the one that reaches the best validation accuracy. Okay.

885
02:03:13.000 --> 02:03:15.540
Ayoub Ghriss: But if you do that. then

886
02:03:16.550 --> 02:03:21.559
Ayoub Ghriss: implicitly, you are using the validation data to train your machine learning model.

887

02:03:21.640 --> 02:03:26.400
Ayoub Ghriss: It means that you are using the validation data to find the best

888
02:03:26.670 --> 02:03:40.250
Ayoub Ghriss: the best model. So it's kind of an implicit or indirect way of of training. So that's why, we always have another data set that's used called testing data or evaluation data.

889
02:03:40.270 --> 02:03:47.089
Ayoub Ghriss: In this case it has. It hasn't been used to to train the network or find the optimal weights.

890
02:03:47.140 --> 02:03:58.480
Ayoub Ghriss: And that's your real. The the real test that you'll be using. So in this case I can see that the valuation, value, valuation, accuracy, and the test accuracy are pretty close.

891
02:03:59.310 --> 02:04:04.610
Ayoub Ghriss: If I go to the test, I'm getting like 95, so it's slightly worse than

892
02:04:04.630 --> 02:04:06.310
Ayoub Ghriss: the

893
02:04:06.870 --> 02:04:11.010
Ayoub Ghriss: slightly worse than the one I observed during the the training.

894
02:04:14.610 --> 02:04:18.229
Ayoub Ghriss: This is just some function that I'm using to display some examples.

895
02:04:23.600 --> 02:04:25.439
Ayoub Ghriss: Okay? So we have.

896
02:04:25.910 --> 02:04:29.869
Ayoub Ghriss: we have an intrusive, we have an imposter cat here.

897
02:04:29.940 --> 02:04:36.700

Ayoub Ghriss: yeah, that's probably one of the ones that we have. We have missed.

898
02:04:40.830 --> 02:04:45.490
Ayoub Ghriss: Okay, we have quite few here. That's the doing is not good. So

899
02:04:45.720 --> 02:04:53.280
Ayoub Ghriss: why are we getting different accuracies and differentrons? Because we have the random augmentation at at the at the

900
02:04:53.630 --> 02:04:59.640
Ayoub Ghriss: in the stack of of the model. Umhm. I think there was a way you can do

901
02:05:00.850 --> 02:05:02.070
Ayoub Ghriss: model.

902
02:05:03.220 --> 02:05:05.780
Ayoub Ghriss: I think. Okay, let me just retry.

903
02:05:07.380 --> 02:05:12.249
Ayoub Ghriss: Oh, sorry. It's accuracy. It's one on the 10 samples. So I'm using the first.

904
02:05:12.290 --> 02:05:13.970
Ayoub Ghriss: So 10 samples.

905
02:05:14.540 --> 02:05:21.249
Ayoub Ghriss: I'm using the first 10 samples here. So that's why it's given these.

906
02:05:23.070 --> 02:05:24.330
Ayoub Ghriss: Okay.

907
02:05:25.750 --> 02:05:27.350
Ayoub Ghriss: since the dog.

908

02:05:28.490 --> 02:05:31.249
Ayoub Ghriss: if the property else cats.

909
02:05:32.060 --> 02:05:34.489
Ayoub Ghriss: Yeah. Good. Nothing.

910
02:05:34.970 --> 02:05:36.190
Ayoub Ghriss: Nothing weird.

911
02:05:38.910 --> 02:05:40.729
Ayoub Ghriss: Now, something is all right.

912
02:05:41.630 --> 02:05:43.110
Ayoub Ghriss: something.

913
02:05:45.490 --> 02:05:51.950
Ayoub Ghriss: Yeah. Just labeling everything cat, even though I have an accuracy one. So something is wrong in this code. Here.

914
02:05:54.450 --> 02:05:58.369
Ayoub Ghriss: data model hat. Let's see if someone can figure out.

915
02:06:04.190 --> 02:06:05.879
Ayoub Ghriss: Yep, it should be 0.

916
02:06:09.670 --> 02:06:11.339
Ayoub Ghriss: There you go.

917
02:06:11.640 --> 02:06:13.669
Ayoub Ghriss: Okay, so we're doing good on

918
02:06:14.150 --> 02:06:16.340
Ayoub Ghriss: on the 10 samples. Here.

919
02:06:17.900 --> 02:06:21.740
Ayoub Ghriss: let's choose. You can choose any random number of samples.

920

02:06:22.400 --> 02:06:27.679
Ayoub Ghriss: Oops just go from the 11 to the twentieth one.

921
02:06:28.690 --> 02:06:30.440
Ayoub Ghriss: Yeah. So here we're missing one.

922
02:06:36.550 --> 02:06:37.330
Yeah.

923
02:06:37.740 --> 02:06:45.889
Ayoub Ghriss: So this one isn't classified as it can't. So there's something wrong. Right? And on on the test data.

924
02:06:45.930 --> 02:06:55.739
Ayoub Ghriss: The question here is to try to find why the classification is wrong and in deep learning. That's a pretty difficult, difficult task.

925
02:06:56.010 --> 02:07:03.479
Ayoub Ghriss: because the neural network you can have multiple, redundant features, and you can also have some

926
02:07:03.640 --> 02:07:07.130
Ayoub Ghriss: jumps when you are looking at

927
02:07:07.190 --> 02:07:09.680
Ayoub Ghriss: the output of the of the networks

928
02:07:09.940 --> 02:07:26.969
Ayoub Ghriss: depending on the activation function. For example, if you use in Relu, you can have region where it's basically always cat. But then, if you move slightly to the left or to the right, you can have something that's called this

929
02:07:27.000 --> 02:07:33.529
Ayoub Ghriss: let's consider, dog. This is what we kind of known as the adversarial examples.

930
02:07:33.620 --> 02:07:43.089
Ayoub Ghriss: We're not covering here, but there are so many ways you

can trick the neural network to predict what you want without altering the input significantly.

931
02:07:44.360 --> 02:07:50.849
Ayoub Ghriss: The augmentation is one way to avoid this. So in this case we deal with the flipping or rotation, you can also add

932
02:07:51.190 --> 02:07:57.250
Ayoub Ghriss: and also add noise, random noise to the input to counter that

933
02:07:57.590 --> 02:07:58.400
Ayoub Ghriss: can.

934
02:07:58.540 --> 02:08:07.640
Ayoub Ghriss: Okay, perfect. Can. Can the background context be taken? That's a plot twist. We're going to come back to that

935
02:08:08.850 --> 02:08:09.650
Ayoub Ghriss: perfect

936
02:08:11.030 --> 02:08:16.690
Ayoub Ghriss: more, I'm going to move to the unsupervised learning part. I think I

937
02:08:16.750 --> 02:08:20.690
Ayoub Ghriss: gave kind of flavor of what supervised learning is.

938
02:08:22.660 --> 02:08:24.609
Ayoub Ghriss: how do you usually choose the number

939
02:08:25.710 --> 02:08:28.109
Ayoub Ghriss: 32? I already insert that

940
02:08:30.430 --> 02:08:35.050
Ayoub Ghriss: the scene you consider pixel colors or object shape to segment objects.

941
02:08:36.990 --> 02:08:48.170

Ayoub Ghriss: So so you can actually show in machine learning that the the first layers are more attentive to the color. But the more convolution you add

942
02:08:48.880 --> 02:08:57.640
Ayoub Ghriss: the more abstract the learning becomes okay. So one thing to see that you can see like filters

943
02:08:57.940 --> 02:08:59.000
Ayoub Ghriss: in

944
02:09:01.000 --> 02:09:02.500
Ayoub Ghriss: convolutions.

945
02:09:09.830 --> 02:09:10.580
Ayoub Ghriss: Yeah.

946
02:09:11.710 --> 02:09:19.669
Ayoub Ghriss: so this is basically the first come, I would say, this is the bottom part of the convolutions, so they are more like detecting the colors.

947
02:09:19.740 --> 02:09:22.020
Ayoub Ghriss: But the more the deeper you go.

948
02:09:22.260 --> 02:09:35.180
Ayoub Ghriss: You start looking at the edges as well, so it's not just the the color of the pixels. You're also looking at the edges, the different directions like an edge from the left to the right or the top to to the bottom.

949
02:09:35.920 --> 02:09:37.970
Ayoub Ghriss: Yeah. But

950
02:09:38.130 --> 02:09:45.639
Ayoub Ghriss: the concept here is that the deeper you go into the neural network, the more abstract the features that you are looking for of the input

951
02:09:51.280 --> 02:09:52.310

Ayoub Ghriss: A,

952
02:09:53.080 --> 02:09:55.180
Ayoub Ghriss: any questions to this part?

953
02:10:00.700 --> 02:10:05.549
Ayoub Ghriss: Have you already played with the code a little bit? No. I added, the

954
02:10:05.710 --> 02:10:08.959
Ayoub Ghriss: do convolutions. Yeah, you can add more.

955
02:10:09.950 --> 02:10:16.550
Ayoub Ghriss: You can add more layers. So in this case it can probably after the here I can just do like

956
02:10:16.700 --> 02:10:19.680
Ayoub Ghriss: I'll leave it 32. But I can do like 4

957
02:10:20.240 --> 02:10:21.180
Ayoub Ghriss: by

958
02:10:22.380 --> 02:10:23.869
Ayoub Ghriss: can do 4 by 2.

959
02:10:24.210 --> 02:10:39.350
Ayoub Ghriss: The trick here is that you use a large stride at the beginning. But as you go deeper you want to use smaller stripe because the images become smaller, and if you take a large jump you might be ignoring some really important aspects of the of the input

960
02:10:42.060 --> 02:10:47.640
Ayoub Ghriss: the convolutions are slower than matrix multiplication. So people try to

961
02:10:48.320 --> 02:10:53.570
Ayoub Ghriss: basically avoid very complex convolutions. And

962
02:10:54.180 --> 02:11:05.759

Ayoub Ghriss: the matrix multiplication is faster, but it takes more computational like, it takes a larger memory. On on the computer. So there's always this trade-off between slower

963
02:11:06.250 --> 02:11:10.440
Ayoub Ghriss: or less. Expensive on the memory site

964
02:11:11.660 --> 02:11:13.810
Ayoub Ghriss: and sample interest. Barely. Okay.

965
02:11:18.500 --> 02:11:21.690
Ayoub Ghriss: Yeah. Because, I added the Max full name here.

966
02:11:22.250 --> 02:11:28.439
Ayoub Ghriss: Yeah. So the trick there is that if you remove this, you're not going to get any good performance. So let's just do that.

967
02:11:28.510 --> 02:11:30.559
Ayoub Ghriss: Just remove the maximum link.

968
02:11:33.720 --> 02:11:38.670
Ayoub Ghriss: I'm seeing your slides and not your code. Oh,

969
02:11:44.300 --> 02:11:48.179
Ty Tuff, Ph.D.: There you go. Could you point people again to that thing you were talking about with the Max?

970
02:11:49.600 --> 02:11:51.790
Ayoub Ghriss: Yeah. So I said, Here, I already have.

971
02:11:52.820 --> 02:11:56.070
Ayoub Ghriss: Okay, let me go all right here.

972
02:12:09.780 --> 02:12:15.360
Ayoub Ghriss: So some people saying, I'm just using, they're just using the sample code and getting just 50 or something like that right?

973
02:12:18.060 --> 02:12:21.720

Ayoub Ghriss: That's correct. Yeah. So let me try here.

974
02:12:21.900 --> 02:12:22.670
Ayoub Ghriss: Yeah.

975
02:12:25.210 --> 02:12:30.179
Ayoub Ghriss: it's getting slower. So I guess a lot of people are training now. So there's some.

976
02:12:31.200 --> 02:12:34.540
Ayoub Ghriss: If you load on the server. Now, yeah. here.

977
02:12:36.810 --> 02:12:50.190
Ayoub Ghriss: So this is one other aspect of the deep learning network. The weights are initial, initialized randomly, and sometimes you get lucky by getting a good initialization that performs good like very well.

978
02:12:50.230 --> 02:12:55.670
Ayoub Ghriss: So usually when you are evaluating machine learning algorithm, you want to have multiple runs.

979
02:12:56.120 --> 02:12:58.890
Ayoub Ghriss: And then you average, because there's very.

980
02:12:58.980 --> 02:13:20.029
Ayoub Ghriss: it's not very highly likely. But the smaller and the neural network, the more effect of randomness you get. So it means that sometimes you can get a lucky shot. That's sometimes just initially, you get like 80% accuracy. In this case the model is not very that it's not that large compared to our machine learning models. So you might get you get started.

981
02:13:20.050 --> 02:13:21.869
Ayoub Ghriss: You may start in some

982
02:13:22.390 --> 02:13:30.870
Ayoub Ghriss: a good neighborhood. Yeah. So some people might get 90 from the beginning, others maybe later.

983

02:13:31.310 --> 02:13:34.180
Ayoub Ghriss: Hmm. okay. so

984
02:13:35.200 --> 02:13:38.589
Ayoub Ghriss: with this, I'm going to move to the unsupervised learning part.

985
02:13:43.320 --> 02:13:44.150
Ayoub Ghriss: Okay?

986
02:13:45.680 --> 02:13:52.380
Ayoub Ghriss: So the unsupervised learning. Part is not unsupervised in a way that you don't get the labels. Okay.

987
02:13:53.970 --> 02:13:59.110
Ayoub Ghriss: but we're still learning a mapping. and the mapping here

988
02:14:00.480 --> 02:14:04.179
Ayoub Ghriss: is not mapping the speakers to the labels.

989
02:14:04.630 --> 02:14:09.139
Ayoub Ghriss: But it's mapping to some code or some

990
02:14:10.400 --> 02:14:14.860
Ayoub Ghriss: lower dimensional representation debts

991
02:14:15.070 --> 02:14:17.779
Ayoub Ghriss: satisfies some constraints. Okay?

992
02:14:19.030 --> 02:14:23.800
Ayoub Ghriss: So in this case, if the code that we are trying to learn is

993
02:14:24.670 --> 02:14:26.879
Ayoub Ghriss: discrete, finite.

994
02:14:26.940 --> 02:14:31.029
Ayoub Ghriss: You can talk about clustering or sparse coding.

995
02:14:31.100 --> 02:14:36.740
Ayoub Ghriss: If it's continuous. You might be talking about dimensionality, reduction, or some generative models.

996
02:14:38.050 --> 02:14:43.469
Ayoub Ghriss: But in all these methods of unsupervised learning

997
02:14:43.800 --> 02:14:45.370
Ayoub Ghriss: the difference is

998
02:14:45.850 --> 02:14:56.339
Ayoub Ghriss: the properties of the space of, or the representation that you are that you? You are trying to learn. And the algorithm that you want to achieve that. Okay?

999
02:14:57.900 --> 02:15:09.430
Ayoub Ghriss: And I'm going to use a very specific example. That's it's not classical, classical in in a sense that it's not as old as the classmate algorithms or the sparse coding.

1000
02:15:09.540 --> 02:15:15.600
Ayoub Ghriss: But it's is very popular in in recent applications. And

1001
02:15:16.100 --> 02:15:19.790
Ayoub Ghriss: it's a way of using neural network to reduce the dimensions

1002
02:15:21.030 --> 02:15:31.040
Ayoub Ghriss: and at the same time have some generative aspect of of the data. So in this case, I'm going to move to the unsupervised learning notebook.

1003
02:15:37.990 --> 02:15:41.529
Ayoub Ghriss: It's very similar to I have seen before. When it comes to the code part

1004
02:15:43.530 --> 02:15:45.729
Ayoub Ghriss: I'm going to use.

1005

02:15:46.500 --> 02:15:53.850
Ayoub Ghriss: we call the Mnis data. So it's basically handwritten digits digits. I can do lot, image show

1006
02:15:54.140 --> 02:15:55.750
Ayoub Ghriss: or macho.

1007
02:15:58.960 --> 02:16:02.830
Ayoub Ghriss: I can take like the first example.

1008
02:16:05.010 --> 02:16:05.690
Ayoub Ghriss: yeah.

1009
02:16:06.150 --> 02:16:08.289
Ayoub Ghriss: this is 5. Trust me.

1010
02:16:12.880 --> 02:16:15.190
Ayoub Ghriss: Yeah. So then you have 0.

1011
02:16:15.260 --> 02:16:16.360
Ayoub Ghriss: And

1012
02:16:16.970 --> 02:16:25.640
Ayoub Ghriss: the goal here, these are 28 by 28. So that's like 7, 84 features. And

1013
02:16:25.960 --> 02:16:27.879
Ayoub Ghriss: the goal of the auto encoder

1014
02:16:28.450 --> 02:16:33.040
Ayoub Ghriss: is to compress the input into a lower dimensional

1015
02:16:33.260 --> 02:16:38.159
Ayoub Ghriss: output in a way that the lower dimensional output

1016
02:16:38.420 --> 02:16:45.649
Ayoub Ghriss: still contains enough information to build back the original input. Okay.

1017
02:16:46.129 --> 02:16:52.960
Ayoub Ghriss: so it's kind of, this compression. Okay? So we're using a neural network that will map

1018
02:16:53.090 --> 02:17:03.189
Ayoub Ghriss: 77, 84 pixels or features to like 4, 10. It's up to you to choose, and based on those 4 features that we have

1019
02:17:03.420 --> 02:17:10.300
Ayoub Ghriss: map to, we are using another neural network. So that's going to be the decoder

1020
02:17:10.620 --> 02:17:16.049
Ayoub Ghriss: that will try to build back the original input from that compressed

1021
02:17:16.260 --> 02:17:18.879
Ayoub Ghriss: from that compressed code.

1022
02:17:20.469 --> 02:17:25.429
Ayoub Ghriss: They don't have to be symmetric or similar in size.

1023
02:17:25.510 --> 02:17:40.700
Ayoub Ghriss: There's no constraints. The only constraint is that the output of the encoder should have the same dimensions as the input of the decoder and vice versa. So the output to decoder should be the same dimension as the input of the encoder.

1024
02:17:41.170 --> 02:17:44.209
Ayoub Ghriss: So you can add another arrow here. So it's kind of a cycle.

1025
02:17:47.410 --> 02:17:48.230
Ayoub Ghriss: Okay?

1026
02:17:49.430 --> 02:18:07.939
Ayoub Ghriss: So the same thing here, I'm creating a sequential model. The difference here is that I'm not using convolutions. And the way I do that is, I'm flattening and the image. So the image is 28 by 28. So I'm just creating one. Dimensional vectors is going to be 28 square.

That's 7, 84,

1027
02:18:08.629 --> 02:18:09.389
Ayoub Ghriss: okay?

1028
02:18:10.730 --> 02:18:13.480
Ayoub Ghriss: And these are stack of

1029
02:18:13.660 --> 02:18:25.829
Ayoub Ghriss: stack of matrix multiplications followed by a leak. Reloach is a special case. It's more like a general case of one of the relo activations and latent dimension.

1030
02:18:25.889 --> 02:18:30.750
Ayoub Ghriss: Latent dimension is the dimension of the code. And that's something that I choose.

1031
02:18:30.920 --> 02:18:31.610
Ayoub Ghriss: Okay.

1032
02:18:33.020 --> 02:18:35.120
Ayoub Ghriss: the decoder is much simpler.

1033
02:18:35.549 --> 02:18:37.650
Ayoub Ghriss: Just to

1034
02:18:37.660 --> 02:18:47.959
Ayoub Ghriss: 3 multi matrix multiplication. One thing to notice here is I'm using activation. Anyone knows why I'm using the sigmoid activation for the decoder.

1035
02:18:56.790 --> 02:19:01.770
Ayoub Ghriss: So I'm going back. This is where I'm loading the images. The images are 20, like

1036
02:19:02.469 --> 02:19:08.980
Ayoub Ghriss: just grayscale. In this case there's no RGB, and the values are between 0 to 2255.

1037

02:19:09.840 --> 02:19:14.760
Ayoub Ghriss: So what happened here is that I'm scaling the pixel values to the region the

1038
02:19:15.590 --> 02:19:17.490
Ayoub Ghriss: interval 0 one. Okay.

1039
02:19:17.799 --> 02:19:21.479
Ayoub Ghriss: so what happens here with the sigmoid in the decoder? I'm saying that

1040
02:19:21.549 --> 02:19:32.959
Ayoub Ghriss: the decoded code should also be in 0 one and the sigmoid activation I showed before it does exactly that. So no matter what the input is the output is always going to be between 0 and one.

1041
02:19:36.190 --> 02:19:38.330
Ayoub Ghriss: So here we're already kind of

1042
02:19:38.770 --> 02:19:42.279
Ayoub Ghriss: helping the model. Learn something useful.

1043
02:19:44.940 --> 02:19:53.670
Ayoub Ghriss: And in this case I'm using the latent dimension of 2. It means that I'm mapping all the pixels to 2 dimensions.

1044
02:19:54.380 --> 02:19:59.610
Ayoub Ghriss: The optimizer is Adam. When I do add them with the string instead of

1045
02:19:59.620 --> 02:20:14.000
Ayoub Ghriss: the item I used here. It means I'm using the default parameters. So in this case the default parameter is going to be 10 to minus 3 for the learning rate. So usually, when you do this, it means that you are just using the default. Parameters for the optimizer.

1046
02:20:14.170 --> 02:20:19.069
Ayoub Ghriss: On the other hand, the loss here is no longer the cross entropy. It's just the mean squared error.

1047

02:20:19.550 --> 02:20:28.180
Ayoub Ghriss: So I'm treating the image just as any vector in some Euclidean space. And I'm computing the distance

1048
02:20:28.680 --> 02:20:31.719
Ayoub Ghriss: it's not really a good distance, right?

1049
02:20:31.730 --> 02:20:41.790
Ayoub Ghriss: Because if I have a digit one and I shift it. Let's try. Probably find some to examples.

1050
02:20:44.840 --> 02:20:46.230
Ayoub Ghriss: Okay, let me see.

1051
02:20:48.850 --> 02:20:52.379
Ayoub Ghriss: Yeah. So the Euclidean distance. You're just

1052
02:20:52.660 --> 02:21:04.799
Ayoub Ghriss: comparing the images pixel-wise. So when the pixel is 0. The distance is 0 when the pixel is one, the distance 0 and vice versa. So imagine here that the one we got here matches

1053
02:21:05.620 --> 02:21:06.450
Ayoub Ghriss: the

1054
02:21:07.170 --> 02:21:17.889
Ayoub Ghriss: left bottom edge of the 0 character. Then you might actually get one cloak that is close to 0. Then, having, for example, a 0 that is all the way to the bottom left.

1055
02:21:18.220 --> 02:21:35.489
Ayoub Ghriss: So the pixel, wise comparisons, using the mean squared error is not a really good distance for images, because you just compare in pixel wise the the images pixel-wise. So if you have one digit, that is a shifted version of the other, you might have get a maximal distance, even though it's the same digit.

1056
02:21:35.610 --> 02:21:47.179
Ayoub Ghriss: But this case it does fairly fairly well. So I'm going to do the same thing. We have no labels. It means that I'm just taking the input. And I want to make sure that the decoded

1057
02:21:47.300 --> 02:21:53.999
Ayoub Ghriss: input is very close to the original data oops. I haven't defined it.

1058
02:21:55.930 --> 02:21:56.780
Ayoub Ghriss: Okay.

1059
02:21:56.900 --> 02:22:00.100
Ayoub Ghriss: it's faster. I guess people are not trained enough.

1060
02:22:08.790 --> 02:22:22.189
Ayoub Ghriss: These are examples I've chosen. They are not random, because I know the order of the images in the original data. So I just chosen them to make sure that I get different digits it. I'm not just showing vibes everywhere.

1061
02:22:23.960 --> 02:22:36.430
Ayoub Ghriss: So this is the original input. And this is the reconstructed one. So it's not doing fairly well. It's not doing that well on this example. On the one digit is constructing it very close.

1062
02:22:36.800 --> 02:22:50.019
Ayoub Ghriss: The 8 3 is not bad, but the 8 is not. It's not like that good but let me see how well I do here. Maybe I get some good shot with the initialization this time.

1063
02:22:51.080 --> 02:22:51.950
Ayoub Ghriss: Day.

1064
02:22:53.160 --> 02:22:56.380
Ayoub Ghriss: Yeah, slightly better, but not too too different.

1065
02:22:56.710 --> 02:23:05.249
Ayoub Ghriss: And this is the aspect that I talked about where I call the generative aspect. So don't forget here that I'm mapping

1066
02:23:05.370 --> 02:23:06.690
Ayoub Ghriss: the input

1067
02:23:07.130 --> 02:23:11.409
Ayoub Ghriss: I'm mapping my original data to Latin data and dimension.

1068
02:23:11.530 --> 02:23:17.650
Ayoub Ghriss: Okay. And I have chosen my latent dimension to be 2. Now.

1069
02:23:18.620 --> 02:23:32.630
Ayoub Ghriss: since I'm just in a 2 dimensional space here. So the coding space, just 2 dimensional space. I can choose some random samples and decode them. And hopefully, I get some new digits that I haven't seen before.

1070
02:23:33.200 --> 02:23:35.589
Ayoub Ghriss: This is basically the trick. So

1071
02:23:35.950 --> 02:23:46.900
Ayoub Ghriss: imagine. Here I map all my data to the space here, and then I choose some data, some point that's very close to the threes, but not exactly any one of them.

1072
02:23:47.490 --> 02:23:50.980
Ayoub Ghriss: and then I decode it, and I get a new

1073
02:23:51.030 --> 02:23:53.369
Ayoub Ghriss: digits that I haven't seen before.

1074
02:23:53.620 --> 02:23:58.469
Ayoub Ghriss: and this is the generative aspect of the autoencoders.

1075
02:23:58.860 --> 02:24:07.460
Ayoub Ghriss: So in this case I have chosen the coordinates minus 3, 2, and I want to decode it and see what get.

1076
02:24:08.220 --> 02:24:11.990
Ayoub Ghriss: Yeah, I get almost 3. Let me try minus one.

1077
02:24:13.180 --> 02:24:14.769

Ayoub Ghriss: Yeah, very close to 5.

1078
02:24:15.220 --> 02:24:18.099
Ayoub Ghriss: but very, not not exactly.

1079
02:24:19.630 --> 02:24:23.269
Ayoub Ghriss: And if you have seen, like all these

1080
02:24:23.800 --> 02:24:40.929
Ayoub Ghriss: AI generated images, the concept is kind of the same, it just. They do it in a more complex way to make sure that you reconstruct something that has high quality and that makes sense, not something that has like a hundred fingers. But the idea here is

1081
02:24:41.340 --> 02:24:45.829
Ayoub Ghriss: you still have. You're still using some large data set that you are

1082
02:24:46.200 --> 02:24:56.509
Ayoub Ghriss: encoding in some space. In that case it would be a very high dimensional one. It's not getting. V. 2 is not going to be, for probably around 2 or 3,000 dimensions.

1083
02:24:57.140 --> 02:24:58.060
Ayoub Ghriss: and

1084
02:24:58.470 --> 02:25:01.530
Ayoub Ghriss: once you figure out how to

1085
02:25:02.470 --> 02:25:12.290
Ayoub Ghriss: get just choose new points in that encoding space, you can decode them, and then you can get new images that you haven't seen before.

1086
02:25:13.370 --> 02:25:18.370
Ayoub Ghriss: In this case it's very simple. I'm just trying to guess, minus one minus 2

1087
02:25:18.390 --> 02:25:22.340
Ayoub Ghriss: practically assume that the encoding is following some

distribution.

1088
02:25:22.520 --> 02:25:35.489
Ayoub Ghriss: and if you do that, then you you are not just choosing randomly. I mean, you're just choosing arbitrarily like samples. You're saying, Okay, I'm assuming the encoding is some Gaussian distribution.

1089
02:25:35.820 --> 02:25:42.199
Ayoub Ghriss: and if I want to generate new samples, I'm just going to follow. I'm just going to take new samples from that Gaussian distribution.

1090
02:25:45.420 --> 02:25:52.319
Ayoub Ghriss: Alright. Now, you can also do some more tricks, but I will leave that this part

1091
02:25:52.330 --> 02:25:53.850
Ayoub Ghriss: for you to work with.

1092
02:25:53.890 --> 02:25:59.490
Ayoub Ghriss: But I'm just going to show you an example. Let me change the dimension from 2 to like 10.

1093
02:26:00.780 --> 02:26:01.610
Ayoub Ghriss: Okay.

1094
02:26:02.510 --> 02:26:05.079
Ayoub Ghriss: we're just making a mulch

1095
02:26:05.870 --> 02:26:07.180
Ayoub Ghriss: simpler model.

1096
02:26:12.190 --> 02:26:18.520
Ayoub Ghriss: So why? Why this unsupervised learning is important in machine learning? Because

1097
02:26:18.820 --> 02:26:21.860
Ayoub Ghriss: it's easy to do. You don't require labels.

1098
02:26:22.030 --> 02:26:29.279
Ayoub Ghriss: because sometimes labels are quite expensive to get like. Imagine some language that's rarely spoken in the world.

1099
02:26:29.400 --> 02:26:32.720
Ayoub Ghriss: and you want to do some supervised learning on it.

1100
02:26:33.390 --> 02:26:36.729
Ayoub Ghriss: If you have speech that has high dimension.

1101
02:26:37.050 --> 02:26:40.720
Ayoub Ghriss: it's expensive to train a model with very few labels.

1102
02:26:40.910 --> 02:26:52.320
Ayoub Ghriss: So one thing you can do is actually use like something that we have just this way you can use. An autoencoder to reduce the dimension, so you can have a speech that has like 10,000 features.

1103
02:26:52.620 --> 02:27:13.040
Ayoub Ghriss: and then you can use an item Coder to map it like to 64, or a hundred in a way that those 100 features are gonna be enough to reconstruct the the original. The original speech in that way, then you can take that code and apply and use it as features for the supervised learning part.

1104
02:27:13.150 --> 02:27:19.980
Ayoub Ghriss: And this is very commonly used. I mean, it's basically used all the time in all machine learning learning models. Now.

1105
02:27:20.500 --> 02:27:25.620
Ayoub Ghriss: especially when the input is high dimensional. So here I did like, 10 dimensions.

1106
02:27:26.760 --> 02:27:32.480
Ayoub Ghriss: I'm getting slightly better, because then features, then dimensions in the code is going to give me more information.

1107
02:27:32.700 --> 02:27:34.890
Ayoub Ghriss: Yeah, this is not gonna work now.

1108
02:27:34.900 --> 02:27:40.410
Ayoub Ghriss: But let me just try something that's 10, like 3, 2, minus one.

1109
02:27:40.690 --> 02:27:44.150
Ayoub Ghriss: minus 2, minus 4.

1110
02:27:45.220 --> 02:27:48.389
Ayoub Ghriss: Okay, I have now 7. Let me add the model.

1111
02:27:51.000 --> 02:27:54.309
Ayoub Ghriss: It's not giving anything. So yeah.

1112
02:27:54.370 --> 02:27:59.320
Ayoub Ghriss: once the the machine increases, it becomes very difficult to generate.

1113
02:27:59.490 --> 02:28:05.149
Ayoub Ghriss: Basically, what happening here, I'm taking a point that is in some empty regions of the code space.

1114
02:28:05.550 --> 02:28:07.620
Ayoub Ghriss: Okay, as a very common theme.

1115
02:28:07.810 --> 02:28:12.879
Ayoub Ghriss: Once you increase the dimension, it becomes exponentially difficult to explore the space.

1116
02:28:13.050 --> 02:28:18.950
Ayoub Ghriss: But I can use those 10 dimensions to do other tasks. For example, I can cluster the data. So

1117
02:28:19.290 --> 02:28:27.989
Ayoub Ghriss: I'm using 2,000. So the original data has like 50,000 samples. I'm just going to use 2,000 of them just to do a fast demonstration.

1118
02:28:28.210 --> 02:28:38.179
Ayoub Ghriss: And here I'm using Tsn ET. Is an unsupervised learning

algorithm that does dimensionality reduction. Okay?

1119
02:28:38.760 --> 02:28:42.419
Ayoub Ghriss: So one way, you can just Google scikit learn.

1120
02:28:45.210 --> 02:28:50.590
Ayoub Ghriss: I'm I'm showing you how to catch the fish. Okay, so just like it learned clustering.

1121
02:28:51.010 --> 02:28:56.870
Ayoub Ghriss: And you have all these machine learning algorithms that you can use for for clustering

1122
02:28:57.360 --> 02:29:09.799
Ayoub Ghriss: all of them here. You can even go to any of these. Yeah, I explained, what's the mathematics behind it? And you can actually just go there. I, for example, go to Key means one of the oldest algorithms.

1123
02:29:12.280 --> 02:29:13.750
Ayoub Ghriss: So

1124
02:29:14.970 --> 02:29:16.910
Ayoub Ghriss: just going to K-means.

1125
02:29:17.930 --> 02:29:19.460
Ayoub Ghriss: And there you go.

1126
02:29:19.670 --> 02:29:23.219
Ayoub Ghriss: So this is the definition of the function.

1127
02:29:23.300 --> 02:29:29.640
Ayoub Ghriss: the different number of parameters. But you're always usually given example of how to use it. Okay?

1128
02:29:31.250 --> 02:29:36.779
Ayoub Ghriss: And they usually have all the same signature. It means that you define the model, and then.

1129

02:29:36.920 --> 02:29:39.500
Ayoub Ghriss: you do the fitting, using the data.

1130
02:29:39.710 --> 02:29:48.790
Ayoub Ghriss: And then you do the prediction. All, usually all the algorithms inside learn have the same structure, define the model, do the fitting and do the prediction.

1131
02:29:50.330 --> 02:29:54.830
Ayoub Ghriss: Yeah, it's a machine learning model. It's unsupervised. It's unsupervised. One

1132
02:29:55.610 --> 02:30:02.419
Ayoub Ghriss: machine learning is not something magical. Just statistics with more powerful computers. Yeah.

1133
02:30:03.980 --> 02:30:08.879
Ayoub Ghriss: okay, so this is the dimensionality reduction of my 10 features. Input.

1134
02:30:10.500 --> 02:30:14.559
Ayoub Ghriss: So in bit is not defined. Okay, I'm just going to fit it here. See what I get

1135
02:30:17.710 --> 02:30:19.510
Ayoub Ghriss: there? You go. Yeah.

1136
02:30:20.440 --> 02:30:27.770
Ayoub Ghriss: So the color here I'm using the true labels to color my, my, my! Embedded here.

1137
02:30:28.250 --> 02:30:41.339
Ayoub Ghriss: and I haven't told. I haven't used the labels in my algorithm at all. I haven't told it. Whether this is one or 2 or 3. I just use the auto encoder, and then I use the sine E, which is a very simple

1138
02:30:41.730 --> 02:30:50.349
Ayoub Ghriss: non-parametric way to reduce the dimension of the input. From 10 to 2, and you can already see that it has

1139
02:30:50.490 --> 02:30:59.599
Ayoub Ghriss: a very good performance at clustering the different digits. So imagine you have someone who has never learned what the numbers are.

1140
02:30:59.850 --> 02:31:03.390
Ayoub Ghriss: and you just tell them to separate the numbers.

1141
02:31:03.610 --> 02:31:12.250
Ayoub Ghriss: separate the images based on similarity. And I would guess, probably around 80 to 90% performance matching the 2 labels.

1142
02:31:12.750 --> 02:31:18.419
Ayoub Ghriss: The algorithm does not know what 0 or one is. It just knows that these are very similar. That's it.

1143
02:31:20.770 --> 02:31:35.369
Ayoub Ghriss: And Autoencoder is one of the simplest, like encoding deep learning models you can use with more powerful ones, with more distributions or assumptions. You can get very like sometimes, like 100% accuracy

1144
02:31:35.390 --> 02:31:39.359
Ayoub Ghriss: on on the eminis data set without telling it to what the real labels are.

1145
02:31:40.400 --> 02:31:41.510
Ayoub Ghriss: Okay.

1146
02:31:45.670 --> 02:31:51.179
Ayoub Ghriss: Now. I want to go back to my original

1147
02:31:52.340 --> 02:31:57.790
Ayoub Ghriss: supervised problem. So I guess the notebook was killed. So I'm just gonna

1148
02:31:59.930 --> 02:32:03.150
Ayoub Ghriss: run everything hopefully, we can have enough time

1149

02:32:03.350 --> 02:32:04.960
Ayoub Ghriss: to get you to

1150
02:32:06.080 --> 02:32:08.680
Ayoub Ghriss: show you the trick example.

1151
02:32:20.740 --> 02:32:23.780
Ayoub Ghriss: So the original question of wanted question was.

1152
02:32:23.910 --> 02:32:29.130
Ayoub Ghriss: how relevant is the background features of the image.

1153
02:32:29.560 --> 02:32:33.330
Ayoub Ghriss: And it actually took me some time to build this

1154
02:32:33.410 --> 02:32:36.390
Ayoub Ghriss: tricky data set in a way that

1155
02:32:37.550 --> 02:32:44.259
Ayoub Ghriss: I have made deliberately that all the dogs pictures had some green in them.

1156
02:32:46.680 --> 02:32:57.220
Ayoub Ghriss: Okay, thank you. Yeah. So in this, in this data set that I have bold, I am and amateur that the dog pictures, almost all of them have some greenery in the background.

1157
02:32:57.870 --> 02:33:01.939
Ayoub Ghriss: So what happens here is that if I take some

1158
02:33:03.200 --> 02:33:05.530
Ayoub Ghriss: images. that

1159
02:33:05.970 --> 02:33:11.069
Ayoub Ghriss: of dogs that don't have the green background, then we can see that

1160
02:33:12.230 --> 02:33:18.619
Ayoub Ghriss: the algorithm actually thinks it's a dog. So the neural

network that I trained

1161
02:33:19.150 --> 02:33:28.060
Ayoub Ghriss: thinks that the label dog is actually due to the green background. not the features of of the animal.

1162
02:33:31.100 --> 02:33:33.080
Ayoub Ghriss: So let me see here.

1163
02:33:35.530 --> 02:33:38.510
Ayoub Ghriss: Yeah, it's not doing that. Well, let me just

1164
02:33:39.660 --> 02:33:41.000
Ayoub Ghriss: use something.

1165
02:33:46.860 --> 02:33:47.850
Ayoub Ghriss: Okay.

1166
02:33:50.780 --> 02:33:51.650
Ayoub Ghriss: right?

1167
02:33:55.430 --> 02:33:57.630
Ayoub Ghriss: So really, look at the

1168
02:34:03.040 --> 02:34:06.739
Ayoub Ghriss: let's see, I'm just gonna wait for that. But

1169
02:34:08.270 --> 02:34:12.510
Ayoub Ghriss: that takes me to the last question or the last part of the

1170
02:34:12.580 --> 02:34:15.609
Ayoub Ghriss: machine learning presentation here

1171
02:34:16.030 --> 02:34:17.830
Ayoub Ghriss: where he talked about dance.

1172
02:34:19.210 --> 02:34:20.690

Ayoub Ghriss: So, okay.

1173
02:34:22.240 --> 02:34:29.290
Ayoub Ghriss: so now, imagine I want to use this deep learning model to build some robot that feeds my pets.

1174
02:34:29.510 --> 02:34:34.010
Ayoub Ghriss: And I have a dog and a cat. So let's say it feeds them twice a day.

1175
02:34:34.500 --> 02:34:39.580
Ayoub Ghriss: But apparently it does not do that. Well when there's not greenery behind.

1176
02:34:39.950 --> 02:34:42.330
Ayoub Ghriss: so it just assumes that it's always a dog.

1177
02:34:43.370 --> 02:34:47.779
Ayoub Ghriss: So in this case, if the dog eats first

1178
02:34:47.970 --> 02:34:50.860
Ayoub Ghriss: in the morning like, say, 2 times in the morning.

1179
02:34:51.180 --> 02:34:59.010
Ayoub Ghriss: and the cat can afterwards. It's not going to feed the cat because it's going to take swimming. It's a dog. It's not going to give it some food light

1180
02:34:59.090 --> 02:35:00.300
Ayoub Ghriss: layer itself.

1181
02:35:00.680 --> 02:35:04.849
Ayoub Ghriss: This is where the question of of fairness or the problem of

1182
02:35:05.990 --> 02:35:16.310
Ayoub Ghriss: the unexpected consequences of using a machine learning model without understanding what actually happens behind it. So let's say here, it seems like it's good. Let me

1183
02:35:16.550 --> 02:35:18.300
Ayoub Ghriss: but load my tricky data.

1184
02:35:19.720 --> 02:35:24.840
Ayoub Ghriss: Oops. what is test? I haven't used test.

1185
02:35:24.860 --> 02:35:29.070
Elsa Culler: Ayub, I think we're still on the slides here.

1186
02:35:32.310 --> 02:35:33.430
Ayoub Ghriss: Hey.

1187
02:35:34.900 --> 02:35:36.790
Ayoub Ghriss: yeah, I'm just saying the

1188
02:35:40.660 --> 02:35:41.420
Ayoub Ghriss: yeah.

1189
02:35:42.640 --> 02:35:51.980
Ayoub Ghriss: So here I'm using other samples where the the cats have actually a green background. And the dogs don't have. we have background. And it has 0 accuracy.

1190
02:35:53.620 --> 02:36:00.730
Ayoub Ghriss: yeah. So in machine learning model, you always want to pay attention to

1191
02:36:01.200 --> 02:36:04.550
Ayoub Ghriss: the data that you are using. And

1192
02:36:05.070 --> 02:36:09.750
Ayoub Ghriss: you have very famous applications like in credits,

1193
02:36:10.010 --> 02:36:14.550
Ayoub Ghriss: in in. And what she didn't know those used byte banks to do some

1194
02:36:14.560 --> 02:36:21.360

Ayoub Ghriss: Loan credit score prediction, or whether you are

1195
02:36:22.850 --> 02:36:26.829
Ayoub Ghriss: a good candidate to get to take a loan based on some machine learning model

1196
02:36:27.060 --> 02:36:33.619
Ayoub Ghriss: that if the features have some bias in them you might be unlikely. I created this. So if I moved to the fairness part.

1197
02:36:34.210 --> 02:36:38.850
Ayoub Ghriss: I created this census synthetic data set where

1198
02:36:40.410 --> 02:36:44.140
Ayoub Ghriss: they say you have set of students.

1199
02:36:44.170 --> 02:36:55.869
Ayoub Ghriss: This is like the speed which is like the number of seconds. They're gonna run a certain distance the swimmings like the distance they're gonna swim in a certain period of time, their Gpa, and then their eye color.

1200
02:36:56.460 --> 02:37:02.120
Ayoub Ghriss: Okay? And then here is basically the label means they're probably accepting a certain sports program.

1201
02:37:03.330 --> 02:37:06.540
Ayoub Ghriss: So when I do this, I'm using sorry

1202
02:37:11.360 --> 02:37:14.850
Ayoub Ghriss: there's no such file. Wait

1203
02:37:16.100 --> 02:37:17.210
sips.

1204
02:37:20.830 --> 02:37:23.960
Ayoub Ghriss: Ok, I guess I have to rerun the notebook. Let me just say

1205

02:37:25.310 --> 02:37:26.320
Ayoub Ghriss: starts.

1206
02:37:29.640 --> 02:37:32.270
Ayoub Ghriss: oh, okay. Cause I change the path. Okay.

1207
02:37:33.520 --> 02:37:34.240
Ayoub Ghriss: yeah.

1208
02:37:36.220 --> 02:37:42.830
Ayoub Ghriss: So the data set is equally balanced between accepted or not accepted. And I'm using a decision tree algorithm

1209
02:37:43.100 --> 02:37:47.759
Ayoub Ghriss: to just train my my model here. And here I'm showing

1210
02:37:47.800 --> 02:37:48.880
Ayoub Ghriss: my

1211
02:37:49.680 --> 02:38:00.349
Ayoub Ghriss: the decision tree or the decision logic. So here it's looking at x 3. So this third feature. So in this case the the features start with 0 0 1, 2, 3, it's the eye color.

1212
02:38:00.670 --> 02:38:06.620
Ayoub Ghriss: So what happens here is that if the eye color is 0, it's always yours, never accepted.

1213
02:38:07.370 --> 02:38:15.170
Ayoub Ghriss: And the reason is that the eye color is highly correlated with the acceptance label.

1214
02:38:16.820 --> 02:38:21.870
Ayoub Ghriss: Okay? So even though if I evaluate my my machine learning model.

1215
02:38:22.210 --> 02:38:25.280
Ayoub Ghriss: wait! What is the score here?

1216

02:38:27.260 --> 02:38:29.499
Ayoub Ghriss: So I'm using the accuracy score

1217
02:38:29.560 --> 02:38:32.289
Ayoub Ghriss: I fit the model. I plot the tree.

1218
02:38:32.610 --> 02:38:34.670
Ayoub Ghriss: Let me evaluate this. Okay?

1219
02:38:42.930 --> 02:38:44.739
Ayoub Ghriss: So the way I evaluate it.

1220
02:38:44.970 --> 02:38:48.740
Ayoub Ghriss: Thought, I put this somewhere. Yeah. But let me just do

1221
02:38:49.140 --> 02:38:50.890
Ayoub Ghriss: its accuracy

1222
02:38:52.060 --> 02:38:53.150
Ayoub Ghriss: score.

1223
02:38:54.190 --> 02:38:57.250
Ayoub Ghriss: and I do then Cllf. Predict

1224
02:38:59.350 --> 02:39:06.940
Ayoub Ghriss: so the order, whether the prediction is first or not is not important for accuracy, but is important for the other models. And

1225
02:39:17.380 --> 02:39:18.620
Ayoub Ghriss: so drop

1226
02:39:24.460 --> 02:39:25.590
Ayoub Ghriss: K,

1227
02:39:29.610 --> 02:39:37.799
Ayoub Ghriss: okay, so in the performance, I'm getting like 70% accuracy. But I'm only getting getting that because I put everyone with

1228
02:39:38.210 --> 02:39:44.260
Ayoub Ghriss: my color 0 whatever that is, black or red, whatever you prefer it just given is like.

1229
02:39:44.570 --> 02:39:46.190
Ayoub Ghriss: not

1230
02:39:47.260 --> 02:40:00.620
Ayoub Ghriss: not been accepted. So if I'm just evaluating the model based on the performance. if 70% is enough for me. I'm not looking at the consequences of actually excluding

1231
02:40:01.310 --> 02:40:09.790
Ayoub Ghriss: students that even might have better speeds women or Gpa. But because my machine learning model actually based this decision

1232
02:40:09.800 --> 02:40:14.299
Ayoub Ghriss: almost uniquely at the First Level on the eye color undiscriminating

1233
02:40:14.640 --> 02:40:19.159
Ayoub Ghriss: against the 0 icon. Okay? You might think then.

1234
02:40:19.430 --> 02:40:22.900
Ayoub Ghriss: well, just the drop, the eye, color, feature. What do you think

1235
02:40:23.160 --> 02:40:24.420
Ayoub Ghriss: we have? 4 min?

1236
02:40:32.650 --> 02:40:36.110
Ayoub Ghriss: What? How would you use this learning improvements?

1237
02:40:44.300 --> 02:40:52.320
Ayoub Ghriss: No guesses. Okay. so I'm going to do the same thing here. Just gonna guessing. Here, I'm gonna remove the eye color.

1238
02:40:54.470 --> 02:40:56.470

Ayoub Ghriss: And let's see what we get.

1239
02:40:56.510 --> 02:40:58.359
Ayoub Ghriss: I'm dropping the accepted.

1240
02:41:02.530 --> 02:41:06.969
Ayoub Ghriss: This is the fairness notebook. I would use learning. Reap on it.

1241
02:41:08.190 --> 02:41:13.460
Ayoub Ghriss: Okay, so I'm dropping here the I color. And I'm gonna see what's happening here. So I don't have the feature

1242
02:41:14.980 --> 02:41:23.059
Ayoub Ghriss: it. But and then the scoring. I'm still getting the same performance. It means that I'm still actually taking the same decision.

1243
02:41:24.450 --> 02:41:26.709
Ayoub Ghriss: Do you want to like any guesses? Why.

1244
02:41:28.960 --> 02:41:36.649
Ayoub Ghriss: even though I drop the eye color, the performance of the algorithm is still the same and is still making the same decisions of excluding the 0 eye color.

1245
02:41:47.640 --> 02:41:51.060
Ayoub Ghriss: Okay, so this will be your homework. And

1246
02:41:54.200 --> 02:41:58.889
Ayoub Ghriss: if you actually define now, let's build an algorithm where we drop the accepted

1247
02:41:59.150 --> 02:42:03.579
Ayoub Ghriss: and we're going to try to guess what's the eye color

1248
02:42:03.600 --> 02:42:06.100
Ayoub Ghriss: based on these 3 features.

1249

02:42:07.090 --> 02:42:13.770
Ayoub Ghriss: So this artificial or this synthetic data set? If you do, you do the analysis so you can

1250
02:42:13.910 --> 02:42:19.079
Ayoub Ghriss: try to predict the eye color based on the speed, the swimming. And Gpa.

1251
02:42:21.750 --> 02:42:24.020
Ayoub Ghriss: can you clarify what that means?

1252
02:42:25.800 --> 02:42:34.590
Ayoub Ghriss: Are you asking about what I'm doing now? Okay. so my, what I'm saying here is that the eye color, or these 3? These

1253
02:42:35.450 --> 02:42:40.079
Ayoub Ghriss: These 3 features are, are highly correlated to the eye color in a way that

1254
02:42:40.420 --> 02:42:46.239
Ayoub Ghriss: I can guess the eye color with high accuracy based on these 3 features. So let's say.

1255
02:42:46.520 --> 02:42:54.889
Ayoub Ghriss: I'm going to use something very simple. Just going to use one here. I'm dropping accepted. And I color. And I'm just gonna use

1256
02:42:56.200 --> 02:42:59.590
Ayoub Ghriss: the target's variable length is going to be a color.

1257
02:43:02.450 --> 02:43:12.089
Ayoub Ghriss: It's not really rigorous, because I'm training and evaluating on the same data. But just give you an idea of the correlation or the high correlation that exists

1258
02:43:12.120 --> 02:43:15.150
Ayoub Ghriss: between the the features.

1259
02:43:27.770 --> 02:43:30.709

Ayoub Ghriss: Yeah, I'm actually predicting. I call it better than

1260
02:43:30.980 --> 02:43:32.550
Ayoub Ghriss: than the accepted ratio.

1261
02:43:34.820 --> 02:43:35.680
Ayoub Ghriss: So

1262
02:43:36.080 --> 02:43:48.759
Ayoub Ghriss: this feature here has been engineered. But it can also happen. For example, your Zip code can be enough information to guess your salary level, or vice versa. So

1263
02:43:49.070 --> 02:43:53.820
Ayoub Ghriss: and this is another thing in the ethical part of machine learning, which is that.

1264
02:43:54.410 --> 02:44:02.749
Ayoub Ghriss: how can I make sure that the features I'm using in my in my machine learning does not reveal sensitive information about

1265
02:44:02.950 --> 02:44:06.849
Ayoub Ghriss: the individuals. or whatever the phenomenon is.

1266
02:44:07.080 --> 02:44:10.960
Ayoub Ghriss: there's something called privacy in machine learning

1267
02:44:11.050 --> 02:44:17.220
Ayoub Ghriss: and also differential privacy, because sometimes you train neural networks

1268
02:44:17.650 --> 02:44:22.280
Ayoub Ghriss: in a way that you don't want, then no network to catch any

1269
02:44:22.580 --> 02:44:28.550
Ayoub Ghriss: information that is sensitive, based on, based on the features.

1270

02:44:29.060 --> 02:44:35.049
Ayoub Ghriss: I guess I covered everything I had to from these 3 notebooks.

1271
02:44:35.170 --> 02:44:38.870
Ayoub Ghriss: You still have the enforcement learning notebook that if you wanna use

1272
02:44:39.130 --> 02:44:45.979
Ayoub Ghriss: it's more like playing with an environment where you have the agents, the white dots. That's right to reach the red dot.

1273
02:44:46.500 --> 02:44:54.969
Ayoub Ghriss: The difference between this and what you would probably see in computer science, where you're trying to find like, what's the shortest path is that here

1274
02:44:55.180 --> 02:45:14.129
Ayoub Ghriss: the algorithm does not care about the structure of the environment. The only thing that it cares about is what's the different actions that they can take. And when the game ends, and basically, when you reach the final goal you reach, you get a reward doesn't matter how many rooms doesn't matter how large the environment is.

1275
02:45:14.470 --> 02:45:20.310
Ayoub Ghriss: So it's more like, a generic way of training an algorithm that can reach a certain goal.

1276
02:45:20.700 --> 02:45:21.870
Ayoub Ghriss: And

1277
02:45:22.180 --> 02:45:36.530
Ayoub Ghriss: it is different from supervising unsupervised in a way that you just provide the environment to the algorithm and the algorithm explores the environment itself. So it's basically it's generating its own labels based on experience.

1278
02:45:37.760 --> 02:45:45.430
Ayoub Ghriss: And it's a bit mathematical, link, a time to simplify it in a way. But if you just run the notebook, you can get

1279
02:45:45.510 --> 02:46:01.499
Ayoub Ghriss: a demonstration of what's happening. There's this video at the end. So it shows you how things are happening when you choose a Rand, a uniform random. So you just blindly explore the environment and the best one, she means that the one that you have you learned based on from the algorithm.

1280
02:46:02.580 --> 02:46:03.240
Ayoub Ghriss: It

1281
02:46:07.350 --> 02:46:13.060
Ayoub Ghriss: okay. any questions. I'm aware this can be kind of

1282
02:46:13.690 --> 02:46:20.190
Ayoub Ghriss: overwhelming for people who haven't seen this before, but I guess I gave all the ingredients that you need

1283
02:46:20.400 --> 02:46:26.209
Ayoub Ghriss: how to explore the Keras Library, the different.

1284
02:46:26.410 --> 02:46:37.319
Ayoub Ghriss: This is just an example how to do a simple neural network on also how to use psychic learn like unsupervised learning. But the supervised one is the same. You're just providing new labels.

1285
02:46:37.750 --> 02:46:40.279
Ayoub Ghriss: and these 2 libraries are

1286
02:46:40.910 --> 02:46:42.680
Ayoub Ghriss: very well documented.

1287
02:46:42.750 --> 02:46:49.089
Ayoub Ghriss: Like Kami's is everything is explained here. They're also giving references and everything.

1288
02:46:49.300 --> 02:46:59.590
Ayoub Ghriss: And the best way for machine learning just to run the code and see what's happening. Then go back to the equations. Otherwise, just starting from the theory can be confusing to a lot of people.

1289
02:47:01.790 --> 02:47:07.499
Ayoub Ghriss: Yeah, with that, I'm ending my presentation. Any questions I'm here.

1290
02:47:07.590 --> 02:47:09.530
Ayoub Ghriss: Otherwise. Thank you for my time.

1291
02:47:10.690 --> 02:47:16.779
Nate Quarderer (Earth Lab/ ESIIL): Awesome. Hey? Can we give it up for our presenter? Are you? Thank you so much, great job.

1292
02:47:18.770 --> 02:47:29.510
Nate Quarderer (Earth Lab/ ESIIL): Kai wants to remind people to shut down your virtual machine before you leave, so please be sure to do that, and give by you some love in the chat or with emojis, or, however, you prefer to do that.

1293
02:47:29.690 --> 02:47:37.420
Nate Quarderer (Earth Lab/ ESIIL): We're we're running up on time. I wanted to just turn it over to Virginia quick! Do you wanna make any announcements or say anything about

1294
02:47:37.500 --> 02:47:42.570
Nate Quarderer (Earth Lab/ ESIIL): next week or the trainings before we dismiss people for the day.

1295
02:47:44.080 --> 02:48:00.390
Virginia Iglesias: Sure, we'll be sending out an email with information. There will be a link to a web page where you'll find a ton of information so hopefully that will answer any questions that you will need for the hackathon.

1296
02:48:00.640 --> 02:48:07.859
Virginia Iglesias: and so yep, looking forward to seeing you all, and thank you for being here

1297
02:48:09.200 --> 02:48:15.610
Nate Quarderer (Earth Lab/ ESIIL): awesome. Virginia, give it up for all of our presenters again. Also Eric Tybee.

1298
02:48:16.800 --> 02:48:22.750
Nate Quarderer (Earth Lab/ ESIIL): So belly probably forgetting some folks. Great job. Everyone. Thanks again for everybody's help. Great job team.

1299
02:48:23.330 --> 02:48:26.650
Nate Quarderer (Earth Lab/ ESIIL): Thank you. Rachel and Virginia, for keeping us on track

1300
02:48:27.990 --> 02:48:36.670
Nate Quarderer (Earth Lab/ ESIIL): do some typing calisthenics. We got a big hackathon coming up next week. Party people. Yes.

1301
02:48:36.940 --> 02:48:48.069
Nate Quarderer (Earth Lab/ ESIIL): we look forward to seeing you next week. Everyone don't forget to stretch, get lots of rest, come ready to ask questions and participate, and feel free to reach out to us. If you need anything before then.