WEBVTT

NOTE duration:"01:19:06"

NOTE recognizability:0.843

NOTE language:en-us

NOTE Confidence: 0.719259674

00:00:00.000 --> 00:00:02.860 Today is the second lecture,

NOTE Confidence: 0.719259674

00:00:02.860 --> 00:00:06.455 second presentation in our series and

NOTE Confidence: 0.719259674

00:00:06.455 --> 00:00:11.600 I'm very excited to introduce Rahul Sai.

NOTE Confidence: 0.719259674

00:00:11.600 --> 00:00:15.062 Rahul is an expert in advanced

NOTE Confidence: 0.719259674

00:00:15.062 --> 00:00:18.080 statistical method including graph theory,

NOTE Confidence: 0.719259674

00:00:18.080 --> 00:00:19.492 network analysis,

NOTE Confidence: 0.719259674

00:00:19.492 --> 00:00:22.316 neural network signal processing

NOTE Confidence: 0.719259674

00:00:22.316 --> 00:00:25.474 and many other impressive terms.

NOTE Confidence: 0.719259674

00:00:25.474 --> 00:00:28.576 He received the PhD in machine

NOTE Confidence: 0.719259674

00:00:28.576 --> 00:00:31.042 learning in Georgia Institute of

NOTE Confidence: 0.719259674

00:00:31.042 --> 00:00:34.178 Technology and now he is a post

NOTE Confidence: 0.719259674

00:00:34.266 --> 00:00:38.880 doctoral researcher in Wosai Institute.

NOTE Confidence: 0.719259674

00:00:38.880 --> 00:00:40.836 And I'm not going to take

NOTE Confidence: 0.719259674

00:00:40.836 --> 00:00:42.520 any more time from you.

NOTE Confidence: 0.719259674

00:00:42.520 --> 00:00:44.520 Please, it's all yours.

NOTE Confidence: 0.91571047

00:00:46.200 --> 00:00:48.840 Thank you, Helen. Hello everyone.

NOTE Confidence: 0.91571047

00:00:48.840 --> 00:00:53.455 I'm Rahul Singh, welcome to the

NOTE Confidence: 0.91571047

00:00:53.455 --> 00:00:56.210 second lecture of this workshop.

NOTE Confidence: 0.91571047

00:00:56.210 --> 00:00:59.150 And today I'm going to talk

NOTE Confidence: 0.91571047

00:00:59.150 --> 00:01:01.320 about graph signal processing.

NOTE Confidence: 0.91571047

00:01:01.320 --> 00:01:03.820 It's like extending the classical

NOTE Confidence: 0.91571047

00:01:03.820 --> 00:01:05.200 signal processing techniques

NOTE Confidence: 0.91571047

00:01:05.200 --> 00:01:07.040 over manifolds or graphs.

NOTE Confidence: 0.873643374

00:01:10.760 --> 00:01:12.280 A little bit about me,

NOTE Confidence: 0.873643374

00:01:12.280 --> 00:01:16.648 I started my academic journey back

NOTE Confidence: 0.873643374

00:01:16.648 --> 00:01:21.171 in India where I studied the signal

NOTE Confidence: 0.873643374

00:01:21.171 --> 00:01:23.319 processing from Indian Institute

NOTE Confidence: 0.873643374

00:01:23.319 --> 00:01:26.200 of Space Science and Technology.

NOTE Confidence: 0.873643374

00:01:26.200 --> 00:01:28.520 And then I went to Iowa State University,

NOTE Confidence: 0.873643374

00:01:28.520 --> 00:01:30.985 where I got my master's

NOTE Confidence: 0.873643374

00:01:30.985 --> 00:01:32.957 degree in image processing.

NOTE Confidence: 0.873643374

00:01:32.960 --> 00:01:34.815 And then I went on for my

NOTE Confidence: 0.873643374

00:01:34.815 --> 00:01:36.670 doctoral studies at Georgia Tech,

NOTE Confidence: 0.873643374

00:01:36.670 --> 00:01:40.400 where I got my PhD in machine learning.

NOTE Confidence: 0.873643374

00:01:40.400 --> 00:01:43.318 And here my most along the line,

NOTE Confidence: 0.873643374

00:01:43.318 --> 00:01:45.776 my research was in signal processing,

NOTE Confidence: 0.873643374

00:01:45.776 --> 00:01:47.504 extending the signal processing

NOTE Confidence: 0.873643374

00:01:47.504 --> 00:01:49.676 techniques to graphs and especially

NOTE Confidence: 0.873643374

00:01:49.676 --> 00:01:53.180 using the structure in the data and how

NOTE Confidence: 0.873643374

00:01:53.264 --> 00:01:56.520 we can process the data that that has

NOTE Confidence: 0.873643374

00:01:56.520 --> 00:01:58.640 some intrinsic structure behind it.

NOTE Confidence: 0.873643374

00:01:58.640 --> 00:02:01.752 And here I am a post doctor fellow

NOTE Confidence: 0.873643374

00:02:01.752 --> 00:02:04.814 at Woosa Institute and I'm doing,

NOTE Confidence: 0.873643374

00:02:04.814 --> 00:02:07.104 I'm applying this machine learning

NOTE Confidence: 0.873643374

00:02:07.104 --> 00:02:09.480 and signal processing methods

NOTE Confidence: 0.873643374

00:02:09.480 --> 00:02:11.400 in computational neuroscience.

NOTE Confidence: 0.873643374

00:02:11.400 --> 00:02:15.080 And my mentors are Smita Krishnaswami

NOTE Confidence: 0.873643374

00:02:15.080 --> 00:02:17.880 from computer science and Joy

NOTE Confidence: 0.873643374

00:02:17.880 --> 00:02:19.560 Hirsch from neuroscience.

NOTE Confidence: 0.852095525555556

00:02:23.400 --> 00:02:28.160 Moving on to our main toolbox that we

NOTE Confidence: 0.852095525555556

00:02:28.160 --> 00:02:32.973 are in the process to learn geometric

NOTE Confidence: 0.852095525555556

00:02:32.973 --> 00:02:35.772 scattering trajective homology.

NOTE Confidence: 0.852095525555556

00:02:35.772 --> 00:02:41.053 So as Tenanje last week has

NOTE Confidence: 0.852095525555556

00:02:41.053 --> 00:02:44.118 given an introduction about it,

NOTE Confidence: 0.852095525555556

00:02:44.120 --> 00:02:46.200 there are four steps here.

NOTE Confidence: 0.852095525555556

00:02:46.200 --> 00:02:48.840 So any data that is residing on a

NOTE Confidence: 0.852095525555556

00:02:48.840 --> 00:02:51.475 manifold or if you have just the data,

NOTE Confidence: 0.852095525555556

00:02:51.480 --> 00:02:54.056 the first step to create the manifold

NOTE Confidence: 0.852095525555556

00:02:54.056 --> 00:02:57.240 or the graph that this graph creation.

NOTE Confidence: 0.852095525555556

00:02:57.240 --> 00:03:00.800 And then we apply this graph signal

NOTE Confidence: 0.852095525555556

00:03:00.800 --> 00:03:02.860 processing tools to represent

NOTE Confidence: 0.852095525555556

00:03:02.860 --> 00:03:06.040 the data lying on this graph.

NOTE Confidence: 0.852095525555556

00:03:06.040 --> 00:03:08.080 And then we do some non,

NOTE Confidence: 0.852095525555556

00:03:08.080 --> 00:03:10.476 some dimensionality reduction techniques

NOTE Confidence: 0.852095525555556

00:03:10.476 --> 00:03:12.872 to visualize these trajectories

NOTE Confidence: 0.852095525555556

00:03:12.872 --> 00:03:16.196 of the data that evolve over time.

NOTE Confidence: 0.852095525555556

00:03:16.200 --> 00:03:18.965 And we analyze these trajectories

NOTE Confidence: 0.852095525555556

00:03:18.965 --> 00:03:21.544 using topological data analysis that

NOTE Confidence: 0.852095525555556

00:03:21.544 --> 00:03:24.074 Tananjay talked about last week.

NOTE Confidence: 0.852095525555556

00:03:24.080 --> 00:03:26.880 So step four we covered last week.

NOTE Confidence: 0.852095525555556

00:03:26.880 --> 00:03:29.280 Today we are going to cover step two,

NOTE Confidence: 0.852095525555556

00:03:29.280 --> 00:03:31.891 that is how to process the data

NOTE Confidence: 0.852095525555556

00:03:31.891 --> 00:03:33.640 lying on a graph.

NOTE Confidence: 0.852095525555556

00:03:33.640 --> 00:03:37.240 And once after today's lecture

NOTE Confidence: 0.852095525555556

00:03:37.240 --> 00:03:40.120 in the third series,

NOTE Confidence: 0.852095525555556

00:03:40.120 --> 00:03:43.048 Dhananjay and Brian are going to

NOTE Confidence: 0.852095525555556

00:03:43.048 --> 00:03:46.074 combine these two techniques to give

NOTE Confidence: 0.852095525555556

00:03:46.074 --> 00:03:49.700 you a overall picture of this geometric

NOTE Confidence: 0.852095525555556

00:03:49.791 --> 00:03:52.560 scattering trajectory homology.

NOTE Confidence: 0.852095525555556

00:03:52.560 --> 00:03:54.480 So let's move on to the second Step,

NOTE Confidence: 0.852095525555556

00:03:54.480 --> 00:03:56.040 2, graph signal processing.

NOTE Confidence: 0.852095525555556

00:03:56.040 --> 00:03:58.380 So what is this graph signal

NOTE Confidence: 0.852095525555556

00:03:58.455 --> 00:03:59.639 processing about?

NOTE Confidence: 0.66339294

00:04:01.960 --> 00:04:05.474 Mostly we in general we have seen

NOTE Confidence: 0.66339294

00:04:05.480 --> 00:04:07.892 there are many tools and concepts

NOTE Confidence: 0.66339294

00:04:07.892 --> 00:04:10.481 that have been developed in classical

NOTE Confidence: 0.66339294

00:04:10.481 --> 00:04:13.194 signal processing domain where we deal

NOTE Confidence: 0.66339294

00:04:13.194 --> 00:04:16.550 with data that is in form of a time

NOTE Confidence: 0.66339294

00:04:16.550 --> 00:04:18.640 series for example, speech, music,

NOTE Confidence: 0.8317622425

00:04:19.680 --> 00:04:20.920 e.g, FM, RI, whatever

NOTE Confidence: 0.650271236666667

00:04:24.080 --> 00:04:24.920 or we also

NOTE Confidence: 0.876820397272727

00:04:24.920 --> 00:04:26.735 deal with images.

NOTE Confidence: 0.876820397272727

00:04:26.735 --> 00:04:31.720 So these are the signals or the data.

NOTE Confidence: 0.876820397272727

00:04:31.720 --> 00:04:34.380 For example, this here is a speech

NOTE Confidence: 0.876820397272727

00:04:34.380 --> 00:04:36.296 signal and this is speech signal is

NOTE Confidence: 0.876820397272727

00:04:36.296 --> 00:04:38.200 for some let's say for 10 seconds.

NOTE Confidence: 0.876820397272727

00:04:38.200 --> 00:04:41.980 And we can see that once we

NOTE Confidence: 0.876820397272727

00:04:41.980 --> 00:04:44.114 sample this data digitally,

NOTE Confidence: 0.876820397272727

00:04:44.114 --> 00:04:47.198 then this is nothing but some

NOTE Confidence: 0.876820397272727

00:04:47.200 --> 00:04:50.480 values at each time point.

NOTE Confidence: 0.876820397272727

00:04:50.480 --> 00:04:52.874 And if you look at these values and look

NOTE Confidence: 0.876820397272727

00:04:52.874 --> 00:04:55.320 at the structure behind these values,

NOTE Confidence: 0.876820397272727

00:04:55.320 --> 00:04:59.360 it is just a very linear looking structure.

NOTE Confidence: 0.876820397272727

00:04:59.360 --> 00:05:03.714 So T1 time T1 is related to T2T2

NOTE Confidence: 0.876820397272727

00:05:03.714 --> 00:05:07.440 is related to TT three because when

NOTE Confidence: 0.876820397272727

00:05:07.440 --> 00:05:09.840 you speak it's related with time.

NOTE Confidence: 0.876820397272727

00:05:09.840 --> 00:05:11.070 And in images,

NOTE Confidence: 0.876820397272727

00:05:11.070 --> 00:05:13.940 the data or the signal that constitute

NOTE Confidence: 0.876820397272727

00:05:14.023 --> 00:05:16.630 this image is nothing but this

NOTE Confidence: 0.876820397272727

00:05:16.630 --> 00:05:21.320 RGB values at lying on this grid.

NOTE Confidence: 0.876820397272727

00:05:21.320 --> 00:05:24.272 So each pixel of this image is a data

NOTE Confidence: 0.876820397272727

00:05:24.272 --> 00:05:27.360 value lying on this particular blue node.

NOTE Confidence: 0.876820397272727

00:05:27.360 --> 00:05:30.237 And once we kind of arrange these

NOTE Confidence: 0.876820397272727

00:05:30.240 --> 00:05:32.560 intensity values of each pixels,

NOTE Confidence: 0.876820397272727

00:05:32.560 --> 00:05:36.354 it takes a form of an image.

NOTE Confidence: 0.876820397272727

00:05:36.360 --> 00:05:37.640 So in classical signal processing,

NOTE Confidence: 0.876820397272727

00:05:37.640 --> 00:05:39.880 we deal with this kind of regular data.

NOTE Confidence: 0.876820397272727

00:05:39.880 --> 00:05:42.118 Regular in the sense the structure

NOTE Confidence: 0.876820397272727

00:05:42.118 --> 00:05:44.600 behind the data is very regular.

NOTE Confidence: 0.876820397272727

00:05:44.600 --> 00:05:47.582 Regular means if you see this

NOTE Confidence: 0.876820397272727

00:05:47.582 --> 00:05:50.233 graph here or little structure

NOTE Confidence: 0.876820397272727

00:05:50.233 --> 00:05:54.260 that is behind this data is just

NOTE Confidence: 0.876820397272727

00:05:54.260 --> 00:05:57.690 one-dimensional path graph and the

NOTE Confidence: 0.876820397272727

00:05:57.690 --> 00:05:59.880 image of image values are lying

NOTE Confidence: 0.876820397272727

00:05:59.880 --> 00:06:02.319 on this two-dimensional grid.

NOTE Confidence: 0.851514991428571

00:06:04.360 --> 00:06:06.120 And in classical signal processing

NOTE Confidence: 0.851514991428571

00:06:06.120 --> 00:06:08.286 there are other concepts like our

NOTE Confidence: 0.851514991428571

00:06:08.286 --> 00:06:10.440 tools like modulation and translation,

NOTE Confidence: 0.925352145

00:06:12.920 --> 00:06:14.760 frequency analysis and all.

NOTE Confidence: 0.925352145

00:06:14.760 --> 00:06:16.520 For example. Why do we need this?

NOTE Confidence: 0.925352145

00:06:16.520 --> 00:06:18.920 What is this modulation?

NOTE Confidence: 0.925352145

00:06:18.920 --> 00:06:24.164 If you connect your thoughts to your radio

NOTE Confidence: 0.925352145

00:06:24.164 --> 00:06:27.400 or even the cable TV or even cell phones,

NOTE Confidence: 0.925352145

00:06:27.400 --> 00:06:30.230 they are working on different

NOTE Confidence: 0.925352145

00:06:30.230 --> 00:06:33.320 frequencies and the data is around like

NOTE Confidence: 0.925352145

00:06:33.320 --> 00:06:36.915 it's it's in the air it it has it is

NOTE Confidence: 0.925352145

00:06:36.920 --> 00:06:40.208 being sent in form of electromagnetic

NOTE Confidence: 0.925352145

00:06:40.208 --> 00:06:42.400 waves through multiple antennas.

NOTE Confidence: 0.925352145

00:06:42.400 --> 00:06:43.212 But all the data,

NOTE Confidence: 0.925352145

00:06:43.212 --> 00:06:45.039 like if you look at if you're sitting

NOTE Confidence: 0.925352145

00:06:45.039 --> 00:06:47.239 in your room there you have Wi-Fi data,

NOTE Confidence: 0.925352145

00:06:47.240 --> 00:06:49.800 your phone data, or even FM,

NOTE Confidence: 0.880805001176471

00:06:51.960 --> 00:06:53.412 A FM radio, whatever.

NOTE Confidence: 0.880805001176471

00:06:53.412 --> 00:06:56.069 But now you need to kind of

NOTE Confidence: 0.880805001176471

00:06:56.069 --> 00:06:58.439 select what you want to receive.

NOTE Confidence: 0.880805001176471

00:06:58.440 --> 00:07:00.360 And in order to select that,

NOTE Confidence: 0.880805001176471

00:07:00.360 --> 00:07:02.316 there is a concept called modulation.

NOTE Confidence: 0.880805001176471

00:07:02.320 --> 00:07:04.605 So you kind of modulate

NOTE Confidence: 0.880805001176471

00:07:04.605 --> 00:07:06.433 your voice band frequencies.

NOTE Confidence: 0.880805001176471

00:07:06.440 --> 00:07:08.232 For example, somebody's talking,

NOTE Confidence: 0.880805001176471

00:07:08.232 --> 00:07:10.024 the frequencies are just

NOTE Confidence: 0.880805001176471

00:07:10.024 --> 00:07:11.786 within like 3.2 kilohertz,

NOTE Confidence: 0.880805001176471

00:07:11.786 --> 00:07:13.518 but you kind of

NOTE Confidence: 0.907403906666667

00:07:15.680 --> 00:07:18.332 shift this frequency to higher frequency

NOTE Confidence: 0.907403906666667

00:07:18.332 --> 00:07:21.768 so that it becomes easy to transmit the

NOTE Confidence: 0.907403906666667

00:07:21.768 --> 00:07:24.680 data because the antenna size is always

NOTE Confidence: 0.823594342857143

00:07:28.080 --> 00:07:30.188 proportional to the wavelength.

NOTE Confidence: 0.823594342857143

00:07:30.188 --> 00:07:32.823 So modulating to high frequencies

NOTE Confidence: 0.823594342857143

00:07:32.823 --> 00:07:35.357 makes the antenna sizes smaller.

NOTE Confidence: 0.823594342857143

00:07:35.360 --> 00:07:38.156 Another concept very familiar is filtering.

NOTE Confidence: 0.823594342857143

00:07:38.160 --> 00:07:40.914 Let's say you have some data in the air

NOTE Confidence: 0.823594342857143

00:07:40.914 --> 00:07:43.328 and you want to denoise the data because

NOTE Confidence: 0.823594342857143

00:07:43.328 --> 00:07:45.920 the data when you send from an antenna,

NOTE Confidence: 0.823594342857143

00:07:45.920 --> 00:07:49.718 it gets very noisy from very

NOTE Confidence: 0.823594342857143

00:07:49.720 --> 00:07:52.160 from many other signals around.

NOTE Confidence: 0.823594342857143

00:07:52.160 --> 00:07:54.160 And with the process the

NOTE Confidence: 0.823594342857143

00:07:54.160 --> 00:07:55.360 concept is filtering.

NOTE Confidence: 0.823594342857143

00:07:55.360 --> 00:07:57.868 You want to remove the noise or

NOTE Confidence: 0.823594342857143

00:07:57.868 --> 00:08:00.436 or kind of smoothen the data,

NOTE Confidence: 0.823594342857143

00:08:00.440 --> 00:08:03.476 then you need this filtering operation.

NOTE Confidence: 0.823594342857143

00:08:03.480 --> 00:08:05.754 So these are basic operations that

NOTE Confidence: 0.823594342857143

00:08:05.754 --> 00:08:08.640 we do in classical signal processing.

NOTE Confidence: 0.823594342857143

00:08:08.640 --> 00:08:11.636 But what if the data that we

NOTE Confidence: 0.823594342857143

00:08:11.636 --> 00:08:14.330 are dealing with is very lying

NOTE Confidence: 0.823594342857143

00:08:14.330 --> 00:08:16.680 on a very irregular setting?

NOTE Confidence: 0.823594342857143

00:08:16.680 --> 00:08:17.811 Like for example,

NOTE Confidence: 0.823594342857143

00:08:17.811 --> 00:08:20.073 here consider a graph or a

NOTE Confidence: 0.823936955882353

00:08:22.280 --> 00:08:23.972 where the each node,

NOTE Confidence: 0.823936955882353

00:08:23.972 --> 00:08:27.207 so each blue circle here we can we

NOTE Confidence: 0.823936955882353

00:08:27.207 --> 00:08:30.312 call it as a node and each red link

NOTE Confidence: 0.823936955882353

00:08:30.312 --> 00:08:34.638 here is we call it as an edge or a link.

NOTE Confidence: 0.823936955882353

00:08:34.640 --> 00:08:38.090 And these edges are representing the

NOTE Confidence: 0.823936955882353

00:08:38.090 --> 00:08:40.999 relationship or the proximity or the

NOTE Confidence: 0.823936955882353

00:08:41.000 --> 00:08:43.862 connectedness of these two nodes or

NOTE Confidence: 0.823936955882353

00:08:43.862 --> 00:08:47.120 entities or these two spatial points.

NOTE Confidence: 0.823936955882353

00:08:47.120 --> 00:08:49.385 These can these nodes can

NOTE Confidence: 0.823936955882353

00:08:49.385 --> 00:08:51.415 be different brain regions.

NOTE Confidence: 0.823936955882353

00:08:51.415 --> 00:08:54.596 It can be a person in a social

NOTE Confidence: 0.823936955882353

00:08:54.596 --> 00:08:56.619 network and all we will see another

NOTE Confidence: 0.823936955882353

00:08:56.619 --> 00:08:58.677 some many examples of this later.

NOTE Confidence: 0.948727781111111

00:09:01.400 --> 00:09:04.073 Yeah. So for example here in a brain region,

NOTE Confidence: 0.583627093333333

00:09:06.320 --> 00:09:10.919 let's say you are recording F nears or fMRI,

NOTE Confidence: 0.583627093333333

00:09:10.920 --> 00:09:14.480 then what it does at the end is

NOTE Confidence: 0.583627093333333

00:09:14.480 --> 00:09:16.798 different brain regions have a

NOTE Confidence: 0.583627093333333

00:09:16.800 --> 00:09:19.520 time series associated with it.

NOTE Confidence: 0.583627093333333

00:09:19.520 --> 00:09:25.175 And today we are not going to take this

NOTE Confidence: 0.583627093333333

00:09:25.175 --> 00:09:28.160 time domain information into into account.

NOTE Confidence: 0.583627093333333

00:09:28.160 --> 00:09:30.896 What we are going to do is at one

NOTE Confidence: 0.583627093333333

00:09:30.896 --> 00:09:33.780 time point we have data collected

NOTE Confidence: 0.583627093333333

00:09:33.780 --> 00:09:37.196 from each of the region of the brain

NOTE Confidence: 0.583627093333333

00:09:37.196 --> 00:09:39.600 and we want to analyse that data,

NOTE Confidence: 0.583627093333333

00:09:39.600 --> 00:09:42.350 develop some tools existing in

NOTE Confidence: 0.583627093333333

00:09:42.350 --> 00:09:44.345 classical signal processing to

NOTE Confidence: 0.583627093333333

00:09:44.345 --> 00:09:45.380 this irregular structure.

NOTE Confidence: 0.583627093333333

00:09:45.380 --> 00:09:46.760 Because brain is like,

NOTE Confidence: 0.583627093333333

00:09:46.760 --> 00:09:48.476 it's not like a 2D grade.

NOTE Confidence: 0.583627093333333

00:09:48.480 --> 00:09:50.565 It's like in A33 dimensional

NOTE Confidence: 0.583627093333333

00:09:50.565 --> 00:09:52.824 space where there are different

NOTE Confidence: 0.583627093333333

00:09:52.824 --> 00:09:55.160 structural regions that talk,

NOTE Confidence: 0.583627093333333

00:09:55.160 --> 00:09:58.082 talk with each other or send

NOTE Confidence: 0.583627093333333

00:09:58.082 --> 00:10:01.080 information from one region to another.

NOTE Confidence: 0.583627093333333

00:10:01.080 --> 00:10:03.912 How do we capture that information

NOTE Confidence: 0.583627093333333

00:10:03.912 --> 00:10:07.833 in the data and kind of get

NOTE Confidence: 0.583627093333333

00:10:07.833 --> 00:10:09.119 better representation?

NOTE Confidence: 0.583627093333333

00:10:09.120 --> 00:10:10.800 Some other examples include

NOTE Confidence: 0.851249573333333

00:10:13.040 --> 00:10:14.636 some other examples of this graph.

NOTE Confidence: 0.851249573333333

00:10:14.640 --> 00:10:17.345 Signals are like some temperature

NOTE Confidence: 0.851249573333333

00:10:17.345 --> 00:10:19.509 or pressures recorded at

NOTE Confidence: 0.851249573333333

00:10:19.509 --> 00:10:21.960 different geographical locations.

NOTE Confidence: 0.851249573333333

00:10:21.960 --> 00:10:23.920 Or it can be a social network

NOTE Confidence: 0.851249573333333

00:10:23.920 --> 00:10:25.180 created by Facebook, Twitter,

NOTE Confidence: 0.851249573333333

00:10:25.180 --> 00:10:28.120 or it can be a transportation network.

NOTE Confidence: 0.851249573333333

00:10:28.120 --> 00:10:30.200 For example, here this is a map of,

NOTE Confidence: 0.851249573333333

00:10:30.200 --> 00:10:34.844 I believe Minnesota and and at

NOTE Confidence: 0.851249573333333

00:10:34.844 --> 00:10:38.455 each node it can be like a traffic

NOTE Confidence: 0.851249573333333

00:10:38.455 --> 00:10:43.040 junction where the signal can be the

NOTE Confidence: 0.851249573333333

00:10:43.040 --> 00:10:45.770 kind of how many cars are passing

NOTE Confidence: 0.851249573333333

00:10:45.770 --> 00:10:47.599 through that traffic and all.

NOTE Confidence: 0.851249573333333

00:10:47.600 --> 00:10:51.101 And it can be traffic at traffic at each

NOTE Confidence: 0.851249573333333

00:10:51.101 --> 00:10:53.719 computer node in a computer network.

NOTE Confidence: 0.851249573333333

00:10:53.720 --> 00:10:55.416 So whenever you can,

NOTE Confidence: 0.851249573333333

00:10:55.416 --> 00:10:57.536 we you see this connectivity

NOTE Confidence: 0.851249573333333

00:10:57.536 --> 00:10:58.799 information around you.

NOTE Confidence: 0.851249573333333

00:10:58.800 --> 00:11:01.430 There are features associated with

NOTE Confidence: 0.851249573333333

00:11:01.430 --> 00:11:03.920 each entity or each node here.

NOTE Confidence: 0.851249573333333

00:11:03.920 --> 00:11:05.999 And we want to process such data.

NOTE Confidence: 0.957965634

00:11:08.160 --> 00:11:10.040 So what are the difficulties

NOTE Confidence: 0.783810505714286

00:11:10.040 --> 00:11:13.192 that rise when we talk about this data

NOTE Confidence: 0.783810505714286

00:11:13.192 --> 00:11:16.160 lying on a very complex structure?

NOTE Confidence: 0.783810505714286

00:11:16.160 --> 00:11:18.840 So if it is A1 dimensional time series,

NOTE Confidence: 0.783810505714286

00:11:18.840 --> 00:11:20.358 you can easily shift the signal,

NOTE Confidence: 0.783810505714286

00:11:20.360 --> 00:11:21.329 translate the signal.

NOTE Confidence: 0.783810505714286

00:11:21.329 --> 00:11:23.590 But then when we talk about in

NOTE Confidence: 0.783810505714286

00:11:23.656 --> 00:11:25.296 graph signal processing settings

NOTE Confidence: 0.783810505714286

00:11:25.296 --> 00:11:27.756 where the data is very irregular,

NOTE Confidence: 0.783810505714286

00:11:27.760 --> 00:11:30.224 then how do we translate or how

NOTE Confidence: 0.783810505714286

00:11:30.224 --> 00:11:32.960 we move from one node to another?

NOTE Confidence: 0.783810505714286

00:11:32.960 --> 00:11:34.628 Because there are so many nodes

NOTE Confidence: 0.783810505714286

00:11:34.628 --> 00:11:36.062 around you that are connected

NOTE Confidence: 0.783810505714286

00:11:36.062 --> 00:11:37.875 to you and there is no kind,

NOTE Confidence: 0.783810505714286

00:11:37.880 --> 00:11:41.264 there is no concept of moving from one

NOTE Confidence: 0.783810505714286

00:11:41.264 --> 00:11:44.720 point to another in a systematic way.

NOTE Confidence: 0.783810505714286

00:11:44.720 --> 00:11:48.918 So how, how do we like what,

NOTE Confidence: 0.783810505714286

00:11:48.920 --> 00:11:51.475 what kind of tools that can help

NOTE Confidence: 0.783810505714286

00:11:51.475 --> 00:11:53.426 to analyze this data properly?

NOTE Confidence: 0.783810505714286

00:11:53.426 --> 00:11:55.756 In order to motivate that,

NOTE Confidence: 0.783810505714286

00:11:55.760 --> 00:11:58.672 let's look at any time series data

NOTE Confidence: 0.783810505714286

00:11:58.672 --> 00:12:02.140 1st and then we will relate this to

NOTE Confidence: 0.783810505714286

00:12:02.140 --> 00:12:03.916 graph signal processing settings.

NOTE Confidence: 0.783810505714286

00:12:03.920 --> 00:12:05.027 So for example,

NOTE Confidence: 0.783810505714286

00:12:05.027 --> 00:12:07.610 let's say you have this time series

NOTE Confidence: 0.783810505714286

00:12:07.691 --> 00:12:10.162 with you which is a combination of

NOTE Confidence: 0.783810505714286

00:12:10.162 --> 00:12:12.776 some some sinusoids or it can be

NOTE Confidence: 0.783810505714286

00:12:12.776 --> 00:12:15.080 some combination of many sinusoids.

NOTE Confidence: 0.783810505714286

00:12:15.080 --> 00:12:18.480 So here this X is a combination of

NOTE Confidence: 0.783810505714286

00:12:18.480 --> 00:12:21.302 these three sinusoids with amplitude

NOTE Confidence: 0.783810505714286

00:12:21.302 --> 00:12:24.794 B1B2 and B3 and different frequencies

NOTE Confidence: 0.783810505714286

00:12:24.794 --> 00:12:27.788 Omega 1, Omega two and omega-3.

NOTE Confidence: 0.783810505714286

00:12:27.788 --> 00:12:29.159 So these frequencies.

NOTE Confidence: 0.783810505714286

00:12:29.160 --> 00:12:31.038 So if the frequency is higher,

NOTE Confidence: 0.783810505714286

00:12:31.040 --> 00:12:33.596 that means your signal is changing

NOTE Confidence: 0.783810505714286

00:12:33.596 --> 00:12:34.874 very rapidly rapidly.

NOTE Confidence: 0.783810505714286

00:12:34.880 --> 00:12:37.232 So this omega-3 is higher than

NOTE Confidence: 0.783810505714286

00:12:37.232 --> 00:12:39.200 Omega two and Omega one.

NOTE Confidence: 0.783810505714286

00:12:39.200 --> 00:12:41.876 So signal is changing very rapidly

NOTE Confidence: 0.783810505714286

00:12:41.880 --> 00:12:45.633 and this Omega one is a small so the

NOTE Confidence: 0.783810505714286

00:12:45.633 --> 00:12:49.640 signal is very smooth changing very slowly.

NOTE Confidence: 0.783810505714286

00:12:49.640 --> 00:12:54.596 So in order to represent this data

NOTE Confidence: 0.783810505714286

00:12:54.600 --> 00:12:57.302 either you can save this whole time

NOTE Confidence: 0.783810505714286

00:12:57.302 --> 00:13:00.104 series and process this process and do

NOTE Confidence: 0.783810505714286

00:13:00.104 --> 00:13:02.462 your processing on this time series

NOTE Confidence: 0.783810505714286

00:13:02.535 --> 00:13:05.118 directly or what you can do is you

NOTE Confidence: 0.783810505714286

00:13:05.118 --> 00:13:08.184 can convert this data in a different

NOTE Confidence: 0.783810505714286

00:13:08.184 --> 00:13:11.013 domain called frequency domain where

NOTE Confidence: 0.783810505714286

00:13:11.013 --> 00:13:13.878 this Fourier transform comes handy.

NOTE Confidence: 0.783810505714286

00:13:13.880 --> 00:13:16.436 So Fourier transform is nothing but

NOTE Confidence: 0.783810505714286

00:13:16.440 --> 00:13:21.052 decomposing any time series in terms

NOTE Confidence: 0.783810505714286

00:13:21.052 --> 00:13:23.556 of different frequency sinusoids.

NOTE Confidence: 0.783810505714286

00:13:23.560 --> 00:13:26.514 So here if you just take the

NOTE Confidence: 0.783810505714286

00:13:26.514 --> 00:13:29.078 Fourier transform of the signal XT,

NOTE Confidence: 0.783810505714286

00:13:29.080 --> 00:13:31.468 you will see that there are

NOTE Confidence: 0.783810505714286

00:13:31.468 --> 00:13:33.060 only three frequency components

NOTE Confidence: 0.783810505714286

00:13:33.128 --> 00:13:35.055 present with different amplitude

NOTE Confidence: 0.783810505714286

00:13:35.055 --> 00:13:37.480 or different weights Omega 1,

NOTE Confidence: 0.783810505714286

00:13:37.480 --> 00:13:39.688 Omega two and omega-3.

NOTE Confidence: 0.783810505714286

00:13:39.688 --> 00:13:43.000 So to represent this signal XT

NOTE Confidence: 0.783810505714286

00:13:43.000 --> 00:13:45.926 the same amount of the the same

NOTE Confidence: 0.783810505714286

00:13:45.926 --> 00:13:47.862 information can be represented

NOTE Confidence: 0.783810505714286

00:13:47.862 --> 00:13:50.832 in this different domain called

NOTE Confidence: 0.783810505714286

00:13:50.832 --> 00:13:53.880 Fourier domain or frequency domain.

NOTE Confidence: 0.783810505714286

00:13:53.880 --> 00:13:57.280 And here in frequency domain,

NOTE Confidence: 0.783810505714286

00:13:57.280 --> 00:13:58.476 as you can see,

NOTE Confidence: 0.783810505714286

00:13:58.476 --> 00:14:00.659 you just need three need to save

NOTE Confidence: 0.783810505714286

00:14:00.659 --> 00:14:02.972 only Omega 1, Omega two,

NOTE Confidence: 0.783810505714286

00:14:02.972 --> 00:14:05.716 omega-3 and corresponding amplitudes.

NOTE Confidence: 0.783810505714286

00:14:05.720 --> 00:14:08.160 But in order to save this XT in a machine,

NOTE Confidence: 0.783810505714286

00:14:08.160 --> 00:14:10.200 you have to save a lot of values

NOTE Confidence: 0.783810505714286

00:14:10.200 --> 00:14:12.494 and it doesn't provide that very

NOTE Confidence: 0.783810505714286

00:14:12.494 --> 00:14:14.639 nice frequency intuition as well.

NOTE Confidence: 0.783810505714286

00:14:14.640 --> 00:14:17.325 And converting this data to

NOTE Confidence: 0.783810505714286

00:14:17.325 --> 00:14:19.924 another domain has some advantages.

NOTE Confidence: 0.783810505714286

00:14:19.924 --> 00:14:20.368 It,

NOTE Confidence: 0.783810505714286

00:14:20.368 --> 00:14:23.032 it is used in filtering operations

NOTE Confidence: 0.783810505714286

00:14:23.032 --> 00:14:26.417 and it and the information and also

NOTE Confidence: 0.783810505714286

00:14:26.417 --> 00:14:28.832 all the communication systems are

NOTE Confidence: 0.783810505714286

00:14:28.913 --> 00:14:31.718 built around this frequency concept.

NOTE Confidence: 0.783810505714286

00:14:31.720 --> 00:14:34.952 So we want to extend the same kind

NOTE Confidence: 0.783810505714286

00:14:34.952 --> 00:14:37.092 of frequency interpretation of this

NOTE Confidence: 0.783810505714286

00:14:37.092 --> 00:14:39.756 time series data to graph signals.

NOTE Confidence: 0.969247926666667

00:14:43.600 --> 00:14:47.118 So how do we do that? So first, let's

NOTE Confidence: 0.905529261818182

00:14:49.960 --> 00:14:51.170 talk about some notations here

NOTE Confidence: 0.905529261818182

00:14:51.170 --> 00:14:52.720 that I'm going to be using.

NOTE Confidence: 0.967458595

00:14:55.840 --> 00:14:57.590 And before that, if you have any

NOTE Confidence: 0.967458595

00:14:57.590 --> 00:15:00.840 questions, feel free to stop me.

NOTE Confidence: 0.967458595

00:15:00.840 --> 00:15:02.772 And Helen also, if you will see

NOTE Confidence: 0.967458595

00:15:02.772 --> 00:15:04.680 something in chat, please let me know.

NOTE Confidence: 0.826644568333333

00:15:06.240 --> 00:15:08.360 Yes, I'm, I'm watching the questions. Yes.

NOTE Confidence: 0.799729945

00:15:11.040 --> 00:15:13.280 So notation. So we have a graph here.

NOTE Confidence: 0.799729945

00:15:13.280 --> 00:15:15.158 We have like 5 node graph

NOTE Confidence: 0.94296246875

00:15:17.520 --> 00:15:20.320 12345. It can be five

NOTE Confidence: 0.94296246875

00:15:20.320 --> 00:15:22.000 different brain regions.

NOTE Confidence: 0.94296246875

00:15:22.000 --> 00:15:25.348 It can be five different geographical

NOTE Confidence: 0.94296246875

00:15:25.348 --> 00:15:28.440 locations in a sensor network.

NOTE Confidence: 0.94296246875

00:15:28.440 --> 00:15:31.095 It can be in a tissue if you if

NOTE Confidence: 0.94296246875

00:15:31.095 --> 00:15:33.836 you can image the cells and all,

NOTE Confidence: 0.94296246875

00:15:33.840 --> 00:15:37.152 it can be each cell and this red,

NOTE Confidence: 0.94296246875

00:15:37.152 --> 00:15:40.680 red lines here are edges or links.

NOTE Confidence: 0.94296246875

00:15:40.680 --> 00:15:42.516 So four is connected to five,

NOTE Confidence: 0.94296246875

00:15:42.520 --> 00:15:45.274 which can be for example in a brain network,

NOTE Confidence: 0.94296246875

00:15:45.280 --> 00:15:50.152 let's say your frontal region connected

NOTE Confidence: 0.94296246875

00:15:50.152 --> 00:15:53.000 to the this temporal parietal

NOTE Confidence: 0.94296246875

00:15:53.000 --> 00:15:55.520 junction or any other you can imagine,

NOTE Confidence: 0.94296246875

00:15:55.520 --> 00:15:57.842 whichever is like in proximity or

NOTE Confidence: 0.94296246875

00:15:57.842 --> 00:16:00.440 there is any other relationship,

NOTE Confidence: 0.94296246875

00:16:00.440 --> 00:16:01.608 we find an edge.

NOTE Confidence: 0.94296246875

00:16:01.608 --> 00:16:03.880 So in order to create a graph,

NOTE Confidence: 0.94296246875

00:16:03.880 --> 00:16:06.148 you can always kind of take

NOTE Confidence: 0.94296246875

00:16:06.148 --> 00:16:08.133 the proximity into account and

NOTE Confidence: 0.94296246875

00:16:08.133 --> 00:16:10.118 create and create these edges.

NOTE Confidence: 0.94296246875

00:16:10.120 --> 00:16:12.838 And then each edge also has

NOTE Confidence: 0.94296246875

00:16:12.838 --> 00:16:15.000 a feature associated with it.

NOTE Confidence: 0.94296246875

00:16:15.000 --> 00:16:18.798 For example, at time point T1,

NOTE Confidence: 0.94296246875

00:16:18.800 --> 00:16:21.398 each of the brain regions will

NOTE Confidence: 0.94296246875

00:16:21.398 --> 00:16:23.846 have some value associated with it

NOTE Confidence: 0.94296246875

00:16:23.846 --> 00:16:26.398 recorded by EGF nears or or F MRI.

NOTE Confidence: 0.94296246875

00:16:26.400 --> 00:16:27.702 It doesn't matter.

NOTE Confidence: 0.94296246875

00:16:27.702 --> 00:16:30.740 And we are just looking at 1:00

NOTE Confidence: 0.94296246875

00:16:30.834 --> 00:16:33.894 time slice of this whole brain

NOTE Confidence: 0.94296246875

00:16:33.894 --> 00:16:36.144 region recording and we are calling

NOTE Confidence: 0.94296246875

00:16:36.144 --> 00:16:37.674 it as a graph signal.

NOTE Confidence: 0.94296246875

00:16:37.680 --> 00:16:40.270 And in order to store this graph

NOTE Confidence: 0.94296246875

00:16:40.270 --> 00:16:42.280 signal information in the machine,

NOTE Confidence: 0.94296246875

00:16:42.280 --> 00:16:45.196 we need to store the graph,

NOTE Confidence: 0.94296246875

00:16:45.200 --> 00:16:48.910 the connectivity and we need to store

NOTE Confidence: 0.94296246875

00:16:48.910 --> 00:16:52.956 this signal values associated with each node.

NOTE Confidence: 0.94296246875

00:16:52.960 --> 00:16:56.280 So when we talk when we talk about

NOTE Confidence: 0.94296246875

00:16:56.280 --> 00:16:58.236 storing this graph in the machine,

NOTE Confidence: 0.94296246875

00:16:58.240 --> 00:17:00.998 we can just store this weight matrix.

NOTE Confidence: 0.94296246875

00:17:01.000 --> 00:17:02.920 What is this weight matrix?

NOTE Confidence: 0.94296246875

00:17:02.920 --> 00:17:07.715 Weight matrix is a is a matrix that is

NOTE Confidence: 0.94296246875

00:17:07.715 --> 00:17:09.870 telling you the relationship between

NOTE Confidence: 0.94296246875

00:17:09.950 --> 00:17:12.800 different nodes or different regions.

NOTE Confidence: 0.94296246875

00:17:12.800 --> 00:17:15.304 So for example here if you see node

NOTE Confidence: 0.94296246875

00:17:15.304 --> 00:17:17.797 two is connected to node three,

NOTE Confidence: 0.94296246875

00:17:17.800 --> 00:17:20.520 node one and node 5.

NOTE Confidence: 0.94296246875

00:17:20.520 --> 00:17:23.208 So if you look at the 2nd row

NOTE Confidence: 0.94296246875

00:17:23.208 --> 00:17:25.424 of this weight matrix 2.

NOTE Confidence: 0.94296246875

00:17:25.424 --> 00:17:27.536 If you look at the 2nd row and

NOTE Confidence: 0.94296246875

00:17:27.536 --> 00:17:30.192 the first column, it is one.

NOTE Confidence: 0.94296246875

00:17:30.192 --> 00:17:32.416 That means this is representing

NOTE Confidence: 0.94296246875

00:17:32.416 --> 00:17:35.440 connection between node two and node one.

NOTE Confidence: 0.94296246875

00:17:35.440 --> 00:17:37.325 Then this diagonal entry is

NOTE Confidence: 0.94296246875

00:17:37.325 --> 00:17:38.833 0 because there is.

NOTE Confidence: 0.94296246875

00:17:38.840 --> 00:17:41.477 This is like a self loop two to two.

NOTE Confidence: 0.94296246875

00:17:41.480 --> 00:17:42.892 There is nothing there.

NOTE Confidence: 0.94296246875

00:17:42.892 --> 00:17:45.479 Then the 2nd row and 3rd column

NOTE Confidence: 0.94296246875

00:17:45.479 --> 00:17:46.640 has entry one.

NOTE Confidence: 0.94296246875

00:17:46.640 --> 00:17:50.996 That means node two is connected to node 3,

NOTE Confidence: 0.94296246875

00:17:51.000 --> 00:17:52.600 then fourth entry as well

NOTE Confidence: 0.94296246875

00:17:52.600 --> 00:17:54.200 as fifth entry are one.

NOTE Confidence: 0.94296246875

00:17:54.200 --> 00:17:56.138 That means two is also connected

NOTE Confidence: 0.94296246875

00:17:56.138 --> 00:17:58.039 to four as well as five.

NOTE Confidence: 0.94296246875

00:17:58.040 --> 00:18:00.120 So this connectivity information

NOTE Confidence: 0.94296246875

00:18:00.120 --> 00:18:03.840 you can store in this weight matrix

NOTE Confidence: 0.94296246875

00:18:03.840 --> 00:18:07.040 and then the features associated

NOTE Confidence: 0.94296246875

00:18:07.040 --> 00:18:10.977 with each node we can store

NOTE Confidence: 0.94296246875

00:18:10.977 --> 00:18:14.637 using this vector or signal F.

NOTE Confidence: 0.94296246875

00:18:14.640 --> 00:18:16.474 And then based on this weight matrix,

NOTE Confidence: 0.94296246875

00:18:16.480 --> 00:18:19.420 there are some other matrices defined like

NOTE Confidence: 0.94296246875

00:18:19.420 --> 00:18:22.199 some degree matrix and Laplacian matrix.

NOTE Confidence: 0.94296246875

00:18:22.200 --> 00:18:24.240 What is this degree matrix?

NOTE Confidence: 0.94296246875

00:18:24.240 --> 00:18:26.270 So degree matrix is the

NOTE Confidence: 0.94296246875

00:18:26.270 --> 00:18:28.800 strength of edges in each node.

NOTE Confidence: 0.94296246875

00:18:28.800 --> 00:18:31.440 So for example in node three,

NOTE Confidence: 0.94296246875

00:18:31.440 --> 00:18:34.560 it is connected to three neighbors.

NOTE Confidence: 0.94296246875

00:18:34.560 --> 00:18:37.038 So if if you look at the

NOTE Confidence: 0.94296246875

00:18:37.040 --> 00:18:39.680 third entry here it is 3.

NOTE Confidence: 0.94296246875

00:18:39.680 --> 00:18:43.040 If you look at node 5 then it

NOTE Confidence: 0.94296246875

00:18:43.040 --> 00:18:44.840 is connected to two of the

NOTE Confidence: 0.94296246875

00:18:44.840 --> 00:18:47.876 nodes node two and node four.

NOTE Confidence: 0.94296246875

00:18:47.880 --> 00:18:50.834 So the 5th entry here is 2.

NOTE Confidence: 0.94296246875

00:18:50.840 --> 00:18:52.639 So it is saying you how many.

NOTE Confidence: 0.806551016666667

00:18:52.640 --> 00:18:54.656 It is telling you how many edges

NOTE Confidence: 0.806551016666667

00:18:54.656 --> 00:18:57.104 are connect, are in are incident

NOTE Confidence: 0.806551016666667

00:18:57.104 --> 00:18:59.434 incidenting on a particular node.

NOTE Confidence: 0.806551016666667

00:18:59.440 --> 00:19:01.180 It's a diagonal matrix.

NOTE Confidence: 0.806551016666667

00:19:01.180 --> 00:19:04.280 And then there is another matrix

NOTE Confidence: 0.806551016666667

00:19:04.280 --> 00:19:08.056 called Laplacian matrix which is

NOTE Confidence: 0.806551016666667

00:19:08.056 --> 00:19:11.040 nothing but the difference between the

NOTE Confidence: 0.806551016666667

00:19:11.040 --> 00:19:13.680 degree matrix and the weight matrix.

NOTE Confidence: 0.857333421818182

00:19:16.480 --> 00:19:19.189 So this Laplacian is a very important

NOTE Confidence: 0.857333421818182

00:19:19.189 --> 00:19:21.896 matrix for us because as we'll

NOTE Confidence: 0.857333421818182

00:19:21.896 --> 00:19:24.704 further see that this Laplacian of

NOTE Confidence: 0.857333421818182

00:19:24.704 --> 00:19:28.513 this graph is going to tell us a lot

NOTE Confidence: 0.857333421818182

00:19:28.513 --> 00:19:31.324 about the variations in the signals of

NOTE Confidence: 0.857333421818182

00:19:31.324 --> 00:19:34.838 variations in the data across the space.

NOTE Confidence: 0.857333421818182

00:19:34.840 --> 00:19:37.400 So we are representing graph as G and

NOTE Confidence: 0.857333421818182

00:19:37.400 --> 00:19:40.518 this V is nothing but a set of vertices.

NOTE Confidence: 0.857333421818182

00:19:40.520 --> 00:19:44.876 So we have like 5 vertices or nodes here

NOTE Confidence: 0.857333421818182

00:19:44.880 --> 00:19:47.995 12345 here West is the weight matrix.

NOTE Confidence: 0.857333421818182

00:19:48.000 --> 00:19:50.590 So let's go further and look at

NOTE Confidence: 0.857333421818182

00:19:50.590 --> 00:19:55.680 the graph Laplacian again closely.

NOTE Confidence: 0.857333421818182

00:19:55.680 --> 00:19:58.440 So we have this particular graph here.

NOTE Confidence: 0.857333421818182

00:19:58.440 --> 00:20:00.360 This is our graph Laplacian.

NOTE Confidence: 0.857333421818182

00:20:00.360 --> 00:20:01.752 As you can see,

NOTE Confidence: 0.857333421818182

00:20:01.752 --> 00:20:04.523 it is symmetric because there is no

NOTE Confidence: 0.857333421818182

00:20:04.523 --> 00:20:06.959 directionality information or anything.

NOTE Confidence: 0.857333421818182

00:20:06.960 --> 00:20:09.984 It is a symmetric matrix and if you look

NOTE Confidence: 0.857333421818182

00:20:09.984 --> 00:20:13.000 at the off diagonal entries which are

NOTE Confidence: 0.893278405384615

00:20:15.720 --> 00:20:18.024 sorry the diagonal entries which are

NOTE Confidence: 0.893278405384615

00:20:18.024 --> 00:20:20.919 nothing but the D the degree values.

NOTE Confidence: 0.893278405384615

00:20:20.920 --> 00:20:23.480 And if you look at the off diagonal

NOTE Confidence: 0.893278405384615

00:20:23.480 --> 00:20:26.685 entries that are some non positive

NOTE Confidence: 0.893278405384615

00:20:26.685 --> 00:20:30.115 or negative values there because it

NOTE Confidence: 0.893278405384615

00:20:30.115 --> 00:20:33.195 is coming from this D minus West.

NOTE Confidence: 0.893278405384615

00:20:33.200 --> 00:20:37.016 And also if you see the rows are

NOTE Confidence: 0.893278405384615

00:20:37.016 --> 00:20:41.360 summing to zero 2 -, 1 -, 1 is 0.

NOTE Confidence: 0.893278405384615

00:20:41.360 --> 00:20:44.804 Similarly any other row and it's also

NOTE Confidence: 0.893278405384615

00:20:44.804 --> 00:20:47.679 positive positive semi definite matrix.

NOTE Confidence: 0.893278405384615

00:20:47.680 --> 00:20:51.328 So if you are not familiar with semi

NOTE Confidence: 0.893278405384615

00:20:51.328 --> 00:20:54.440 definite so it is a matrix whose

NOTE Confidence: 0.893278405384615

00:20:54.440 --> 00:20:57.560 eigen values are always non negative.

NOTE Confidence: 0.893278405384615

00:20:57.560 --> 00:20:59.822 We will talk about this more

NOTE Confidence: 0.893278405384615

00:20:59.822 --> 00:21:00.953 and later slides

NOTE Confidence: 0.80179316

00:21:03.640 --> 00:21:08.440 then what this Laplacian gives us.

NOTE Confidence: 0.80179316

00:21:08.440 --> 00:21:11.433 Let's oh, before that,

NOTE Confidence: 0.80179316

00:21:11.433 --> 00:21:13.470 let's talk a little bit about what

NOTE Confidence: 0.80179316

00:21:13.528 --> 00:21:15.098 is eigen values and eigenvectors

NOTE Confidence: 0.80179316

00:21:15.098 --> 00:21:18.878 of a matrix R So let's say you have

NOTE Confidence: 0.80179316

00:21:18.878 --> 00:21:21.248 a matrix like just an N cross.

NOTE Confidence: 0.80179316

00:21:21.248 --> 00:21:22.431 And for example, here,

NOTE Confidence: 0.80179316

00:21:22.431 --> 00:21:24.599 this is just a matrix of four entries

NOTE Confidence: 0.97663206

00:21:27.080 --> 00:21:32.160 and the eigenvectors are those vectors.

NOTE Confidence: 0.97663206

00:21:32.160 --> 00:21:34.758 If you operate this matrix on,

NOTE Confidence: 0.97663206

00:21:34.760 --> 00:21:37.713 the direction is not going to change either.

NOTE Confidence: 0.97663206

00:21:37.713 --> 00:21:40.078 It's going to be scaled,

NOTE Confidence: 0.97663206

00:21:40.080 --> 00:21:41.800 but the direction is not going to change.

NOTE Confidence: 0.97663206

00:21:41.800 --> 00:21:46.040 For example, here if you see so AU,

NOTE Confidence: 0.97663206

00:21:46.040 --> 00:21:50.240 if U is an eigenvector of a matrix A,

NOTE Confidence: 0.97663206

00:21:50.240 --> 00:21:53.516 then it is just scaled by a factor Lambda,

NOTE Confidence: 0.97663206

00:21:53.520 --> 00:21:58.595 which is a scalar value called eigenvalue.

NOTE Confidence: 0.97663206

00:21:58.600 --> 00:22:00.478 So in this first figure here,

NOTE Confidence: 0.97663206

00:22:00.480 --> 00:22:04.480 if you see the vector 10,

NOTE Confidence: 0.97663206

00:22:04.480 --> 00:22:06.980 if you just multiply matrix

NOTE Confidence: 0.97663206

00:22:06.980 --> 00:22:09.064 A and this vector 10,

NOTE Confidence: 0.97663206

00:22:09.064 --> 00:22:13.013 you can see that you can write it as

NOTE Confidence: 0.97663206

00:22:13.013 --> 00:22:15.959 scaled version of this vector itself.

NOTE Confidence: 0.97663206

00:22:15.960 --> 00:22:18.919 This is the eigen value here 2.

NOTE Confidence: 0.97663206

00:22:18.919 --> 00:22:21.151 So it is it is just scaled from

NOTE Confidence: 0.97663206

00:22:21.151 --> 00:22:23.240 1:00 to 2:00 here and similarly

NOTE Confidence: 0.97663206

00:22:23.240 --> 00:22:27.480 another eigen value which is -1 here.

NOTE Confidence: 0.97663206

00:22:27.480 --> 00:22:32.080 So if you look at this vector one and -3,

NOTE Confidence: 0.97663206

00:22:32.080 --> 00:22:36.916 then if you just multiply A to this vector,

NOTE Confidence: 0.97663206

00:22:36.916 --> 00:22:38.710 it is going to be stretching

NOTE Confidence: 0.97663206

00:22:38.782 --> 00:22:40.278 in the other direction,

NOTE Confidence: 0.97663206

00:22:40.280 --> 00:22:42.640 but the direction is not going to change.

NOTE Confidence: 0.64558012

00:22:44.960 --> 00:22:48.398 Then given a matrix squared matrix,

NOTE Confidence: 0.64558012

00:22:48.400 --> 00:22:52.240 of course there are we are we are

NOTE Confidence: 0.64558012

00:22:52.240 --> 00:22:53.600 introducing these two quantities,

NOTE Confidence: 0.64558012

00:22:53.600 --> 00:22:56.640 eigen values and eigen vectors.

NOTE Confidence: 0.64558012

00:22:56.640 --> 00:22:59.040 So eigen values are scalars.

NOTE Confidence: 0.64558012

00:22:59.040 --> 00:23:02.450 So if A is a N cross N matrix then

NOTE Confidence: 0.64558012

00:23:02.553 --> 00:23:06.060 you will have N number of eigenvalues

NOTE Confidence: 0.64558012

00:23:06.060 --> 00:23:10.079 and N number of eigenvectors as well.

NOTE Confidence: 0.64558012

00:23:10.080 --> 00:23:13.820 So let's do Then what is the use of this

NOTE Confidence: 0.64558012

00:23:13.920 --> 00:23:18.036 eigenvalues and eigenvectors in our context?

NOTE Confidence: 0.64558012

00:23:18.040 --> 00:23:21.211 In our context, when we want to

NOTE Confidence: 0.64558012

00:23:21.211 --> 00:23:24.277 analyze the data lying on a graph,

NOTE Confidence: 0.64558012

00:23:24.280 --> 00:23:26.915 these eigenvalues and eigenvectors provide

NOTE Confidence: 0.64558012

00:23:26.915 --> 00:23:30.303 us the intuition of frequency in graph

NOTE Confidence: 0.64558012

00:23:30.303 --> 00:23:32.999 domain as we will see in next slides.

NOTE Confidence: 0.64558012

00:23:33.000 --> 00:23:38.990 So given a graph we use a term called

NOTE Confidence: 0.64558012

00:23:38.990 --> 00:23:41.478 spectrum or sometimes frequency.

NOTE Confidence: 0.64558012

00:23:41.480 --> 00:23:44.240 So eigenvalues of the Laplacian matrix,

NOTE Confidence: 0.64558012

00:23:44.240 --> 00:23:49.016 we call it as graph spectrum and eigen.

NOTE Confidence: 0.64558012

00:23:49.016 --> 00:23:51.953 And these eigenvectors that

NOTE Confidence: 0.64558012

00:23:51.953 --> 00:23:54.518 we just arrange in columns.

NOTE Confidence: 0.64558012

00:23:54.520 --> 00:23:56.438 So we have these four nodes here.

NOTE Confidence: 0.95369165

00:24:02.560 --> 00:24:06.515 Actually there is a little mistake here.

NOTE Confidence: 0.95369165

00:24:06.520 --> 00:24:07.498 This particular graph,

NOTE Confidence: 0.95369165

00:24:07.498 --> 00:24:09.128 this Laplacian is not corresponding

NOTE Confidence: 0.95369165

00:24:09.128 --> 00:24:10.680 to this particular graph.

NOTE Confidence: 0.95369165

00:24:10.680 --> 00:24:14.397 But I'll, I'll, I'll clarify it later.

NOTE Confidence: 0.95369165

00:24:14.400 --> 00:24:19.480 So using these eigenvalues and eigenvectors,

NOTE Confidence: 0.95369165

00:24:19.480 --> 00:24:22.035 how do we define this Fourier transform?

NOTE Confidence: 0.95369165

00:24:22.040 --> 00:24:23.768 So if you look at the

NOTE Confidence: 0.95369165

00:24:23.768 --> 00:24:24.632 classical Fourier transform,

NOTE Confidence: 0.95369165

00:24:24.640 --> 00:24:26.878 we have a time series XT.

NOTE Confidence: 0.95369165

00:24:26.880 --> 00:24:29.008 If you if we want to define

NOTE Confidence: 0.95369165

00:24:29.008 --> 00:24:30.479 the Fourier transform of this,

NOTE Confidence: 0.95369165

00:24:30.480 --> 00:24:34.086 we just use this complex exponentials

NOTE Confidence: 0.95369165

00:24:34.086 --> 00:24:37.075 or sinusoids because you can

NOTE Confidence: 0.95369165

00:24:37.075 --> 00:24:38.910 always decompose this complex

NOTE Confidence: 0.95369165

00:24:38.910 --> 00:24:41.160 exponential into sines and cosines.

NOTE Confidence: 0.95369165

00:24:41.160 --> 00:24:43.638 And this is the frequency here Omega.

NOTE Confidence: 0.95369165

00:24:43.640 --> 00:24:46.594 So usually when you see in FM

NOTE Confidence: 0.95369165

00:24:46.600 --> 00:24:49.240 it's like megahertz or in you

NOTE Confidence: 0.95369165

00:24:49.240 --> 00:24:51.896 in your in your voice signal,

NOTE Confidence: 0.95369165

00:24:51.896 --> 00:24:57.095 it is in kilohertz and all and you

NOTE Confidence: 0.95369165

00:24:57.095 --> 00:24:59.345 are decomposing the signal XT in

NOTE Confidence: 0.95369165

00:24:59.345 --> 00:25:02.280 terms of this frequency coefficients.

NOTE Confidence: 0.95369165

00:25:02.280 --> 00:25:04.880 Similarly, in our graph settings,

NOTE Confidence: 0.95369165

00:25:04.880 --> 00:25:07.984 what we can do is we can decompose

NOTE Confidence: 0.95369165

00:25:07.984 --> 00:25:08.800 our signal,

NOTE Confidence: 0.95369165

00:25:08.800 --> 00:25:11.648 a graph signal which is lying on the

NOTE Confidence: 0.95369165

00:25:11.648 --> 00:25:14.170 vertices or the nodes of the graph

NOTE Confidence: 0.95369165

00:25:14.170 --> 00:25:17.840 using these eigenvectors of the Laplacian.

NOTE Confidence: 0.95369165

00:25:17.840 --> 00:25:20.240 So you just compute the eigenvectors

NOTE Confidence: 0.95369165

00:25:20.240 --> 00:25:24.895 of the Laplacian and then use them to

NOTE Confidence: 0.95369165

00:25:24.895 --> 00:25:27.479 compute this graph Fourier transform.

NOTE Confidence: 0.879324026153846

00:25:30.280 --> 00:25:33.808 So any given graph you have a feature

NOTE Confidence: 0.879324026153846

00:25:33.808 --> 00:25:36.359 associated with each vertex vertex.

NOTE Confidence: 0.879324026153846

00:25:36.360 --> 00:25:39.720 You apply graph Fourier transform on

NOTE Confidence: 0.879324026153846

00:25:39.720 --> 00:25:45.005 this and you get to represent the same

NOTE Confidence: 0.879324026153846

00:25:45.005 --> 00:25:47.620 information that is contained in the

NOTE Confidence: 0.879324026153846

00:25:47.620 --> 00:25:51.123 data lying on a graph in this graph

NOTE Confidence: 0.879324026153846

00:25:51.123 --> 00:25:53.728 frequency domain where this lambdas

NOTE Confidence: 0.879324026153846

00:25:53.728 --> 00:25:56.868 are nothing but the eigenvalues of

NOTE Confidence: 0.879324026153846

00:25:56.868 --> 00:25:59.196 the graph Laplacian that you can

NOTE Confidence: 0.879324026153846

00:25:59.200 --> 00:26:01.198 think it of as just frequencies.

NOTE Confidence: 0.879324026153846

00:26:01.200 --> 00:26:05.080 The the like the omegas in classical domain

NOTE Confidence: 0.876891659523809

00:26:08.560 --> 00:26:09.964 like cosines, whatever Omegas

NOTE Confidence: 0.876891659523809

00:26:09.964 --> 00:26:12.070 that is related that is the

NOTE Confidence: 0.876891659523809

00:26:12.134 --> 00:26:13.979 analogous to the graph frequencies

NOTE Confidence: 0.876891659523809

00:26:13.979 --> 00:26:16.200 that we are talking about here.

NOTE Confidence: 0.876891659523809

00:26:16.200 --> 00:26:19.164 And for each frequency there will

NOTE Confidence: 0.876891659523809

00:26:19.164 --> 00:26:22.722 be an associated number called graph

NOTE Confidence: 0.876891659523809

00:26:22.722 --> 00:26:24.960 Fourier transform coefficient.

NOTE Confidence: 0.876891659523809

00:26:24.960 --> 00:26:27.704 So it is saying that in this

NOTE Confidence: 0.876891659523809

00:26:27.704 --> 00:26:30.928 signal or in this data you have

NOTE Confidence: 0.876891659523809

00:26:30.928 --> 00:26:33.452 some content of frequency 1,

NOTE Confidence: 0.876891659523809

00:26:33.452 --> 00:26:36.128 Omega some or Lambda some content

NOTE Confidence: 0.876891659523809

00:26:36.128 --> 00:26:38.157 of frequency 2-3 and so on.

NOTE Confidence: 0.97116065

00:26:41.080 --> 00:26:43.160 And I'm going to skip

NOTE Confidence: 0.97116065

00:26:43.160 --> 00:26:44.531 this frequency ordering,

NOTE Confidence: 0.97116065

00:26:44.531 --> 00:26:47.730 but let let's look at a closer

NOTE Confidence: 0.97116065

00:26:47.814 --> 00:26:50.802 look into this graph in this

NOTE Confidence: 0.97116065

00:26:50.802 --> 00:26:52.794 eigenvectors of this Laplacian.

NOTE Confidence: 0.97116065

00:26:52.800 --> 00:26:54.235 So this is a six node graph.

NOTE Confidence: 0.97116065

00:26:54.240 --> 00:26:56.502 So I have plotted this six

NOTE Confidence: 0.97116065

00:26:56.502 --> 00:26:58.480 eigenvectors on top of this.

NOTE Confidence: 0.97116065

00:26:58.480 --> 00:27:01.000 So if you look at the very first eigenvector,

NOTE Confidence: 0.97116065

00:27:01.000 --> 00:27:03.800 use zero, which is corresponding

NOTE Confidence: 0.97116065

00:27:03.800 --> 00:27:06.040 to the lowest eigenvalue.

NOTE Confidence: 0.97116065

00:27:06.040 --> 00:27:09.099 So sometimes we call this eigenvectors as

NOTE Confidence: 0.97116065

00:27:09.099 --> 00:27:12.781 harmonics or like you can relate it to

NOTE Confidence: 0.97116065

00:27:12.781 --> 00:27:15.076 sinusoids in classical signal processing.

NOTE Confidence: 0.97116065

00:27:15.080 --> 00:27:18.379 So if you look at U0, there's no change.

NOTE Confidence: 0.97116065

00:27:18.379 --> 00:27:20.960 It's like a constant signal across the graph.

NOTE Confidence: 0.97116065

00:27:20.960 --> 00:27:24.720 So if you move from node one to three to six,

NOTE Confidence: 0.97116065

00:27:24.720 --> 00:27:25.755 there's no change.

NOTE Confidence: 0.97116065

00:27:25.755 --> 00:27:27.480 It's like a constant signal.

NOTE Confidence: 0.97116065

00:27:27.480 --> 00:27:31.920 There's no change at all if you look at U1.

NOTE Confidence: 0.97116065

00:27:31.920 --> 00:27:34.719 So if you move from node one to two,

NOTE Confidence: 0.97116065

00:27:34.720 --> 00:27:36.760 you can see it's like positive to negative.

NOTE Confidence: 0.97116065

00:27:36.760 --> 00:27:39.383 So you can see a little something

NOTE Confidence: 0.97116065

00:27:39.383 --> 00:27:41.441 like sine wave occurring here if

NOTE Confidence: 0.97116065

00:27:41.441 --> 00:27:43.758 you move from one node to another.

NOTE Confidence: 0.97116065

00:27:43.760 --> 00:27:47.120 So if you move from node 2 to one 2-3,

NOTE Confidence: 0.97116065

00:27:47.120 --> 00:27:49.466 you can see that it's going

NOTE Confidence: 0.97116065

00:27:49.466 --> 00:27:50.639 from negative values,

NOTE Confidence: 0.97116065

00:27:50.640 --> 00:27:52.440 it's going to positive values.

NOTE Confidence: 0.97116065

00:27:52.440 --> 00:27:54.240 So it's like change,

NOTE Confidence: 0.97116065

00:27:54.240 --> 00:27:56.613 it is the changing change

NOTE Confidence: 0.97116065

00:27:56.613 --> 00:27:59.478 across the space is increasing.

NOTE Confidence: 0.97116065

00:27:59.480 --> 00:28:03.160 And if you look at the eigen vector

NOTE Confidence: 0.97116065

00:28:03.160 --> 00:28:07.080 corresponding to the largest eigen value,

NOTE Confidence: 0.97116065

00:28:07.080 --> 00:28:09.438 the change is even like larger.

NOTE Confidence: 0.97116065

00:28:09.440 --> 00:28:11.560 So if you look at this node,

NOTE Confidence: 0.97116065

00:28:11.560 --> 00:28:13.536 two to five positive, negative,

NOTE Confidence: 0.97116065

00:28:13.536 --> 00:28:16.444 then five to four positive, negative.

NOTE Confidence: 0.97116065

00:28:16.444 --> 00:28:19.118 So the changes are very rapid here.

NOTE Confidence: 0.97116065

00:28:19.120 --> 00:28:23.160 So the if you look at from U0 to U5,

NOTE Confidence: 0.97116065

00:28:23.160 --> 00:28:25.323 the change in the signal or change

NOTE Confidence: 0.97116065

00:28:25.323 --> 00:28:27.240 in this data lying on this,

NOTE Confidence: 0.97116065

00:28:27.240 --> 00:28:30.092 each node is increasing.

NOTE Confidence: 0.97116065

00:28:30.092 --> 00:28:32.952 That means these are the harmonics.

NOTE Confidence: 0.97116065

00:28:32.952 --> 00:28:35.652 We can treat this as a harmonic

NOTE Confidence: 0.97116065

00:28:35.652 --> 00:28:37.080 like sine waves.

NOTE Confidence: 0.97116065

00:28:37.080 --> 00:28:38.840 So here there's no change,

NOTE Confidence: 0.97116065

00:28:38.840 --> 00:28:40.420 a little bit change,

NOTE Confidence: 0.97116065

00:28:40.420 --> 00:28:42.528 very smoothly varying signal towards

NOTE Confidence: 0.97116065

00:28:42.528 --> 00:28:44.640 very rapidly changing signal.

NOTE Confidence: 0.97116065

00:28:44.640 --> 00:28:47.844 And we want to represent our data in terms

NOTE Confidence: 0.97116065

00:28:47.844 --> 00:28:51.240 of these eigenvectors or graph harmonics.

NOTE Confidence: 0.9653577575

00:28:54.240 --> 00:28:56.760 Do we have any questions at this point? I

NOTE Confidence: 0.8672279

00:29:02.800 --> 00:29:06.180 just want to follow up and make

NOTE Confidence: 0.8672279

00:29:06.180 --> 00:29:07.760 sure they understand correctly.

NOTE Confidence: 0.8672279

00:29:07.760 --> 00:29:11.640 So essentially I can various

NOTE Confidence: 0.736592698333333

00:29:14.680 --> 00:29:19.756 just weights on certain ordinate functions.

NOTE Confidence: 0.736592698333333

00:29:19.760 --> 00:29:22.280 For instance, if we had the

NOTE Confidence: 0.736592698333333

00:29:22.280 --> 00:29:23.960 simple 3 dimensional space,

NOTE Confidence: 0.736592698333333

00:29:23.960 --> 00:29:25.884 right, linear, linear, linear,

NOTE Confidence: 0.736592698333333

00:29:25.884 --> 00:29:28.770 then we have coordinates in each

NOTE Confidence: 0.736592698333333

00:29:28.849 --> 00:29:31.537 space and we have a vector and we

NOTE Confidence: 0.736592698333333

00:29:31.537 --> 00:29:34.119 can define this vector much easier.

NOTE Confidence: 0.736592698333333

00:29:34.120 --> 00:29:36.626 And here it is pretty much the

NOTE Confidence: 0.736592698333333

00:29:36.626 --> 00:29:38.919 same but more complex basis.

NOTE Confidence: 0.736592698333333

00:29:38.920 --> 00:29:41.080 So you have this

NOTE Confidence: 0.943132876

00:29:44.040 --> 00:29:48.960 different types of changes and you

NOTE Confidence: 0.943132876

00:29:48.960 --> 00:29:51.376 effectively decompose your signal

NOTE Confidence: 0.943132876

00:29:51.376 --> 00:29:55.579 which can have any kind of shape

NOTE Confidence: 0.943132876

00:29:55.579 --> 00:29:59.775 and form in the weighted sum of

NOTE Confidence: 0.943132876

00:29:59.775 --> 00:30:04.040 this more simple type of signals.

NOTE Confidence: 0.84347158125

00:30:04.600 --> 00:30:07.715 Yes, exactly. And it is also providing

NOTE Confidence: 0.84347158125

00:30:07.715 --> 00:30:09.210 the intuition of variations,

NOTE Confidence: 0.84347158125

00:30:09.210 --> 00:30:10.960 how quickly it is changing,

NOTE Confidence: 0.84347158125

00:30:10.960 --> 00:30:13.396 how slowly it is changing accordingly.

NOTE Confidence: 0.847098691428571

00:30:13.920 --> 00:30:16.636 Yeah. So you started with this sinusoid,

NOTE Confidence: 0.847098691428571

00:30:16.640 --> 00:30:21.090 which was more complicated than

NOTE Confidence: 0.847098691428571

00:30:21.090 --> 00:30:24.920 just a simple axis, Jakarta's axis.

NOTE Confidence: 0.847098691428571

00:30:24.920 --> 00:30:27.685 And this is more complex than sinusoid

NOTE Confidence: 0.847098691428571

00:30:27.685 --> 00:30:30.120 because you have more complex, Yes.

NOTE Confidence: 0.724532579285714

00:30:30.360 --> 00:30:32.760 So in sinusoids we had just one dimension

NOTE Confidence: 0.724532579285714

00:30:32.760 --> 00:30:34.959 time it was changing across that.

NOTE Confidence: 0.724532579285714

00:30:34.960 --> 00:30:36.904 But here we have like very

NOTE Confidence: 0.724532579285714

00:30:36.904 --> 00:30:37.876 irregular dimension here.

NOTE Confidence: 0.724532579285714

00:30:37.880 --> 00:30:40.180 So it's like non Euclidean

NOTE Confidence: 0.724532579285714

00:30:40.180 --> 00:30:42.570 space and still the changes are

NOTE Confidence: 0.724532579285714

00:30:42.570 --> 00:30:44.114 according to the connectivity.

NOTE Confidence: 0.724532579285714

00:30:44.120 --> 00:30:45.877 So if we move from one node,

NOTE Confidence: 0.724532579285714

00:30:45.880 --> 00:30:47.290 the nearby nodes,

NOTE Confidence: 0.724532579285714

00:30:47.290 --> 00:30:49.640 how smoothly or how rapidly

NOTE Confidence: 0.724532579285714

00:30:49.640 --> 00:30:51.404 my information is varying,

NOTE Confidence: 0.724532579285714

00:30:51.404 --> 00:30:54.630 that is kind of provides a sense

NOTE Confidence: 0.724532579285714

00:30:54.630 --> 00:30:56.582 of frequency or variability

NOTE Confidence: 0.724532579285714

00:30:56.582 --> 00:30:58.720 in this irregular graph.

NOTE Confidence: 0.93347286

00:31:03.400 --> 00:31:06.280 OK, so let's move on.

NOTE Confidence: 0.93347286

00:31:06.280 --> 00:31:09.826 So another important thing to notice

NOTE Confidence: 0.93347286

00:31:09.826 --> 00:31:14.104 here is that we have numbered this,

NOTE Confidence: 0.93347286

00:31:14.104 --> 00:31:18.705 we have numbered this graph like the we

NOTE Confidence: 0.93347286

00:31:18.705 --> 00:31:21.400 have assigned #1 to this particular node.

NOTE Confidence: 0.93347286

00:31:21.400 --> 00:31:23.840 We have assigned #4 to this particular node.

NOTE Confidence: 0.93347286

00:31:23.840 --> 00:31:28.112 What what if we kind of scramble these

NOTE Confidence: 0.93347286

00:31:28.112 --> 00:31:31.519 numbers like we can like change node

NOTE Confidence: 0.93347286

00:31:31.519 --> 00:31:34.717 three to node one and accordingly

NOTE Confidence: 0.89352732

00:31:36.800 --> 00:31:39.638 the the structure is not changing,

NOTE Confidence: 0.89352732

00:31:39.640 --> 00:31:41.640 but the representation might change.

NOTE Confidence: 0.89352732

00:31:41.640 --> 00:31:44.678 So for example, let's look at this.

NOTE Confidence: 0.89352732

00:31:44.680 --> 00:31:47.392 What the effect of this vertex

NOTE Confidence: 0.89352732

00:31:47.392 --> 00:31:49.200 indexing or node indexing,

NOTE Confidence: 0.89352732

00:31:49.200 --> 00:31:52.050 how it changes the graph harmonics

NOTE Confidence: 0.89352732

00:31:52.050 --> 00:31:53.475 or signal representations.

NOTE Confidence: 0.89352732

00:31:53.480 --> 00:31:57.275 So we start with again a very simple,

NOTE Confidence: 0.89352732

00:31:57.280 --> 00:31:59.994 very small graph.

NOTE Confidence: 0.89352732

00:31:59.994 --> 00:32:02.279 We have this graph here,

NOTE Confidence: 0.89352732

00:32:02.280 --> 00:32:04.320 we have this weight matrix here.

NOTE Confidence: 0.89352732

00:32:04.320 --> 00:32:06.318 So here sometimes

NOTE Confidence: 0.840137585

00:32:08.480 --> 00:32:11.904 it is beneficial to assign a

NOTE Confidence: 0.840137585

00:32:11.904 --> 00:32:15.065 scalar value to each of the edges

NOTE Confidence: 0.840137585

00:32:15.065 --> 00:32:17.640 as well or each of the links.

NOTE Confidence: 0.840137585

00:32:17.640 --> 00:32:21.160 So when we assign a scalar value here,

NOTE Confidence: 0.840137585

00:32:21.160 --> 00:32:23.810 that means how well connected

NOTE Confidence: 0.840137585

00:32:23.810 --> 00:32:26.960 node two and node three are.

NOTE Confidence: 0.840137585

00:32:26.960 --> 00:32:29.214 So if you compare in this particular

NOTE Confidence: 0.840137585

00:32:29.214 --> 00:32:31.159 figure here node two and node 3,

NOTE Confidence: 0.840137585

00:32:31.160 --> 00:32:33.656 the connectivity strength is .2 as

NOTE Confidence: 0.840137585

00:32:33.656 --> 00:32:36.689 compared to node three and four where

NOTE Confidence: 0.840137585

00:32:36.689 --> 00:32:39.235 the connectivity strength can be .7.

NOTE Confidence: 0.840137585

00:32:39.235 --> 00:32:41.605 So this can be for example

NOTE Confidence: 0.840137585

00:32:41.605 --> 00:32:43.640 in your computer networks,

NOTE Confidence: 0.840137585

00:32:43.640 --> 00:32:46.454 sometimes it can reflect the bandwidth

NOTE Confidence: 0.840137585

00:32:46.454 --> 00:32:50.039 how much of data one link can carry.

NOTE Confidence: 0.840137585

00:32:50.040 --> 00:32:53.076 So you can assign weights accordingly.

NOTE Confidence: 0.861264512

00:32:55.760 --> 00:32:57.120 And in a social network,

NOTE Confidence: 0.861264512

00:32:57.120 --> 00:32:59.680 sometimes you can say, OK,

NOTE Confidence: 0.861264512

00:32:59.680 --> 00:33:01.696 this is my best friend versus

NOTE Confidence: 0.861264512

00:33:01.696 --> 00:33:03.040 friend versus just acquaintances.

NOTE Confidence: 0.861264512

00:33:03.040 --> 00:33:05.440 So they're the, even though you,

NOTE Confidence: 0.861264512

00:33:05.440 --> 00:33:07.336 you are friend with you are

NOTE Confidence: 0.861264512

00:33:07.336 --> 00:33:08.600 friends on social network,

NOTE Confidence: 0.861264512

00:33:08.600 --> 00:33:11.785 but it's still there are some kind

NOTE Confidence: 0.861264512

00:33:11.785 --> 00:33:15.415 of weights are associated with your

NOTE Confidence: 0.861264512

00:33:15.415 --> 00:33:18.284 relationship with your relationship with

NOTE Confidence: 0.861264512

00:33:18.284 --> 00:33:20.876 other people in the social network as well.

NOTE Confidence: 0.861264512

00:33:20.880 --> 00:33:23.355 So this weight matrix here

NOTE Confidence: 0.861264512

00:33:23.355 --> 00:33:25.788 instead of the 011 entries,

NOTE Confidence: 0.861264512

00:33:25.788 --> 00:33:28.158 there are weights associated here.

NOTE Confidence: 0.861264512

00:33:28.160 --> 00:33:29.700 So for example the 1st

NOTE Confidence: 0.861264512

00:33:29.700 --> 00:33:31.240 row and the second column,

NOTE Confidence: 0.861264512

00:33:31.240 --> 00:33:33.720 the entries .3 that means

NOTE Confidence: 0.861264512

00:33:33.720 --> 00:33:36.200 node one and node 2.

NOTE Confidence: 0.861264512

00:33:36.200 --> 00:33:38.640 The connection strength is .3.

NOTE Confidence: 0.861264512

00:33:38.640 --> 00:33:41.112 And accordingly you can fill in

NOTE Confidence: 0.861264512

00:33:41.112 --> 00:33:44.292 this matrix and then you can compute

NOTE Confidence: 0.861264512

00:33:44.292 --> 00:33:46.707 this Laplacian matrix using this

NOTE Confidence: 0.861264512

00:33:46.707 --> 00:33:49.420 degree minus W which turns out to be

NOTE Confidence: 0.861264512

00:33:49.420 --> 00:33:52.029 this and then you can compute these

NOTE Confidence: 0.861264512

00:33:52.029 --> 00:33:54.394 eigen values and eigen vectors.

NOTE Confidence: 0.861264512

00:33:54.400 --> 00:33:57.046 So let us look at a very nice property

NOTE Confidence: 0.861264512

00:33:57.046 --> 00:33:59.359 of this Laplacian matrix here.

NOTE Confidence: 0.861264512

00:33:59.360 --> 00:34:03.448 As you will see the first eigen value

NOTE Confidence: 0.861264512

00:34:03.448 --> 00:34:08.173 is always going to be 0 always and

NOTE Confidence: 0.861264512

00:34:08.173 --> 00:34:11.064 the other eigen values are going to

NOTE Confidence: 0.861264512

00:34:11.064 --> 00:34:13.600 be like non negative eigen values.

NOTE Confidence: 0.680669108

00:34:15.720 --> 00:34:18.240 That is why it is called positive

NOTE Confidence: 0.680669108

00:34:18.240 --> 00:34:20.712 semi definite matrix and why this

NOTE Confidence: 0.680669108

00:34:20.712 --> 00:34:23.032 first eigenvalue is 0 because

NOTE Confidence: 0.680669108

00:34:23.032 --> 00:34:25.838 this rows are summing to zero.

NOTE Confidence: 0.680669108

00:34:25.840 --> 00:34:29.064 And also if you look at the first

NOTE Confidence: 0.680669108

00:34:29.064 --> 00:34:31.035 eigenvector which is the first

NOTE Confidence: 0.680669108

00:34:31.035 --> 00:34:33.195 column in this big matrix U,

NOTE Confidence: 0.680669108

00:34:33.200 --> 00:34:35.120 it is just point 5.5.

NOTE Confidence: 0.680669108

00:34:35.120 --> 00:34:36.324 It's like not changing.

NOTE Confidence: 0.680669108

00:34:36.324 --> 00:34:38.480 So that's what I was showing here.

NOTE Confidence: 0.680669108

00:34:38.480 --> 00:34:41.240 There's no change that is U0.

NOTE Confidence: 0.680669108

00:34:41.240 --> 00:34:44.000 So this is an eigenvector corresponding

NOTE Confidence: 0.680669108

00:34:44.000 --> 00:34:46.720 to this eigenvalue zero in the

NOTE Confidence: 0.680669108

00:34:46.720 --> 00:34:49.360 second column here is the eigenvector

NOTE Confidence: 0.680669108

00:34:49.360 --> 00:34:52.112 corresponding to the eigenvalue

NOTE Confidence: 0.680669108

00:34:52.112 --> 00:34:55.840 here and the third eigenvector

NOTE Confidence: 0.680669108

00:34:55.840 --> 00:34:57.900 corresponding this eigenvalue 4th

NOTE Confidence: 0.680669108

00:34:57.900 --> 00:35:00.475 1 corresponding to this eigenvalue.

NOTE Confidence: 0.680669108

00:35:00.480 --> 00:35:02.712 So the information in the data

NOTE Confidence: 0.680669108

00:35:02.712 --> 00:35:05.352 in the graph we have represented

NOTE Confidence: 0.680669108

00:35:05.352 --> 00:35:08.556 the information in terms of these

NOTE Confidence: 0.680669108

00:35:08.556 --> 00:35:10.760 eigenvalues and eigenvectors.

NOTE Confidence: 0.680669108

00:35:10.760 --> 00:35:14.564 Once we have these eigenvalues and

NOTE Confidence: 0.680669108

00:35:14.564 --> 00:35:17.124 eigenvectors and always remember that

NOTE Confidence: 0.680669108

00:35:17.124 --> 00:35:20.010 this eigenvalues are nothing but the

NOTE Confidence: 0.680669108

00:35:20.092 --> 00:35:22.720 frequencies and this U's are nothing.

NOTE Confidence: 0.680669108

00:35:22.720 --> 00:35:25.756 They are analogous to the sinusoids

NOTE Confidence: 0.680669108

00:35:25.756 --> 00:35:27.274 of different frequencies.

NOTE Confidence: 0.680669108

00:35:27.280 --> 00:35:31.352 And once we have that we can do some

NOTE Confidence: 0.680669108

00:35:31.352 --> 00:35:33.992 frequency analysis on this graphs.

NOTE Confidence: 0.680669108

00:35:34.000 --> 00:35:38.896 So if you look at this vertex

NOTE Confidence: 0.680669108

00:35:38.896 --> 00:35:40.720 ordering here again.

NOTE Confidence: 0.680669108

00:35:40.720 --> 00:35:41.920 So this is just first case.

NOTE Confidence: 0.680669108

00:35:41.920 --> 00:35:44.302 Here I have just plotted all

NOTE Confidence: 0.680669108

00:35:44.302 --> 00:35:46.360 the four eigen vectors here.

NOTE Confidence: 0.680669108

00:35:46.360 --> 00:35:48.719 The first one is of course constant.

NOTE Confidence: 0.680669108

00:35:48.720 --> 00:35:49.920 But if what if?

NOTE Confidence: 0.85272479875

00:35:54.400 --> 00:35:57.360 What if We want to represent this signal.

NOTE Confidence: 0.85272479875

00:35:57.360 --> 00:36:01.652 So we have at node 1A value of five,

NOTE Confidence: 0.85272479875

00:36:01.652 --> 00:36:06.235 at node 2A value of two node 3A value of 6,

NOTE Confidence: 0.85272479875

00:36:06.240 --> 00:36:08.958 at node four we have a value of nine.

NOTE Confidence: 0.85272479875

00:36:08.960 --> 00:36:11.468 And we want to represent this

NOTE Confidence: 0.85272479875

00:36:11.468 --> 00:36:14.015 signal as a combination of these

NOTE Confidence: 0.85272479875

00:36:14.015 --> 00:36:16.352 harmonics or the eigen vectors.

NOTE Confidence: 0.85272479875

00:36:16.352 --> 00:36:20.540 So this signal F1 which is a vector here

NOTE Confidence: 0.85272479875

00:36:20.540 --> 00:36:23.778 can be represented as a linear combination

NOTE Confidence: 0.85272479875

00:36:23.778 --> 00:36:28.156 of these eigen vectors of this laplacian.

NOTE Confidence: 0.85272479875

00:36:28.160 --> 00:36:30.968 And these coefficients are nothing but

NOTE Confidence: 0.85272479875

00:36:30.968 --> 00:36:33.840 the graph Fourier transform coefficients.

NOTE Confidence: 0.85272479875

00:36:33.840 --> 00:36:39.148 So I need 11 worth of weight of

NOTE Confidence: 0.85272479875

00:36:39.148 --> 00:36:41.633 first eigen vector negative point

NOTE Confidence: 0.85272479875

00:36:41.640 --> 00:36:44.080 1.83 weighted second eigenvector and

NOTE Confidence: 0.85272479875

00:36:44.080 --> 00:36:47.393 accordingly this is the weight of the

NOTE Confidence: 0.85272479875

00:36:47.393 --> 00:36:49.793 third eigenvector and I can represent

NOTE Confidence: 0.85272479875

00:36:49.793 --> 00:36:52.525 the signal and so I have decomposed

NOTE Confidence: 0.85272479875

00:36:52.525 --> 00:36:55.674 the signal in these four frequency in

NOTE Confidence: 0.85272479875

00:36:55.674 --> 00:36:58.278 terms of these four frequency harmonics.

NOTE Confidence: 0.85272479875

00:36:58.280 --> 00:37:03.566 So it is saying that I have as

NOTE Confidence: 0.85272479875

00:37:03.566 --> 00:37:05.196 compared to the low frequencies.

NOTE Confidence: 0.85272479875

00:37:05.200 --> 00:37:08.325 I have high frequency information

NOTE Confidence: 0.85272479875

00:37:08.325 --> 00:37:10.200 as well here.

NOTE Confidence: 0.85272479875

00:37:10.200 --> 00:37:14.880 And if I change the vertex ordering here,

NOTE Confidence: 0.85272479875

00:37:14.880 --> 00:37:18.720 so here this one became four and

NOTE Confidence: 0.85272479875

00:37:18.720 --> 00:37:22.192 what if I just represent 11 here?

NOTE Confidence: 0.85272479875

00:37:22.192 --> 00:37:25.104 So if you write down the Laplacian,

NOTE Confidence: 0.85272479875

00:37:25.104 --> 00:37:26.434 it is going to change.

NOTE Confidence: 0.85272479875

00:37:26.440 --> 00:37:28.080 So it is going to be a permuted.

NOTE Confidence: 0.85272479875

00:37:28.080 --> 00:37:30.439 So in the second, in the second-half,

NOTE Confidence: 0.85272479875

00:37:30.440 --> 00:37:33.275 when we change the order ordering here,

NOTE Confidence: 0.85272479875

00:37:33.280 --> 00:37:36.512 this Laplacian is going to be a permuted

NOTE Confidence: 0.85272479875

00:37:36.512 --> 00:37:39.360 version of this Laplacian on the left.

NOTE Confidence: 0.85272479875

00:37:39.360 --> 00:37:42.560 But the representation here changed.

NOTE Confidence: 0.85272479875

00:37:42.560 --> 00:37:44.528 And if I compute,

NOTE Confidence: 0.85272479875

00:37:44.528 --> 00:37:47.480 but if I compute these eigenvalues,

NOTE Confidence: 0.85272479875

00:37:47.480 --> 00:37:48.688 as you can see,

NOTE Confidence: 0.85272479875

00:37:48.688 --> 00:37:50.198 just write down the Laplacian,

NOTE Confidence: 0.85272479875

00:37:50.200 --> 00:37:51.571 compute the eigenvalues,

NOTE Confidence: 0.85272479875

00:37:51.571 --> 00:37:54.628 it's going to be similar in exactly

NOTE Confidence: 0.85272479875

00:37:54.628 --> 00:37:56.476 the same in both the cases.

NOTE Confidence: 0.85272479875

00:37:56.480 --> 00:37:59.000 It doesn't matter how you order

NOTE Confidence: 0.85272479875

00:37:59.000 --> 00:37:59.840 these vertices.

NOTE Confidence: 0.85272479875

00:37:59.840 --> 00:38:02.836 And if you look at these eigenvectors,

NOTE Confidence: 0.85272479875

00:38:02.840 --> 00:38:03.920 first one is constant.

NOTE Confidence: 0.85272479875

00:38:03.920 --> 00:38:06.320 If you look at the 2nd eigenvector,

NOTE Confidence: 0.85272479875

00:38:06.320 --> 00:38:08.328 it's just a permutation.

NOTE Confidence: 0.85272479875

00:38:08.328 --> 00:38:11.505 So node one became node 4.

NOTE Confidence: 0.85272479875

00:38:11.505 --> 00:38:16.440 So if you look at the 4th,

NOTE Confidence: 0.85272479875

00:38:16.440 --> 00:38:17.860 the 1st value here,

NOTE Confidence: 0.85272479875

00:38:17.860 --> 00:38:20.480 it went to the 4th value here.

NOTE Confidence: 0.85272479875

00:38:20.480 --> 00:38:22.120 So it just permuted accordingly.

NOTE Confidence: 0.85272479875

00:38:22.120 --> 00:38:23.672 So in this case,

NOTE Confidence: 0.85272479875

00:38:23.672 --> 00:38:26.000 we don't have to worry about

NOTE Confidence: 0.85272479875

00:38:26.000 --> 00:38:30.160 the ordering of these vertices.

NOTE Confidence: 0.85272479875

00:38:30.160 --> 00:38:32.560 So again, if I plot these,

NOTE Confidence: 0.85272479875

00:38:32.560 --> 00:38:33.859 all the eigenvectors,

NOTE Confidence: 0.85272479875

00:38:33.859 --> 00:38:36.864 you can see first one is constant

NOTE Confidence: 0.85272479875

00:38:36.864 --> 00:38:38.568 node ordering orderings are

NOTE Confidence: 0.85272479875

00:38:38.568 --> 00:38:40.599 different in these two cases.

NOTE Confidence: 0.85272479875

00:38:40.600 --> 00:38:45.594 But you see the association here the the

NOTE Confidence: 0.85272479875

00:38:45.594 --> 00:38:49.279 eigenvectors are also permuting accordingly.

NOTE Confidence: 0.85272479875

00:38:49.280 --> 00:38:51.080 So what does that tell us?

NOTE Confidence: 0.85272479875

00:38:51.080 --> 00:38:54.475 So it tells us that no matter

NOTE Confidence: 0.85272479875

00:38:54.475 --> 00:38:58.455 what how you order the vertices,

NOTE Confidence: 0.85272479875

00:38:58.455 --> 00:39:00.880 your representation in the frequency

NOTE Confidence: 0.85272479875

00:39:00.880 --> 00:39:03.918 domain is not going to change at all.

NOTE Confidence: 0.85272479875

00:39:03.920 --> 00:39:06.600 So if you look at the signal here,

NOTE Confidence: 0.85272479875

00:39:06.600 --> 00:39:07.440 so 5269.

NOTE Confidence: 0.85272479875

00:39:07.440 --> 00:39:11.400 So why is five in the first entry here?

NOTE Confidence: 0.85272479875

00:39:11.400 --> 00:39:14.838 Because node one or this particular

NOTE Confidence: 0.85272479875

00:39:14.840 --> 00:39:17.594 region in the brain has a value of five.

NOTE Confidence: 0.85272479875

00:39:17.600 --> 00:39:19.032 That's why 5 here.

NOTE Confidence: 0.85272479875

00:39:19.032 --> 00:39:20.822 And these are the corresponding

NOTE Confidence: 0.85272479875

00:39:20.822 --> 00:39:21.760 eigen vectors.

NOTE Confidence: 0.85272479875

00:39:21.760 --> 00:39:23.695 These are the Fourier coefficients

NOTE Confidence: 0.85272479875

00:39:23.695 --> 00:39:26.692 as you can see see in both cases

NOTE Confidence: 0.85272479875

00:39:26.692 --> 00:39:28.988 F1 and F2 which are exactly the

NOTE Confidence: 0.85272479875

00:39:29.070 --> 00:39:31.455 same information but with different

NOTE Confidence: 0.85272479875

00:39:31.455 --> 00:39:33.480 vertex ordering or node ordering.

NOTE Confidence: 0.85272479875

00:39:33.480 --> 00:39:35.532 But there's no change in the

NOTE Confidence: 0.85272479875

00:39:35.532 --> 00:39:36.558 Fourier domain as.

NOTE Confidence: 0.85272479875

00:39:36.560 --> 00:39:40.800 So it doesn't matter which ordering you use.

NOTE Confidence: 0.901116399166667

00:39:40.800 --> 00:39:42.585 Again, I think this is a repetition

NOTE Confidence: 0.901116399166667

00:39:42.585 --> 00:39:44.120 of the same concept here.

NOTE Confidence: 0.901116399166667

00:39:44.120 --> 00:39:46.500 Again, this another case,

NOTE Confidence: 0.901116399166667

00:39:46.500 --> 00:39:50.878 there's no change in the in the signal

NOTE Confidence: 0.901116399166667

00:39:50.878 --> 00:39:52.276 representation frequency domain.

NOTE Confidence: 0.901116399166667

00:39:52.280 --> 00:39:55.436 So the message here is that

NOTE Confidence: 0.901116399166667

00:39:55.440 --> 00:39:59.022 when we order the vertices 1234,

NOTE Confidence: 0.901116399166667

00:39:59.022 --> 00:40:00.288 it doesn't matter.

NOTE Confidence: 0.901116399166667

00:40:00.288 --> 00:40:03.172 We can just take the, take the,

NOTE Confidence: 0.901116399166667

00:40:03.172 --> 00:40:05.874 we can just write down the Laplacian

NOTE Confidence: 0.901116399166667

00:40:05.874 --> 00:40:08.534 and compute the eigenvalues and

NOTE Confidence: 0.901116399166667

00:40:08.534 --> 00:40:10.730 eigenvectors that will correspond

NOTE Confidence: 0.901116399166667

00:40:10.730 --> 00:40:13.970 to the graph frequencies and graph

NOTE Confidence: 0.901116399166667

00:40:13.970 --> 00:40:17.741 harmonics or graph sinus or as you can

NOTE Confidence: 0.901116399166667

00:40:17.741 --> 00:40:20.320 call it and and there's no change in

NOTE Confidence: 0.901116399166667

00:40:20.320 --> 00:40:21.600 the frequency domain representation.

NOTE Confidence: 0.95231982

00:40:25.360 --> 00:40:32.040 Now once we have this sort of frequency

NOTE Confidence: 0.95231982

00:40:32.040 --> 00:40:38.400 interpretation of our data lying on a graph,

NOTE Confidence: 0.95231982

00:40:38.400 --> 00:40:41.200 we want to define some other concepts

NOTE Confidence: 0.95231982

00:40:41.200 --> 00:40:42.489 like convolution, modulation,

NOTE Confidence: 0.95231982

00:40:42.489 --> 00:40:45.423 translation that are very basic operations

NOTE Confidence: 0.95231982

00:40:45.423 --> 00:40:47.680 in classical segment processing.

NOTE Confidence: 0.95231982

00:40:47.680 --> 00:40:51.028 But how do we do, how do we define

NOTE Confidence: 0.95231982

00:40:51.028 --> 00:40:54.040 these kind of concepts in graph domain?

NOTE Confidence: 0.95231982

00:40:54.040 --> 00:40:57.309 So all the all the all these concepts

NOTE Confidence: 0.95231982

00:40:57.309 --> 00:41:00.480 are defined in the graph domain with

NOTE Confidence: 0.95231982

00:41:00.480 --> 00:41:03.120 the help of graph Fourier transform.

NOTE Confidence: 0.95231982

00:41:03.120 --> 00:41:04.772 All of it is based on the

NOTE Confidence: 0.95231982

00:41:04.772 --> 00:41:05.480 graph Fourier transform.

NOTE Confidence: 0.95231982

00:41:05.480 --> 00:41:07.167 So the basic building block is going

NOTE Confidence: 0.95231982

00:41:07.167 --> 00:41:09.160 to be the Fourier transform here.

NOTE Confidence: 0.95231982

00:41:09.160 --> 00:41:11.596 So again we have a graph here and

NOTE Confidence: 0.95231982

00:41:11.596 --> 00:41:14.212 we have Laplacian and its eigen

NOTE Confidence: 0.95231982

00:41:14.212 --> 00:41:17.294 values and eigen vectors shown here.

NOTE Confidence: 0.95231982

00:41:17.294 --> 00:41:21.123 And let's say I had two different

NOTE Confidence: 0.95231982

00:41:21.123 --> 00:41:24.467 signals or two different data points

NOTE Confidence: 0.95231982

00:41:24.467 --> 00:41:27.880 lying on this this same graph.

NOTE Confidence: 0.95231982

00:41:27.880 --> 00:41:32.320 How do I convolve these two graph signals?

NOTE Confidence: 0.95231982

00:41:32.320 --> 00:41:37.213 So the way it is done is what we do

NOTE Confidence: 0.95231982

00:41:37.213 --> 00:41:39.799 is first compute the graph Fourier

NOTE Confidence: 0.95231982

00:41:39.799 --> 00:41:42.279 transform of the 1st signal F here,

NOTE Confidence: 0.95231982

00:41:42.280 --> 00:41:44.705 then compute the graph Fourier

NOTE Confidence: 0.95231982

00:41:44.705 --> 00:41:47.799 transform of the 2nd signal G here,

NOTE Confidence: 0.95231982

00:41:47.800 --> 00:41:50.800 do point wise multiplication and

NOTE Confidence: 0.95231982

00:41:50.800 --> 00:41:54.305 then take the inverse transform that

NOTE Confidence: 0.95231982

00:41:54.305 --> 00:41:58.345 is define that is going to be your

NOTE Confidence: 0.95231982

00:41:58.345 --> 00:42:02.394 convolve signal H so F convolve.

NOTE Confidence: 0.95231982

00:42:02.394 --> 00:42:05.866 So F convolve with G gives us this

NOTE Confidence: 0.95231982

00:42:05.866 --> 00:42:09.696 H signal here and this convolution

NOTE Confidence: 0.95231982

00:42:09.696 --> 00:42:11.960 operation in classical signal processing,

NOTE Confidence: 0.95231982

00:42:11.960 --> 00:42:13.960 this convolution operation is also

NOTE Confidence: 0.95231982

00:42:13.960 --> 00:42:15.798 called like filtering operation where

NOTE Confidence: 0.95231982

00:42:15.798 --> 00:42:18.170 this can be your original signal and

NOTE Confidence: 0.95231982

00:42:18.170 --> 00:42:20.114 this can be your filter or a kernel.

NOTE Confidence: 0.95231982

00:42:20.120 --> 00:42:22.794 How are we you going to come

NOTE Confidence: 0.95231982

00:42:22.800 --> 00:42:24.480 compute this convolved product?

NOTE Confidence: 0.95231982

00:42:24.480 --> 00:42:28.304 That is how you do it in graph domain.

NOTE Confidence: 0.95231982

00:42:28.304 --> 00:42:31.496 And another side information here is

NOTE Confidence: 0.95231982

00:42:31.496 --> 00:42:34.856 that this basic operation of graph

NOTE Confidence: 0.95231982

00:42:34.856 --> 00:42:38.156 convolution which is started from the

NOTE Confidence: 0.95231982

00:42:38.160 --> 00:42:40.260 intuition of this with graph Fourier

NOTE Confidence: 0.95231982

00:42:40.260 --> 00:42:42.778 transform was the base is the basic

NOTE Confidence: 0.95231982

00:42:42.778 --> 00:42:44.878 building block of graph neural networks.

NOTE Confidence: 0.95231982

00:42:44.880 --> 00:42:47.405 If you've heard of that graph

NOTE Confidence: 0.95231982

00:42:47.405 --> 00:42:50.430 neural networks and there are

NOTE Confidence: 0.95231982

00:42:50.430 --> 00:42:52.680 many other architectures proposed,

NOTE Confidence: 0.95231982

00:42:52.680 --> 00:42:55.280 simplified architectures proposed based on

NOTE Confidence: 0.95231982

00:42:55.280 --> 00:42:58.600 the same concept here the graph convolution.

NOTE Confidence: 0.782700665714286

00:43:02.040 --> 00:43:03.958 So why are we defining these concepts?

NOTE Confidence: 0.782700665714286

00:43:03.960 --> 00:43:06.688 So because these concepts are going to help

NOTE Confidence: 0.782700665714286

00:43:06.688 --> 00:43:09.917 us to represent the signal that we have

NOTE Confidence: 0.91468887

00:43:12.280 --> 00:43:13.704 at each time point.

NOTE Confidence: 0.91468887

00:43:13.704 --> 00:43:15.484 That's why we are discussing

NOTE Confidence: 0.91468887

00:43:15.484 --> 00:43:18.080 this these concepts and this

NOTE Confidence: 0.91468887

00:43:18.080 --> 00:43:20.320 is another graph translation.

NOTE Confidence: 0.91468887

00:43:20.320 --> 00:43:22.396 How are we going to define

NOTE Confidence: 0.91468887

00:43:22.396 --> 00:43:23.434 the graph translation?

NOTE Confidence: 0.91468887

00:43:23.440 --> 00:43:24.916 I believe we can skip it.

NOTE Confidence: 0.91468887

00:43:24.920 --> 00:43:27.800 It's not that important here in our context.

NOTE Confidence: 0.91468887

00:43:27.800 --> 00:43:31.256 But the message here is that if you

NOTE Confidence: 0.91468887

00:43:31.256 --> 00:43:34.184 want to define these classical signal

NOTE Confidence: 0.91468887

00:43:34.184 --> 00:43:37.560 processing concepts in graph domain,

NOTE Confidence: 0.91468887

00:43:37.560 --> 00:43:40.200 graph Fourier transform is the way

NOTE Confidence: 0.91468887

00:43:40.200 --> 00:43:42.488 that helps us to theoretically

NOTE Confidence: 0.91468887

00:43:42.488 --> 00:43:43.874 define these concepts.

NOTE Confidence: 0.91468887

00:43:43.880 --> 00:43:46.620 Otherwise because of this irregular

NOTE Confidence: 0.91468887

00:43:46.620 --> 00:43:49.360 structure is irregular structure of

NOTE Confidence: 0.91468887

00:43:49.441 --> 00:43:52.392 this this graph that prohibits us to

NOTE Confidence: 0.91468887

00:43:52.392 --> 00:43:55.080 kind of extend the concepts directly.

NOTE Confidence: 0.91468887

00:43:55.080 --> 00:43:57.290 But then graph Fourier transform

NOTE Confidence: 0.91468887

00:43:57.290 --> 00:44:00.043 comes into picture and eases our

NOTE Confidence: 0.91468887

00:44:00.043 --> 00:44:01.435 theoretical understanding of

NOTE Confidence: 0.91468887

00:44:01.435 --> 00:44:04.219 this data lying on this complex

NOTE Confidence: 0.91468887

00:44:04.296 --> 00:44:06.399 network irregular structures.

NOTE Confidence: 0.857373557142857

00:44:10.520 --> 00:44:13.397 So this is a table which is

NOTE Confidence: 0.857373557142857

00:44:13.400 --> 00:44:15.820 comparing the the signal processing

NOTE Confidence: 0.857373557142857

00:44:15.820 --> 00:44:18.240 concept side by side here.

NOTE Confidence: 0.857373557142857

00:44:18.240 --> 00:44:20.640 So this is Fourier transform.

NOTE Confidence: 0.857373557142857

00:44:20.640 --> 00:44:22.992 In Fourier transform sinusoids

NOTE Confidence: 0.857373557142857

00:44:22.992 --> 00:44:25.932 or complex exponentials are our

NOTE Confidence: 0.857373557142857

00:44:25.932 --> 00:44:29.190 basis and we kind of represent our

NOTE Confidence: 0.857373557142857

00:44:29.190 --> 00:44:31.697 signal as a linear combination

NOTE Confidence: 0.857373557142857

00:44:31.697 --> 00:44:35.255 of these sinusoids and same.

NOTE Confidence: 0.857373557142857

00:44:35.255 --> 00:44:39.845 Similarly the graph in graph signal

NOTE Confidence: 0.857373557142857

00:44:39.845 --> 00:44:43.516 processing we represent our signal F as A

NOTE Confidence: 0.879879146363637

00:44:45.560 --> 00:44:47.840 as a linear combination of the

NOTE Confidence: 0.879879146363637

00:44:47.840 --> 00:44:50.040 eigenvectors of the graph Laplacian

NOTE Confidence: 0.879879146363637

00:44:50.040 --> 00:44:54.466 and convolution sum we defined

NOTE Confidence: 0.879879146363637

00:44:54.466 --> 00:44:55.878 through graph Fourier transform.

NOTE Confidence: 0.879879146363637

00:44:55.880 --> 00:44:58.450 Then similar. Similarly you can

NOTE Confidence: 0.879879146363637

00:44:58.450 --> 00:45:00.506 define translation and modulation

NOTE Confidence: 0.879879146363637

00:45:00.506 --> 00:45:03.198 using this graph Fourier transform.

NOTE Confidence: 0.80954694

00:45:05.400 --> 00:45:09.177 SO question, I think this table

NOTE Confidence: 0.80954694

00:45:09.177 --> 00:45:10.862 really helps me to understand

NOTE Confidence: 0.80954694

00:45:10.862 --> 00:45:13.027 what question I have because you

NOTE Confidence: 0.80954694

00:45:13.027 --> 00:45:14.917 kind of put everything together.

NOTE Confidence: 0.922110275333333

00:45:19.040 --> 00:45:22.392 What I assume, and I need you to

NOTE Confidence: 0.922110275333333

00:45:22.392 --> 00:45:25.478 correct me if I'm right or wrong,

NOTE Confidence: 0.922110275333333

00:45:25.480 --> 00:45:28.725 is that graph signal processing

NOTE Confidence: 0.922110275333333

00:45:28.725 --> 00:45:31.970 gives us better approximation of

NOTE Confidence: 0.922110275333333

00:45:32.078 --> 00:45:35.600 complex signals because it has richer

NOTE Confidence: 0.729307975

00:45:38.320 --> 00:45:39.835 like basic function,

NOTE Confidence: 0.729307975

00:45:39.835 --> 00:45:42.360 richer set of basic function.

NOTE Confidence: 0.729307975

00:45:42.360 --> 00:45:44.334 Or is there any other advantage if

NOTE Confidence: 0.729307975

00:45:44.334 --> 00:45:46.143 you compare this classical signal

NOTE Confidence: 0.729307975

00:45:46.143 --> 00:45:48.398 processing with graph signal processing,

NOTE Confidence: 0.729307975

00:45:48.400 --> 00:45:50.560 What is the key advantage here?

NOTE Confidence: 0.81290614

00:45:51.200 --> 00:45:52.800 Yeah. So first of all,

NOTE Confidence: 0.81290614

00:45:52.800 --> 00:45:54.396 these concepts here,

NOTE Confidence: 0.81290614

00:45:54.396 --> 00:45:57.056 you cannot define in the

NOTE Confidence: 0.81290614

00:45:57.056 --> 00:45:59.040 graph domain directly.

NOTE Confidence: 0.81290614

00:45:59.040 --> 00:46:01.280 So whatever we see in the first,

NOTE Confidence: 0.81290614

00:46:01.280 --> 00:46:02.400 in the second column here,

NOTE Confidence: 0.81290614

00:46:02.400 --> 00:46:03.531 the classical signal,

NOTE Confidence: 0.81290614

00:46:03.531 --> 00:46:05.416 these concepts are not applicable

NOTE Confidence: 0.81290614

00:46:05.416 --> 00:46:06.960 directly in graph domain.

NOTE Confidence: 0.81290614

00:46:06.960 --> 00:46:09.326 You can't define it because of the

NOTE Confidence: 0.81290614

00:46:09.326 --> 00:46:11.000 irregular structure of the graph.

NOTE Confidence: 0.777348333888889

00:46:11.440 --> 00:46:13.780 So you're talking about pretty much

NOTE Confidence: 0.777348333888889

00:46:13.780 --> 00:46:16.693 behaviour of signal in one node versus

NOTE Confidence: 0.777348333888889

00:46:16.693 --> 00:46:18.833 behaviour or signal multiple nodes.

NOTE Confidence: 0.777348333888889

00:46:18.840 --> 00:46:20.600 That's the biggest difference.

NOTE Confidence: 0.777348333888889

00:46:20.600 --> 00:46:23.118 Oh, yes, exactly. OK, OK, thank

NOTE Confidence: 0.7090725025

00:46:23.280 --> 00:46:24.600 you. Yeah, yeah.

NOTE Confidence: 0.7090725025

00:46:24.600 --> 00:46:26.800 And in classical signal crossing,

NOTE Confidence: 0.7090725025

00:46:26.800 --> 00:46:28.876 the graph is always the same.

NOTE Confidence: 0.7090725025

00:46:28.880 --> 00:46:31.032 It's like T1T2T3T4,

NOTE Confidence: 0.7090725025

00:46:31.032 --> 00:46:34.474 it's arranged by time. It's not.

NOTE Confidence: 0.7090725025

00:46:34.474 --> 00:46:35.959 Images are arranged by a

NOTE Confidence: 0.7090725025

00:46:35.959 --> 00:46:37.400 very nicely spatial location.

NOTE Confidence: 0.7090725025

00:46:37.400 --> 00:46:39.760 You look at the images like from X to Y,

NOTE Confidence: 0.7090725025

00:46:39.760 --> 00:46:42.052 it's very nicely arranged.

NOTE Confidence: 0.7090725025

00:46:42.052 --> 00:46:44.840 But here if you change the graph,

NOTE Confidence: 0.7090725025

00:46:44.840 --> 00:46:45.700 everything changes.

NOTE Confidence: 0.7090725025

00:46:45.700 --> 00:46:48.280 If you change the graphic structure,

NOTE Confidence: 0.7090725025

00:46:48.280 --> 00:46:49.480 everything changes.

NOTE Confidence: 0.7090725025

00:46:49.480 --> 00:46:51.880 So it's like behaviour,

NOTE Confidence: 0.7090725025

00:46:51.880 --> 00:46:53.842 the time behaviour at one node

NOTE Confidence: 0.7090725025

00:46:53.842 --> 00:46:55.904 versus time behaviour if you want

NOTE Confidence: 0.7090725025

00:46:55.904 --> 00:46:57.974 to combine the all the different

NOTE Confidence: 0.7090725025

00:46:57.974 --> 00:47:00.361 regions together and if you want to

NOTE Confidence: 0.7090725025

00:47:00.361 --> 00:47:01.996 consider the information coming from

NOTE Confidence: 0.7090725025

00:47:02.000 --> 00:47:03.998 my neighbours in the brain regions.

NOTE Confidence: 0.7090725025

00:47:04.000 --> 00:47:06.149 So I have to take this into

NOTE Confidence: 0.7090725025

00:47:06.149 --> 00:47:07.725 account using this graph

NOTE Confidence: 0.7090725025

00:47:07.725 --> 00:47:09.477 signal processing techniques.

NOTE Confidence: 0.721327888

00:47:09.560 --> 00:47:12.984 So basically dimensionality component,

NOTE Confidence: 0.721327888

00:47:12.984 --> 00:47:15.274 right? So not necessarily complexity,

NOTE Confidence: 0.721327888

00:47:15.274 --> 00:47:17.662 but it's more about how much

NOTE Confidence: 0.721327888

00:47:17.662 --> 00:47:19.120 information you can describe.

NOTE Confidence: 0.774498074285714

00:47:20.400 --> 00:47:22.536 Yeah, this is space can be another dimension

NOTE Confidence: 0.774498074285714

00:47:22.536 --> 00:47:24.640 if you can look at it. Yeah, thank you.

NOTE Confidence: 0.7225602

00:47:32.720 --> 00:47:33.160 OK,

NOTE Confidence: 0.9432956

00:47:35.400 --> 00:47:39.390 so this is a very interesting example

NOTE Confidence: 0.9432956

00:47:39.390 --> 00:47:42.785 of graph Fourier transform how it

NOTE Confidence: 0.9432956

00:47:42.785 --> 00:47:46.718 can be helpful and it can give us the

NOTE Confidence: 0.9432956

00:47:46.720 --> 00:47:49.840 information lying in the data directly

NOTE Confidence: 0.9432956

00:47:49.840 --> 00:47:54.180 like with a just very little effort,

NOTE Confidence: 0.9432956

00:47:54.180 --> 00:47:57.812 you can find the find some of the

NOTE Confidence: 0.9432956

00:47:57.812 --> 00:47:59.677 structure in the data easily.

NOTE Confidence: 0.9432956

00:47:59.680 --> 00:48:01.920 So this is a sensor net sensor network.

NOTE Confidence: 0.9432956

00:48:01.920 --> 00:48:05.120 So this is a map of United States,

NOTE Confidence: 0.9432956

00:48:05.120 --> 00:48:08.480 Florida, we are somewhere here.

NOTE Confidence: 0.9432956

00:48:08.480 --> 00:48:13.899 So this is just each node is

NOTE Confidence: 0.9432956

00:48:13.899 --> 00:48:16.518 representing geographical location

NOTE Confidence: 0.9432956

00:48:16.520 --> 00:48:18.728 and the connectivity of the nodes

NOTE Confidence: 0.9432956

00:48:18.728 --> 00:48:22.100 is based on the the geography

NOTE Confidence: 0.9432956

00:48:22.100 --> 00:48:26.008 like the the nodes are connected,

NOTE Confidence: 0.9432956

00:48:26.008 --> 00:48:28.016 the two geographical locations

NOTE Confidence: 0.9432956

00:48:28.016 --> 00:48:30.429 are connected by these links if

NOTE Confidence: 0.9432956

00:48:30.429 --> 00:48:32.693 they are close to each other like

NOTE Confidence: 0.9432956

00:48:32.693 --> 00:48:33.958 for example this California is

NOTE Confidence: 0.9432956

00:48:33.958 --> 00:48:35.520 not connected to the East Coast.

NOTE Confidence: 0.9432956

00:48:35.520 --> 00:48:39.088 So only this California regions

NOTE Confidence: 0.9432956

00:48:39.088 --> 00:48:41.448 based on the distance between

NOTE Confidence: 0.9432956

00:48:41.448 --> 00:48:43.760 the geographical locations,

NOTE Confidence: 0.9432956

00:48:43.760 --> 00:48:45.880 you can connect these nodes.

NOTE Confidence: 0.9432956

00:48:45.880 --> 00:48:49.674 And once you have this nodes and

NOTE Confidence: 0.9432956

00:48:49.674 --> 00:48:52.000 connectivity information and you

NOTE Confidence: 0.9432956

00:48:52.000 --> 00:48:54.480 have this temperature readings

NOTE Confidence: 0.9432956

00:48:54.480 --> 00:48:58.316 at at one time point you have

NOTE Confidence: 0.9432956

00:48:58.316 --> 00:49:01.696 temperature readings of all the these

NOTE Confidence: 0.9432956

00:49:01.696 --> 00:49:04.000 sensors distributed across the US.

NOTE Confidence: 0.893463449090909

00:49:06.400 --> 00:49:08.199 As you can see S is little

NOTE Confidence: 0.893463449090909

00:49:08.199 --> 00:49:09.400 hotter than the north.

NOTE Confidence: 0.893463449090909

00:49:09.400 --> 00:49:13.278 But this is just one time slice.

NOTE Confidence: 0.893463449090909

00:49:13.280 --> 00:49:15.705 And let's say some sensors

NOTE Confidence: 0.893463449090909

00:49:15.705 --> 00:49:16.675 are malfunctioning,

NOTE Confidence: 0.893463449090909

00:49:16.680 --> 00:49:19.440 they are not giving you the correct reading.

NOTE Confidence: 0.893463449090909

00:49:19.440 --> 00:49:21.520 So by the nature of the data here,

NOTE Confidence: 0.893463449090909

00:49:21.520 --> 00:49:23.136 it is smooth data.

NOTE Confidence: 0.893463449090909

00:49:23.136 --> 00:49:26.040 What what does this smooth means here?

NOTE Confidence: 0.893463449090909

00:49:26.040 --> 00:49:28.868 If you move from 1 geographical location

NOTE Confidence: 0.893463449090909

00:49:28.868 --> 00:49:31.879 to a nearby geographical location,

NOTE Confidence: 0.893463449090909

00:49:31.880 --> 00:49:32.786 the temperature change

NOTE Confidence: 0.893463449090909

00:49:32.786 --> 00:49:34.598 is not going to be much.

NOTE Confidence: 0.893463449090909

00:49:34.600 --> 00:49:36.728 So temperature in New York is more or

NOTE Confidence: 0.893463449090909

00:49:36.728 --> 00:49:39.035 less going to be temperature in New Haven,

NOTE Confidence: 0.893463449090909

00:49:39.040 --> 00:49:42.001 but it is going to it can be very

NOTE Confidence: 0.893463449090909

00:49:42.001 --> 00:49:43.742 different from the temperature

NOTE Confidence: 0.893463449090909

00:49:43.742 --> 00:49:46.400 in Texas or Florida or Arizona.

NOTE Confidence: 0.893463449090909

00:49:46.400 --> 00:49:49.075 But let's say some sensors

NOTE Confidence: 0.893463449090909

00:49:49.075 --> 00:49:50.680 are men malfunctioning.

NOTE Confidence: 0.893463449090909

00:49:50.680 --> 00:49:53.568 So if it is malfunctioning,

NOTE Confidence: 0.893463449090909

00:49:53.568 --> 00:49:55.380 it is going to destroy this

NOTE Confidence: 0.893463449090909

00:49:55.441 --> 00:49:57.356 smoothness structure in the data.

NOTE Confidence: 0.893463449090909

00:49:57.360 --> 00:49:59.200 So if you just,

NOTE Confidence: 0.893463449090909

00:49:59.200 --> 00:50:02.608 if let's say the sensor network is perfect,

NOTE Confidence: 0.893463449090909

00:50:02.608 --> 00:50:04.960 if you take the Fourier graph,

NOTE Confidence: 0.893463449090909

00:50:04.960 --> 00:50:07.840 Fourier transform of the data here

NOTE Confidence: 0.893463449090909

00:50:07.840 --> 00:50:12.280 you will see all the you will see low

NOTE Confidence: 0.893463449090909

00:50:12.280 --> 00:50:14.680 frequency components are dominant here.

NOTE Confidence: 0.893463449090909

00:50:14.680 --> 00:50:16.798 There are no high frequency components.

NOTE Confidence: 0.893463449090909

00:50:16.800 --> 00:50:17.105 Why?

NOTE Confidence: 0.893463449090909

00:50:17.105 --> 00:50:18.935 The same reason the temperature is

NOTE Confidence: 0.893463449090909

00:50:18.935 --> 00:50:21.306 not going to change rapidly if you

NOTE Confidence: 0.893463449090909

00:50:21.306 --> 00:50:23.800 move from 11 geographical location to other.

NOTE Confidence: 0.893463449090909

00:50:23.800 --> 00:50:26.476 But if the sensor is malfunctioning,

NOTE Confidence: 0.893463449090909

00:50:26.480 --> 00:50:29.259 then what you will see is there

NOTE Confidence: 0.893463449090909

00:50:29.259 --> 00:50:32.350 will be like a spike in high

NOTE Confidence: 0.893463449090909

00:50:32.350 --> 00:50:35.016 frequency where you will you can

NOTE Confidence: 0.893463449090909

00:50:35.016 --> 00:50:37.156 tell that the temperature changed

NOTE Confidence: 0.893463449090909

00:50:37.156 --> 00:50:39.953 by a significant amount if I move

NOTE Confidence: 0.893463449090909

00:50:39.953 --> 00:50:41.753 from New Haven to Hartford,

NOTE Confidence: 0.893463449090909

00:50:41.760 --> 00:50:44.400 which is usually not the case.

NOTE Confidence: 0.893463449090909

00:50:44.400 --> 00:50:46.514 So if you just take the Fourier

NOTE Confidence: 0.893463449090909

00:50:46.514 --> 00:50:48.748 transform of one time point and if

NOTE Confidence: 0.893463449090909

00:50:48.748 --> 00:50:50.638 you get a high frequency spikes,

NOTE Confidence: 0.893463449090909

00:50:50.640 --> 00:50:53.027 you can tell there are some sensors

NOTE Confidence: 0.893463449090909

00:50:53.027 --> 00:50:55.967 in this sensor network that are not

NOTE Confidence: 0.893463449090909

00:50:55.967 --> 00:50:58.277 performing well that are malfunctioning.

NOTE Confidence: 0.893463449090909

00:50:58.280 --> 00:51:02.786 But can we pinpoint which which

NOTE Confidence: 0.893463449090909

00:51:02.786 --> 00:51:06.840 sensor is not is is malfunctioning?

NOTE Confidence: 0.965045558

00:51:15.160 --> 00:51:17.360 Do we have any answers?

NOTE Confidence: 0.965045558

00:51:17.360 --> 00:51:19.400 You can think of it a little bit

NOTE Confidence: 0.93110911875

00:51:26.450 --> 00:51:29.880 not based on this, based on this

NOTE Confidence: 0.93110911875

00:51:29.880 --> 00:51:31.650 graph, just looking at this graph,

NOTE Confidence: 0.8569088

00:51:32.250 --> 00:51:34.966 yeah. So if one sensor is malfunctioning,

NOTE Confidence: 0.8569088

00:51:34.970 --> 00:51:39.050 can we tell like not based on the in general.

NOTE Confidence: 0.8569088

00:51:39.050 --> 00:51:41.346 I have a one slice of temperature

NOTE Confidence: 0.8569088

00:51:41.346 --> 00:51:43.719 readings all over the US and I

NOTE Confidence: 0.8569088

00:51:43.719 --> 00:51:45.364 take the graph Fourier transform.

NOTE Confidence: 0.8569088

00:51:45.370 --> 00:51:49.598 I see a high frequency spike here.

NOTE Confidence: 0.8569088

00:51:49.600 --> 00:51:52.840 Then that tells me that there is an issue.

NOTE Confidence: 0.8569088

00:51:52.840 --> 00:51:55.400 There is an issue in in the network,

NOTE Confidence: 0.8569088

00:51:55.400 --> 00:51:57.480 but this doesn't tell me

NOTE Confidence: 0.8569088

00:51:57.480 --> 00:51:59.560 where exactly the issue is.

NOTE Confidence: 0.8569088

00:51:59.560 --> 00:52:02.200 So this graph Fourier transform

NOTE Confidence: 0.8569088

00:52:02.200 --> 00:52:04.312 is a global transform.

NOTE Confidence: 0.8569088

00:52:04.320 --> 00:52:08.440 Global in the sense it captures the frequency

NOTE Confidence: 0.8569088

00:52:08.440 --> 00:52:10.480 information globally lying on the graph.

NOTE Confidence: 0.8569088

00:52:10.480 --> 00:52:13.612 It cannot pinpoint where exactly the

NOTE Confidence: 0.8569088

00:52:13.612 --> 00:52:16.520 those graph frequencies are changing,

NOTE Confidence: 0.8569088

00:52:16.520 --> 00:52:19.040 where exactly in the space that is going on.

NOTE Confidence: 0.8569088

00:52:19.040 --> 00:52:20.960 It just tells you there is a change.

NOTE Confidence: 0.8569088

00:52:20.960 --> 00:52:22.960 I don't know where there is a change.

NOTE Confidence: 0.8569088

00:52:22.960 --> 00:52:24.850 There is this high frequency or low

NOTE Confidence: 0.8569088

00:52:24.850 --> 00:52:26.159 frequency information in the graph,

NOTE Confidence: 0.8569088

00:52:26.160 --> 00:52:28.920 but I don't know where in the space.

NOTE Confidence: 0.8569088

00:52:28.920 --> 00:52:30.551 So that is kind of a limitation

NOTE Confidence: 0.8569088

00:52:30.551 --> 00:52:31.800 of this Fourier transform,

NOTE Confidence: 0.8569088

00:52:31.800 --> 00:52:32.358 but it's still

NOTE Confidence: 0.833044787777778

00:52:32.520 --> 00:52:34.878 is it, is it, is it something you can,

NOTE Confidence: 0.833044787777778

00:52:34.880 --> 00:52:37.775 for instance, answer by iteratively

NOTE Confidence: 0.833044787777778

00:52:37.775 --> 00:52:41.130 dropping one of the notes and

NOTE Confidence: 0.833044787777778

00:52:41.130 --> 00:52:43.800 see if your spike disappears.

NOTE Confidence: 0.876686577142857

00:52:44.440 --> 00:52:46.610 That can be one strategy if that

NOTE Confidence: 0.876686577142857

00:52:46.610 --> 00:52:49.080 is the if that is the goal. But

NOTE Confidence: 0.649263286363636

00:52:49.280 --> 00:52:51.038 but but overall, generally we just

NOTE Confidence: 0.649263286363636

00:52:51.038 --> 00:52:52.640 know the system doesn't work. Well,

NOTE Confidence: 0.82788419875

00:52:52.880 --> 00:52:55.000 yeah, that is that can be one strategy.

NOTE Confidence: 0.82788419875

00:52:55.000 --> 00:52:58.344 But then we have some advanced tools that

NOTE Confidence: 0.82788419875

00:52:58.344 --> 00:53:01.946 can tell us that can help us tell pinpoint

NOTE Confidence: 0.82788419875

00:53:01.946 --> 00:53:05.276 exact sensor which is malfunctioning.

NOTE Confidence: 0.82788419875

00:53:05.280 --> 00:53:07.758 So for that we need this,

NOTE Confidence: 0.82788419875

00:53:07.760 --> 00:53:10.896 we need to combine the this idea of

NOTE Confidence: 0.82788419875

00:53:10.896 --> 00:53:13.756 the space as well as this idea of

NOTE Confidence: 0.82788419875

00:53:13.756 --> 00:53:15.986 frequency and we want to localize

NOTE Confidence: 0.82788419875

00:53:15.986 --> 00:53:18.674 it in space as well as frequency.

NOTE Confidence: 0.82788419875

00:53:18.680 --> 00:53:21.543 Then there the wavelets come into fake

NOTE Confidence: 0.82788419875

00:53:21.543 --> 00:53:24.319 picture or the graph is scattering

NOTE Confidence: 0.82788419875

00:53:24.320 --> 00:53:28.870 comes into the picture. So yeah.

NOTE Confidence: 0.82788419875

00:53:28.870 --> 00:53:33.280 So to motivate that here again,

NOTE Confidence: 0.82788419875

00:53:33.280 --> 00:53:35.917 if we look at so in the previous slides,

NOTE Confidence: 0.82788419875

00:53:35.920 --> 00:53:38.160 it is slide, it was about this

NOTE Confidence: 0.82788419875

00:53:38.160 --> 00:53:40.400 temperature recordings on the graph.

NOTE Confidence: 0.82788419875

00:53:40.400 --> 00:53:43.136 But if you just look at a very

NOTE Confidence: 0.82788419875

00:53:43.136 --> 00:53:44.679 classical one-dimensional chirp signal,

NOTE Confidence: 0.82788419875

00:53:44.680 --> 00:53:47.336 so let's say it's like starting,

NOTE Confidence: 0.82788419875

00:53:47.336 --> 00:53:49.640 starting at time zero,

NOTE Confidence: 0.82788419875

00:53:49.640 --> 00:53:51.758 it is a slowly varying signal,

NOTE Confidence: 0.82788419875

00:53:51.760 --> 00:53:52.526 low frequency.

NOTE Confidence: 0.82788419875

00:53:52.526 --> 00:53:55.590 Then as the time passes it's it is

NOTE Confidence: 0.82788419875

00:53:55.668 --> 00:53:58.760 varying rapidly and that is chirp up.

NOTE Confidence: 0.82788419875

00:53:58.760 --> 00:54:01.320 So frequency is going up.

NOTE Confidence: 0.82788419875

00:54:01.320 --> 00:54:04.002 And if you look at the just take the

NOTE Confidence: 0.82788419875

00:54:04.002 --> 00:54:05.968 Fourier transform of this signal,

NOTE Confidence: 0.82788419875

00:54:05.968 --> 00:54:07.760 you will see that

NOTE Confidence: 0.597456245

00:54:10.000 --> 00:54:13.933 OK, just and just reverse the signal now.

NOTE Confidence: 0.597456245

00:54:13.933 --> 00:54:17.800 So in the time T, time T is equal to 0.

NOTE Confidence: 0.597456245

00:54:17.800 --> 00:54:19.465 The signal is changing rapidly

NOTE Confidence: 0.597456245

00:54:19.465 --> 00:54:21.920 and then as the time progresses,

NOTE Confidence: 0.597456245

00:54:21.920 --> 00:54:23.744 the frequency goes down.

NOTE Confidence: 0.597456245

00:54:23.744 --> 00:54:26.480 These two are very different signals.

NOTE Confidence: 0.942714204285714

00:54:29.120 --> 00:54:31.276 But if you take the Fourier transform,

NOTE Confidence: 0.942714204285714

00:54:31.280 --> 00:54:33.436 it's going to look exactly the same.

NOTE Confidence: 0.942714204285714

00:54:33.440 --> 00:54:34.090 It cannot.

NOTE Confidence: 0.942714204285714

00:54:34.090 --> 00:54:36.040 So that that's the message here.

NOTE Confidence: 0.942714204285714

00:54:36.040 --> 00:54:38.662 The Fourier transform cannot point out

NOTE Confidence: 0.942714204285714

00:54:38.662 --> 00:54:41.439 where exactly the changes are happening.

NOTE Confidence: 0.942714204285714

00:54:41.440 --> 00:54:43.480 It can tell you what changes are happening,

NOTE Confidence: 0.942714204285714

00:54:43.480 --> 00:54:45.538 but it cannot tell you where exactly

NOTE Confidence: 0.942714204285714

00:54:45.538 --> 00:54:47.718 the thing these changes are happening.

NOTE Confidence: 0.942714204285714

00:54:47.720 --> 00:54:50.338 So in order to solve this problem

NOTE Confidence: 0.942714204285714

00:54:50.338 --> 00:54:53.142 or in order to discriminate or

NOTE Confidence: 0.942714204285714

00:54:53.142 --> 00:54:55.270 localize discriminate such issues

NOTE Confidence: 0.942714204285714

00:54:55.270 --> 00:54:59.016 or in order to localize the this

NOTE Confidence: 0.942714204285714

00:54:59.016 --> 00:55:02.760 time and frequency together, we

NOTE Confidence: 0.7713631883333333

00:55:05.040 --> 00:55:07.278 we study the concept called wavelet.

NOTE Confidence: 0.828924318

00:55:09.560 --> 00:55:12.160 So what are these wavelets?

NOTE Confidence: 0.828924318

00:55:12.160 --> 00:55:15.360 Wavelets are just nothing but a small wave.

NOTE Confidence: 0.828924318

00:55:15.360 --> 00:55:19.144 And in, in in literature you can find

NOTE Confidence: 0.828924318

00:55:19.144 --> 00:55:22.166 there are many different wavelet

NOTE Confidence: 0.828924318

00:55:22.166 --> 00:55:24.996 shapes that people have studied.

NOTE Confidence: 0.828924318

00:55:25.000 --> 00:55:28.616 Some of the very popular ones are this

NOTE Confidence: 0.828924318

00:55:28.616 --> 00:55:31.917 R wavelets and this debash wavelets.

NOTE Confidence: 0.828924318

00:55:31.920 --> 00:55:33.664 Also this Mexican hat,

NOTE Confidence: 0.828924318

00:55:33.664 --> 00:55:37.092 So these are the small waves that you

NOTE Confidence: 0.828924318

00:55:37.092 --> 00:55:40.372 just want to use in order to show in

NOTE Confidence: 0.828924318

00:55:40.372 --> 00:55:43.710 order to represent your data or represent

NOTE Confidence: 0.828924318

00:55:43.710 --> 00:55:46.610 your signal that is localized in time

NOTE Confidence: 0.828924318

00:55:46.610 --> 00:55:49.160 as well as in frequency together.

NOTE Confidence: 0.828924318

00:55:49.160 --> 00:55:51.752 And we will see how to do that exactly.

NOTE Confidence: 0.89782807

00:55:55.040 --> 00:55:57.715 So what this wavelet does

NOTE Confidence: 0.89782807

00:55:57.715 --> 00:56:00.481 is it takes shifts of this.

NOTE Confidence: 0.89782807

00:56:00.481 --> 00:56:03.190 So this is a very small kind

NOTE Confidence: 0.89782807

00:56:03.287 --> 00:56:05.304 of a wave and you want to

NOTE Confidence: 0.89782807

00:56:05.304 --> 00:56:06.834 analyse this time domain signal.

NOTE Confidence: 0.89782807

00:56:06.840 --> 00:56:10.460 So what you can do is just align

NOTE Confidence: 0.89782807

00:56:10.460 --> 00:56:14.000 this wavelet at this time window

NOTE Confidence: 0.89782807

00:56:14.000 --> 00:56:16.165 and compute the correlation between

NOTE Confidence: 0.89782807

00:56:16.165 --> 00:56:18.766 the signal and this wavelet and

NOTE Confidence: 0.89782807

00:56:18.766 --> 00:56:20.596 see how correlated they are.

NOTE Confidence: 0.89782807

00:56:20.600 --> 00:56:22.634 Then shift the same wavelet to

NOTE Confidence: 0.89782807

00:56:22.634 --> 00:56:25.048 another time window and compute some

NOTE Confidence: 0.89782807

00:56:25.048 --> 00:56:27.518 correlation coefficients or inner product,

NOTE Confidence: 0.89782807

00:56:27.520 --> 00:56:30.560 whatever you call it.

NOTE Confidence: 0.89782807

00:56:30.560 --> 00:56:35.280 And then and then save those numbers,

NOTE Confidence: 0.89782807

00:56:35.280 --> 00:56:37.534 save those numbers and then you kind

NOTE Confidence: 0.89782807

00:56:37.534 --> 00:56:39.912 of stretch the wavelet like scale the

NOTE Confidence: 0.89782807

00:56:39.912 --> 00:56:42.560 wavelet so the changes are not as rapid.

NOTE Confidence: 0.89782807

00:56:42.560 --> 00:56:45.640 So if you look at this wavelet at the bottom,

NOTE Confidence: 0.89782807

00:56:45.640 --> 00:56:46.624 it is a,

NOTE Confidence: 0.89782807

00:56:46.624 --> 00:56:48.592 it is like a stretched version

NOTE Confidence: 0.89782807

00:56:48.592 --> 00:56:50.685 of this first wavelet here.

NOTE Confidence: 0.89782807

00:56:50.685 --> 00:56:52.960 So this is called scaling.

NOTE Confidence: 0.89782807

00:56:52.960 --> 00:56:56.264 So you can scale your wavelet to

NOTE Confidence: 0.89782807

00:56:56.264 --> 00:56:59.625 like either stretch it or compress it.

NOTE Confidence: 0.89782807

00:56:59.625 --> 00:57:03.409 And based on that you can you

NOTE Confidence: 0.89782807

00:57:03.409 --> 00:57:06.103 can capture the high or low

NOTE Confidence: 0.89782807

00:57:06.103 --> 00:57:08.440 frequency variations in the signal.

NOTE Confidence: 0.89782807

00:57:08.440 --> 00:57:12.360 So that's how this wavelet wavelets work with

NOTE Confidence: 0.89782807

00:57:12.360 --> 00:57:15.478 different shifts and scales in time domain.

NOTE Confidence: 0.89782807

00:57:15.480 --> 00:57:17.200 And this is a very nice example here.

NOTE Confidence: 0.89782807

00:57:17.200 --> 00:57:18.800 If you look at this,

NOTE Confidence: 0.89782807

00:57:18.800 --> 00:57:22.800 the chirped up signal where in the beginning

NOTE Confidence: 0.89782807

00:57:22.800 --> 00:57:25.852 slowly varying signal and then as the

NOTE Confidence: 0.89782807

00:57:25.852 --> 00:57:29.078 time progresses the frequency is increasing.

NOTE Confidence: 0.89782807

00:57:29.080 --> 00:57:31.480 Then if you look at this

NOTE Confidence: 0.89782807

00:57:31.480 --> 00:57:33.760 wavelet coefficients here,

NOTE Confidence: 0.89782807

00:57:33.760 --> 00:57:37.655 so this vertical axis here is the

NOTE Confidence: 0.89782807

00:57:37.655 --> 00:57:40.013 frequency and the horizontal axis here

NOTE Confidence: 0.89782807

00:57:40.013 --> 00:57:41.720 is time and these are the coefficients.

NOTE Confidence: 0.89782807

00:57:41.720 --> 00:57:44.205 So if you can see this frequency

NOTE Confidence: 0.89782807

00:57:44.205 --> 00:57:45.879 is increasing slowly in time.

NOTE Confidence: 0.89782807

00:57:45.880 --> 00:57:49.072 So this transform is not only giving

NOTE Confidence: 0.89782807

00:57:49.072 --> 00:57:51.852 us information about what frequency

NOTE Confidence: 0.89782807

00:57:51.852 --> 00:57:53.768 components are present in your

NOTE Confidence: 0.89782807

00:57:53.768 --> 00:57:56.174 data and but it is also telling

NOTE Confidence: 0.89782807

00:57:56.174 --> 00:57:59.373 us that at what time windows these

NOTE Confidence: 0.89782807

00:57:59.373 --> 00:58:01.240 frequency components are present.

NOTE Confidence: 0.89782807

00:58:01.240 --> 00:58:02.955 And these coefficients are just

NOTE Confidence: 0.89782807

00:58:02.955 --> 00:58:04.670 computed by just aligning these

NOTE Confidence: 0.89782807

00:58:04.733 --> 00:58:06.476 we have list to the time window,

NOTE Confidence: 0.89782807

00:58:06.480 --> 00:58:08.775 computing these inner product or

NOTE Confidence: 0.89782807

00:58:08.775 --> 00:58:10.611 correlations and just plotting

NOTE Confidence: 0.89782807

00:58:10.611 --> 00:58:12.119 it here together.

NOTE Confidence: 0.89782807

00:58:12.120 --> 00:58:15.156 And if you plot this same

NOTE Confidence: 0.89782807

00:58:15.160 --> 00:58:17.360 coefficients with chirp down signal,

NOTE Confidence: 0.89782807

00:58:17.360 --> 00:58:19.688 it's going to look exactly the

NOTE Confidence: 0.89782807

00:58:19.688 --> 00:58:21.760 opposite and you can identify

NOTE Confidence: 0.89782807

00:58:21.760 --> 00:58:23.860 the OR discriminate between the

NOTE Confidence: 0.89782807

00:58:23.860 --> 00:58:27.998 two different kind of signals.

NOTE Confidence: 0.89782807

00:58:28.000 --> 00:58:31.198 Any questions here?

NOTE Confidence: 0.89782807

00:58:31.200 --> 00:58:33.160 Then we move to the last part.

NOTE Confidence: 0.70217913

00:58:45.910 --> 00:58:46.270 OK,

NOTE Confidence: 0.91511804

00:58:51.870 --> 00:58:53.806 so now we have a little bit idea

NOTE Confidence: 0.91511804

00:58:53.806 --> 00:58:55.310 about what these wavelets are.

NOTE Confidence: 0.91511804

00:58:55.310 --> 00:58:58.229 So why we came to this wavelet?

NOTE Confidence: 0.91511804

00:58:58.230 --> 00:59:00.606 Because this Fourier transform

NOTE Confidence: 0.91511804

00:59:00.606 --> 00:59:03.287 doesn't tell us where exactly in

NOTE Confidence: 0.91511804

00:59:03.287 --> 00:59:05.380 time and that that can be very

NOTE Confidence: 0.91511804

00:59:05.448 --> 00:59:07.525 important information sometime you

NOTE Confidence: 0.91511804

00:59:07.525 --> 00:59:08.680 don't want to miss out on that,

NOTE Confidence: 0.91511804

00:59:08.680 --> 00:59:09.880 where exactly it is present.

NOTE Confidence: 0.91511804

00:59:09.880 --> 00:59:13.706 So wavelets give you more kind of

NOTE Confidence: 0.91511804

00:59:13.706 --> 00:59:16.271 more powerful representation of the

NOTE Confidence: 0.91511804

00:59:16.271 --> 00:59:19.756 signal in time domain as well as now

NOTE Confidence: 0.91511804

00:59:19.760 --> 00:59:23.480 how do we kind of extend these this

NOTE Confidence: 0.91511804

00:59:23.480 --> 00:59:25.959 wavelet concept to graph domain?

NOTE Confidence: 0.91511804

00:59:25.960 --> 00:59:26.920 Can we do it?

NOTE Confidence: 0.929759392727273

00:59:32.080 --> 00:59:35.167 So I'm not going to talk much

NOTE Confidence: 0.929759392727273

00:59:35.167 --> 00:59:37.000 about this mathematics here,

NOTE Confidence: 0.929759392727273

00:59:37.000 --> 00:59:42.232 not to scare you, but it is just a

NOTE Confidence: 0.929759392727273

00:59:42.232 --> 00:59:45.439 little bit that this is just a shifting.

NOTE Confidence: 0.929759392727273

00:59:45.440 --> 00:59:47.680 So there is a mother wavelet, sigh.

NOTE Confidence: 0.929759392727273

00:59:47.680 --> 00:59:51.040 And this mother wavelet can be

NOTE Confidence: 0.929759392727273

00:59:51.040 --> 00:59:54.040 hare debash or Mexican head.

NOTE Confidence: 0.929759392727273

00:59:54.040 --> 00:59:55.840 This is a mother wavelet.

NOTE Confidence: 0.929759392727273

00:59:55.840 --> 00:59:59.387 And then you kind of shift those wavelets,

NOTE Confidence: 0.929759392727273

00:59:59.387 --> 01:00:01.432 stretch those wavelets and compress

NOTE Confidence: 0.929759392727273

01:00:01.432 --> 01:00:03.746 those wavelets and and then kind

NOTE Confidence: 0.929759392727273

01:00:03.746 --> 01:00:06.278 of save those wavelets at different

NOTE Confidence: 0.929759392727273

01:00:06.278 --> 01:00:08.474 scales and different translations

NOTE Confidence: 0.929759392727273

01:00:08.474 --> 01:00:10.616 of different shifts.

NOTE Confidence: 0.929759392727273

01:00:10.616 --> 01:00:12.920 So these shifts, yes,

NOTE Confidence: 0.929759392727273

01:00:12.920 --> 01:00:13.560 we have a question.

NOTE Confidence: 0.864193354

01:00:14.240 --> 01:00:16.200 So again, just to clarify,

NOTE Confidence: 0.864193354

01:00:16.200 --> 01:00:18.360 basically we're doing the same stuff.

NOTE Confidence: 0.864193354

01:00:18.360 --> 01:00:21.504 We are coming up with a

NOTE Confidence: 0.864193354

01:00:21.504 --> 01:00:23.600 set of base functions.

NOTE Confidence: 0.864193354

01:00:23.600 --> 01:00:26.960 So then we can decompose our complex

NOTE Confidence: 0.864193354

01:00:26.960 --> 01:00:30.440 signal into all these base functions.

NOTE Confidence: 0.864193354

01:00:30.440 --> 01:00:33.770 And the whole creativity here is

NOTE Confidence: 0.864193354

01:00:33.770 --> 01:00:37.920 that we are trying to not just

NOTE Confidence: 0.864193354

01:00:37.920 --> 01:00:40.779 stay in one-dimensional signal,

NOTE Confidence: 0.864193354

01:00:40.779 --> 01:00:43.533 but we want to have higher

NOTE Confidence: 0.864193354

01:00:43.533 --> 01:00:45.800 sensitivity to spatial structure,

NOTE Confidence: 0.864193354

01:00:45.800 --> 01:00:47.768 we want to have higher sensitivity

NOTE Confidence: 0.864193354

01:00:47.768 --> 01:00:48.752 to temporal structure.

NOTE Confidence: 0.864193354

01:00:48.760 --> 01:00:51.556 And that's why we're coming up

NOTE Confidence: 0.864193354

01:00:51.556 --> 01:00:53.496 with more complex combination

NOTE Confidence: 0.864193354

01:00:53.496 --> 01:00:55.893 set of base functions, right.

NOTE Confidence: 0.864193354

01:00:55.893 --> 01:00:57.158 So that's the story here.

NOTE Confidence: 0.864193354

01:00:57.760 --> 01:00:59.080 OK,

NOTE Confidence: 0.732691245714286

01:00:59.080 --> 01:01:01.440 that's a, that's a very good point. Yeah. And

NOTE Confidence: 0.805610183

01:01:04.960 --> 01:01:07.060 yeah, so this similar basically we

NOTE Confidence: 0.805610183

01:01:07.060 --> 01:01:09.709 are trying to get the represent the

NOTE Confidence: 0.805610183

01:01:09.709 --> 01:01:11.664 signal using this different set

NOTE Confidence: 0.805610183

01:01:11.664 --> 01:01:13.609 of basis functions or harmonics

NOTE Confidence: 0.805610183

01:01:13.609 --> 01:01:16.157 or whatever if you can call it.

NOTE Confidence: 0.805610183

01:01:16.160 --> 01:01:19.856 And this side is our mother wavelet

NOTE Confidence: 0.805610183

01:01:19.856 --> 01:01:24.080 and this S acts as a scaling operation.

NOTE Confidence: 0.805610183

01:01:24.080 --> 01:01:28.280 So if you put S as like two or four,

NOTE Confidence: 0.805610183

01:01:28.280 --> 01:01:30.734 so usually people use dyadic wavelets

NOTE Confidence: 0.805610183

01:01:30.734 --> 01:01:33.126 like usually this is scaling factor

NOTE Confidence: 0.805610183

01:01:33.126 --> 01:01:35.490 is in powers of two because you

NOTE Confidence: 0.805610183

01:01:35.490 --> 01:01:37.440 cannot like it's in continuous form.

NOTE Confidence: 0.805610183

01:01:37.440 --> 01:01:41.048 So you cannot like store infinite

NOTE Confidence: 0.805610183

01:01:41.048 --> 01:01:42.440 number of wavelets.

NOTE Confidence: 0.805610183

01:01:42.440 --> 01:01:45.176 You can store or analyze only a finite

NOTE Confidence: 0.805610183

01:01:45.176 --> 01:01:47.998 number of wavelets in in your computer.

NOTE Confidence: 0.805610183

01:01:48.000 --> 01:01:50.880 So you kind of discretize your scales

NOTE Confidence: 0.805610183

01:01:50.880 --> 01:01:52.880 and you discretize your translating

NOTE Confidence: 0.805610183

01:01:52.880 --> 01:01:54.853 to different time windows and

NOTE Confidence: 0.805610183

01:01:54.853 --> 01:01:56.077 you fix those numbers.

NOTE Confidence: 0.805610183

01:01:56.080 --> 01:01:59.280 And once you have fixed those bases,

NOTE Confidence: 0.805610183

01:01:59.280 --> 01:02:01.165 you can compute the coefficients

NOTE Confidence: 0.805610183

01:02:01.165 --> 01:02:02.673 corresponding to the corresponding

NOTE Confidence: 0.805610183

01:02:02.673 --> 01:02:04.258 to different wavelets at

NOTE Confidence: 0.805610183

01:02:04.258 --> 01:02:05.794 different scales and shifts.

NOTE Confidence: 0.8376716

01:02:08.200 --> 01:02:12.240 Then similarly, is there any limit on

NOTE Confidence: 0.900184521

01:02:12.240 --> 01:02:15.920 the the how much data you need to

NOTE Confidence: 0.900184521

01:02:15.920 --> 01:02:18.360 have for what is the relationship?

NOTE Confidence: 0.900184521

01:02:18.360 --> 01:02:19.573 Just intuitively, again,

NOTE Confidence: 0.900184521

01:02:19.573 --> 01:02:22.291 when we are thinking about more

NOTE Confidence: 0.900184521

01:02:22.291 --> 01:02:24.320 simple analysis, we also have to

NOTE Confidence: 0.900184521

01:02:24.320 --> 01:02:26.107 think in terms of power, right?

NOTE Confidence: 0.900184521

01:02:26.107 --> 01:02:29.203 So this is how sensitive we are to

NOTE Confidence: 0.900184521

01:02:29.203 --> 01:02:32.119 changes versus how large our sample is.

NOTE Confidence: 0.900184521

01:02:32.120 --> 01:02:35.153 This seems to be a very

NOTE Confidence: 0.900184521

01:02:35.153 --> 01:02:37.718 different in terms of intuition,

NOTE Confidence: 0.900184521

01:02:37.720 --> 01:02:40.678 but are there any basic rules

NOTE Confidence: 0.900184521

01:02:40.678 --> 01:02:43.400 like how many functions can you

NOTE Confidence: 0.900184521

01:02:43.400 --> 01:02:45.450 incorporate versus how much data you

NOTE Confidence: 0.900184521

01:02:45.450 --> 01:02:47.040 need to have like anything at all?

NOTE Confidence: 0.900184521

01:02:47.040 --> 01:02:47.400 Very, very,

NOTE Confidence: 0.79503767

01:02:49.000 --> 01:02:51.238 very good, very good question here.

NOTE Confidence: 0.79503767

01:02:51.240 --> 01:02:54.726 So it's up to the user,

NOTE Confidence: 0.79503767

01:02:54.726 --> 01:02:56.556 researcher, whatever you call it,

NOTE Confidence: 0.855268286666667

01:02:58.600 --> 01:03:02.488 to kind of find this by intuition

NOTE Confidence: 0.855268286666667

01:03:02.488 --> 01:03:05.774 or kind of learn it by applying this

NOTE Confidence: 0.855268286666667

01:03:05.774 --> 01:03:08.359 wavelets to multiple data sets.

NOTE Confidence: 0.855268286666667

01:03:08.360 --> 01:03:10.004 So once you have some kind

NOTE Confidence: 0.855268286666667

01:03:10.004 --> 01:03:11.760 of sense of the data set,

NOTE Confidence: 0.855268286666667

01:03:11.760 --> 01:03:14.118 what kind of data it is,

NOTE Confidence: 0.855268286666667

01:03:14.120 --> 01:03:15.936 you can always use.

NOTE Confidence: 0.855268286666667

01:03:15.936 --> 01:03:18.660 You can start with like two

NOTE Confidence: 0.855268286666667

01:03:18.756 --> 01:03:21.491 or three different scales and

NOTE Confidence: 0.855268286666667

01:03:21.491 --> 01:03:23.608 like at different time shifts,

NOTE Confidence: 0.855268286666667

01:03:23.608 --> 01:03:27.042 like a very few time shifts in a time

NOTE Confidence: 0.855268286666667

01:03:27.042 --> 01:03:29.373 window depending on how long is the

NOTE Confidence: 0.855268286666667

01:03:29.454 --> 01:03:31.603 recording versus how much variation.

NOTE Confidence: 0.855268286666667

01:03:31.603 --> 01:03:33.850 So it is up to the user

NOTE Confidence: 0.855268286666667

01:03:33.927 --> 01:03:35.717 to kind of determine it.

NOTE Confidence: 0.855268286666667

01:03:35.720 --> 01:03:38.100 But if you have like if your

NOTE Confidence: 0.855268286666667

01:03:38.100 --> 01:03:39.120 complexity is increased,

NOTE Confidence: 0.855268286666667

01:03:39.120 --> 01:03:41.160 because if you take more and more wavelets,

NOTE Confidence: 0.855268286666667

01:03:41.160 --> 01:03:43.190 your resolution is going to increase and

NOTE Confidence: 0.855268286666667

01:03:43.190 --> 01:03:45.678 that puts a computational burden on you.

NOTE Confidence: 0.855268286666667

01:03:45.680 --> 01:03:47.598 So you have to like kind of

NOTE Confidence: 0.855268286666667

01:03:47.598 --> 01:03:48.972 compromise between the computational

NOTE Confidence: 0.855268286666667

01:03:48.972 --> 01:03:51.288 burden versus the data set that

NOTE Confidence: 0.855268286666667

01:03:51.288 --> 01:03:53.760 you are that you have in hand.

NOTE Confidence: 0.855268286666667

01:03:53.760 --> 01:03:55.520 So there is no hard and fast rule.

NOTE Confidence: 0.855268286666667

01:03:55.520 --> 01:03:57.272 How many scales or how many

NOTE Confidence: 0.855268286666667

01:03:57.272 --> 01:03:59.080 shifts you have to use here.

NOTE Confidence: 0.855268286666667

01:03:59.080 --> 01:04:01.754 It comes from just from the application.

NOTE Confidence: 0.855268286666667

01:04:01.760 --> 01:04:02.120 Yeah,

NOTE Confidence: 0.5553504

01:04:05.200 --> 01:04:05.440 OK.

NOTE Confidence: 0.873147588

01:04:08.400 --> 01:04:10.200 So this is the wavelet.

NOTE Confidence: 0.873147588

01:04:10.200 --> 01:04:12.216 So in classical domain again this way

NOTE Confidence: 0.873147588

01:04:12.216 --> 01:04:14.518 you can define this wavelet coefficients

NOTE Confidence: 0.942589566

01:04:16.560 --> 01:04:19.880 and then using this intuition

NOTE Confidence: 0.942589566

01:04:19.880 --> 01:04:21.680 of this because this complex.

NOTE Confidence: 0.942589566

01:04:21.680 --> 01:04:24.230 So whenever you see exponential of

NOTE Confidence: 0.942589566

01:04:24.230 --> 01:04:28.596 J Omega that is going to be our

NOTE Confidence: 0.942589566

01:04:28.596 --> 01:04:30.354 eigenvector Laplacian eigenvector.

NOTE Confidence: 0.942589566

01:04:30.360 --> 01:04:33.128 So or even if you see sine sine

NOTE Confidence: 0.942589566

01:04:33.128 --> 01:04:36.263 wave or sine Omega Omega X or cosine

NOTE Confidence: 0.942589566

01:04:36.263 --> 01:04:39.664 Omega X that is literally analogous

NOTE Confidence: 0.942589566

01:04:39.664 --> 01:04:43.440 to our Laplacian eigenvectors.

NOTE Confidence: 0.942589566

01:04:43.440 --> 01:04:45.000 And if you have so here,

NOTE Confidence: 0.942589566

01:04:45.000 --> 01:04:47.952 if you have this A a means the shift.

NOTE Confidence: 0.942589566

01:04:47.960 --> 01:04:51.038 So that is nothing.

NOTE Confidence: 0.570994058

01:04:53.080 --> 01:04:56.860 Oh, this Phi dash, sorry, this side side

NOTE Confidence: 0.570994058

01:04:56.860 --> 01:05:00.080 hat is nothing but this filter here.

NOTE Confidence: 0.570994058

01:05:00.080 --> 01:05:02.520 So I don't want to get into the details here,

NOTE Confidence: 0.570994058

01:05:02.520 --> 01:05:05.537 but the point here is that we

NOTE Confidence: 0.570994058

01:05:05.537 --> 01:05:08.304 want to decompose our signal in

NOTE Confidence: 0.570994058

01:05:08.304 --> 01:05:11.160 in terms of different scales and

NOTE Confidence: 0.570994058

01:05:11.248 --> 01:05:13.798 different shifts in the space.

NOTE Confidence: 0.570994058

01:05:13.800 --> 01:05:16.476 So what exactly is happening here?

NOTE Confidence: 0.570994058

01:05:16.480 --> 01:05:21.316 So we define some kernels for this wavelet.

NOTE Confidence: 0.570994058

01:05:21.320 --> 01:05:23.120 So we don't have to worry about that.

NOTE Confidence: 0.570994058

01:05:23.120 --> 01:05:24.877 If you put it in the toolbox,

NOTE Confidence: 0.570994058

01:05:24.880 --> 01:05:28.684 it just you just need to decide the scales,

NOTE Confidence: 0.570994058

01:05:28.684 --> 01:05:32.826 how many scales you want in the graph

NOTE Confidence: 0.570994058

01:05:32.826 --> 01:05:35.804 and at each node it is computed.

NOTE Confidence: 0.570994058

01:05:35.804 --> 01:05:38.120 So here this is just a ring graph.

NOTE Confidence: 0.570994058

01:05:38.120 --> 01:05:41.203 Here if you look at wavelet at at

NOTE Confidence: 0.570994058

01:05:41.203 --> 01:05:44.399 a scale T is equal to 19 or 20.

NOTE Confidence: 0.570994058

01:05:44.400 --> 01:05:49.278 So if you so high scale means low frequency

NOTE Confidence: 0.570994058

01:05:49.280 --> 01:05:52.238 and low scale means high frequency.

NOTE Confidence: 0.965352261666667

01:05:55.880 --> 01:05:58.651 So if you look at here the

NOTE Confidence: 0.965352261666667

01:05:58.651 --> 01:06:00.397 changes are going to be rapid

NOTE Confidence: 0.971886326666667

01:06:02.480 --> 01:06:07.124 in case of wavelet as low scales and high

NOTE Confidence: 0.971886326666667

01:06:07.124 --> 01:06:10.040 scales the changes are going to be slow.

NOTE Confidence: 0.971886326666667

01:06:10.040 --> 01:06:13.718 But let's not get into mathematics,

NOTE Confidence: 0.971886326666667

01:06:13.720 --> 01:06:15.764 but just intuition wise.

NOTE Confidence: 0.971886326666667

01:06:15.764 --> 01:06:18.319 High scale is low frequency,

NOTE Confidence: 0.971886326666667

01:06:18.320 --> 01:06:22.760 low scale is high frequency information. And

NOTE Confidence: 0.8157095

01:06:25.200 --> 01:06:25.880 now

NOTE Confidence: 0.6187192

01:06:28.880 --> 01:06:29.480 in a word,

NOTE Confidence: 0.71374172

01:06:32.640 --> 01:06:34.090 in this workshop, we are

NOTE Confidence: 0.71374172

01:06:34.090 --> 01:06:35.920 remember that we, we were, we are

NOTE Confidence: 0.818695838

01:06:38.120 --> 01:06:41.648 developing the concepts required for this

NOTE Confidence: 0.818695838

01:06:41.648 --> 01:06:44.000 geometric scattering trajectory homology.

NOTE Confidence: 0.818695838

01:06:44.000 --> 01:06:48.236 So it involves these four steps,

NOTE Confidence: 0.818695838

01:06:48.240 --> 01:06:48.990 graph creation.

NOTE Confidence: 0.818695838

01:06:48.990 --> 01:06:51.240 And once you have a graph,

NOTE Confidence: 0.818695838

01:06:51.240 --> 01:06:54.040 you, you want to represent

NOTE Confidence: 0.818695838

01:06:54.040 --> 01:06:56.264 this graph signal somehow.

NOTE Confidence: 0.818695838

01:06:56.264 --> 01:06:58.440 And this is scattering.

NOTE Confidence: 0.818695838

01:06:58.440 --> 01:07:00.108 This particular figure is

NOTE Confidence: 0.818695838

01:07:00.108 --> 01:07:02.692 for a **** graph scattering.

NOTE Confidence: 0.818695838

01:07:02.692 --> 01:07:05.356 And in the next,

NOTE Confidence: 0.818695838

01:07:05.360 --> 01:07:07.188 in the next lecture,

NOTE Confidence: 0.818695838

01:07:07.188 --> 01:07:10.030 Dhananjay and Brian are going to

NOTE Confidence: 0.818695838

01:07:10.030 --> 01:07:13.460 talk about how to use these graph

NOTE Confidence: 0.818695838

01:07:13.573 --> 01:07:17.227 wavelets in order to define exactly

NOTE Confidence: 0.818695838

01:07:17.227 --> 01:07:19.679 this graph scattering transform.

NOTE Confidence: 0.818695838

01:07:19.679 --> 01:07:23.795 So we have, once we have this graph

NOTE Confidence: 0.818695838

01:07:23.795 --> 01:07:25.160 scattering transform defined,

NOTE Confidence: 0.818695838

01:07:25.160 --> 01:07:27.878 it goes to the dimensionality reduction.

NOTE Confidence: 0.818695838

01:07:27.880 --> 01:07:30.560 And then we do some kind of a

NOTE Confidence: 0.818695838

01:07:30.560 --> 01:07:34.060 topological data analysis to find

NOTE Confidence: 0.818695838

01:07:34.060 --> 01:07:36.710 the find the shapes that that

NOTE Confidence: 0.818695838

01:07:36.710 --> 01:07:39.120 come come across from this data.

NOTE Confidence: 0.818695838

01:07:39.120 --> 01:07:40.996 And in the first in last week,

NOTE Confidence: 0.818695838

01:07:41.000 --> 01:07:42.224 in the first lecture,

NOTE Confidence: 0.818695838

01:07:42.224 --> 01:07:43.754 Dhananjay has talked about this,

NOTE Confidence: 0.818695838

01:07:43.760 --> 01:07:46.040 how to deal with these shapes.

NOTE Confidence: 0.818695838

01:07:46.040 --> 01:07:48.360 Today in this step two,

NOTE Confidence: 0.818695838

01:07:48.360 --> 01:07:51.015 we have learned the concepts

NOTE Confidence: 0.818695838

01:07:51.015 --> 01:07:53.508 of frequency in graphs,

NOTE Confidence: 0.818695838

01:07:53.508 --> 01:07:57.078 frequency in in this irregular

NOTE Confidence: 0.818695838

01:07:57.078 --> 01:08:01.440 structure and we have we have

NOTE Confidence: 0.818695838

01:08:01.440 --> 01:08:04.640 gone through Fourier transform

NOTE Confidence: 0.818695838

01:08:04.640 --> 01:08:06.720 and its extension to graphs.

NOTE Confidence: 0.818695838

01:08:06.720 --> 01:08:09.540 And we also introduced concepts

NOTE Confidence: 0.818695838

01:08:09.540 --> 01:08:12.360 of wavelet a little bit.

NOTE Confidence: 0.818695838

01:08:12.360 --> 01:08:14.580 And this representations now are

NOTE Confidence: 0.818695838

01:08:14.580 --> 01:08:17.949 going to be based on this graph

NOTE Confidence: 0.818695838

01:08:17.949 --> 01:08:21.192 wavelets so that we get a richer

NOTE Confidence: 0.818695838

01:08:21.192 --> 01:08:23.544 representation of this data once

NOTE Confidence: 0.818695838

01:08:23.544 --> 01:08:25.804 we have the representation of

NOTE Confidence: 0.818695838

01:08:25.804 --> 01:08:28.680 the data at each time point.

NOTE Confidence: 0.818695838

01:08:28.680 --> 01:08:31.044 So we after we have this

NOTE Confidence: 0.818695838

01:08:31.044 --> 01:08:33.160 representation at each time point,

NOTE Confidence: 0.818695838

01:08:33.160 --> 01:08:35.095 we do this dimensionality reduction

NOTE Confidence: 0.818695838

01:08:35.095 --> 01:08:38.207 and we get a nice trajectory and

NOTE Confidence: 0.818695838

01:08:38.207 --> 01:08:40.776 then we kind of analyze the shape

NOTE Confidence: 0.818695838

01:08:40.776 --> 01:08:43.717 of that trajectory using this TDA.

NOTE Confidence: 0.442470995

01:08:48.120 --> 01:08:48.880 So, OK,

NOTE Confidence: 0.5297138

01:08:51.320 --> 01:08:53.475 so I'm going to just remind again what

NOTE Confidence: 0.5297138

01:08:53.475 --> 01:08:55.265 we have covered and then in my lecture

NOTE Confidence: 0.5297138

01:08:55.265 --> 01:08:57.239 here and if we have any more questions,

NOTE Confidence: 0.5297138

01:08:57.240 --> 01:08:59.400 we can go go through that.

NOTE Confidence: 0.5297138

01:08:59.400 --> 01:09:02.830 So we today we introduced graph signal

NOTE Confidence: 0.5297138

01:09:02.830 --> 01:09:07.114 processing where we and also we kind of,

NOTE Confidence: 0.5297138

01:09:07.120 --> 01:09:10.333 we made an analogy from classical signal

NOTE Confidence: 0.5297138

01:09:10.333 --> 01:09:12.879 processing and how to extend the,

NOTE Confidence: 0.5297138

01:09:12.880 --> 01:09:15.760 the important tools in graph domain.

NOTE Confidence: 0.5297138

01:09:15.760 --> 01:09:18.526 And we found that this Laplacian

NOTE Confidence: 0.5297138

01:09:18.526 --> 01:09:20.370 eigenvalues and eigenvectors are

NOTE Confidence: 0.5297138

01:09:20.450 --> 01:09:22.746 the basic building blocks of all

NOTE Confidence: 0.5297138

01:09:22.746 --> 01:09:24.276 the concepts in graph domain.

NOTE Confidence: 0.5297138

01:09:24.280 --> 01:09:25.584 Most of the conference,

NOTE Confidence: 0.5297138

01:09:25.584 --> 01:09:28.040 most of the concepts in graph domain.

NOTE Confidence: 0.5297138

01:09:28.040 --> 01:09:29.840 And we also introduced what is

NOTE Confidence: 0.5297138

01:09:29.840 --> 01:09:31.640 the use of the wavelet.

NOTE Confidence: 0.5297138

01:09:31.640 --> 01:09:34.808 And next is how do we use this graph

NOTE Confidence: 0.5297138

01:09:34.808 --> 01:09:37.360 wavelets to define graph scattering?

NOTE Confidence: 0.5297138

01:09:37.360 --> 01:09:39.360 And that that's not that's going to be

NOTE Confidence: 0.664584633333333

01:09:41.880 --> 01:09:46.424 presented by the Ranjay in next week lecture.

NOTE Confidence: 0.664584633333333

01:09:46.424 --> 01:09:52.384 And then we're going to combine these Step 2,

NOTE Confidence: 0.664584633333333

01:09:52.384 --> 01:09:55.755 all these steps together in the next

NOTE Confidence: 0.664584633333333

01:09:55.755 --> 01:09:58.170 lecture to give a picture of geometric

NOTE Confidence: 0.664584633333333

01:09:58.241 --> 01:10:00.240 scattering, scattering homology.

NOTE Confidence: 0.87561429

01:10:02.640 --> 01:10:04.626 Thank you. That ends my lecture

NOTE Confidence: 0.87561429

01:10:04.626 --> 01:10:05.839 today. Thanks for listening.

NOTE Confidence: 0.601720488

01:10:06.680 --> 01:10:08.720 Thank you, thank you, Rahul.

NOTE Confidence: 0.601720488

01:10:08.720 --> 01:10:11.107 So I want to give an opportunity

NOTE Confidence: 0.601720488

01:10:11.107 --> 01:10:13.720 to our our other participants.

NOTE Confidence: 0.601720488

01:10:13.720 --> 01:10:16.036 If you guys have any questions,

NOTE Confidence: 0.601720488

01:10:16.040 --> 01:10:18.532 I can see that you're sticking around

NOTE Confidence: 0.601720488

01:10:18.532 --> 01:10:22.800 through all this advanced stuff.

NOTE Confidence: 0.601720488

01:10:22.800 --> 01:10:24.768 Any any comments,

NOTE Confidence: 0.601720488

01:10:24.768 --> 01:10:28.040 any questions before I will jump in.

NOTE Confidence: 0.907462621818182

01:10:33.920 --> 01:10:36.170 So for me, for me it's

NOTE Confidence: 0.907462621818182

01:10:36.170 --> 01:10:38.640 the same kind of question.

NOTE Confidence: 0.907462621818182

01:10:38.640 --> 01:10:40.304 My understanding from today,

NOTE Confidence: 0.907462621818182

01:10:40.304 --> 01:10:42.800 I was I was really curious

NOTE Confidence: 0.907462621818182

01:10:42.884 --> 01:10:45.416 about this approach and I was

NOTE Confidence: 0.907462621818182

01:10:45.416 --> 01:10:47.760 wondering what is the backbone.

NOTE Confidence: 0.907462621818182

01:10:47.760 --> 01:10:50.592 And again, the way I understand

NOTE Confidence: 0.907462621818182

01:10:50.592 --> 01:10:56.200 that is that we are after some

NOTE Confidence: 0.907462621818182

01:10:56.200 --> 01:11:00.400 functional and practical decomposition

NOTE Confidence: 0.907462621818182

01:11:00.400 --> 01:11:03.555 of very complex phenomena into

NOTE Confidence: 0.907462621818182

01:11:03.555 --> 01:11:07.008 something that we can manage,

NOTE Confidence: 0.907462621818182

01:11:07.008 --> 01:11:09.792 something we understand.

NOTE Confidence: 0.907462621818182

01:11:09.792 --> 01:11:13.445 And we just going from say

NOTE Confidence: 0.907462621818182

01:11:13.445 --> 01:11:18.426 Taylor to Fourier to graph based

NOTE Confidence: 0.907462621818182

01:11:18.426 --> 01:11:20.878 decomposition and then aiding

NOTE Confidence: 0.907462621818182

01:11:20.878 --> 01:11:22.717 this temporal component.

NOTE Confidence: 0.907462621818182

01:11:22.720 --> 01:11:26.544 So we just generating more

NOTE Confidence: 0.907462621818182

01:11:26.544 --> 01:11:28.920 complex base functions.

NOTE Confidence: 0.950298148

01:11:31.600 --> 01:11:34.051 This is new to me, but in my

NOTE Confidence: 0.950298148

01:11:34.051 --> 01:11:35.559 intuition comes from this,

NOTE Confidence: 0.950298148

01:11:35.560 --> 01:11:38.240 you know, single dimensional story.

NOTE Confidence: 0.826706301666667

01:11:40.480 --> 01:11:44.476 What is the danger with overheating?

NOTE Confidence: 0.826706301666667

01:11:44.480 --> 01:11:49.297 Does it exist? Overheating like can be

NOTE Confidence: 0.826706301666667

01:11:49.297 --> 01:11:52.051 overheat because that would be concerned

NOTE Confidence: 0.826706301666667

01:11:52.051 --> 01:11:54.960 with more simple decompositions, right?

NOTE Confidence: 0.842704062727273

01:11:57.200 --> 01:12:00.218 You do understand correctly that all

NOTE Confidence: 0.842704062727273

01:12:00.218 --> 01:12:04.305 data is treated as signal or are there

NOTE Confidence: 0.842704062727273

01:12:04.305 --> 01:12:07.155 any components to denoise the process,

NOTE Confidence: 0.842704062727273

01:12:07.160 --> 01:12:09.720 right, to separate noisy components?

NOTE Confidence: 0.842704062727273

01:12:09.720 --> 01:12:10.920 Like for instance,

NOTE Confidence: 0.842704062727273

01:12:10.920 --> 01:12:13.320 you just assume that something of

NOTE Confidence: 0.842704062727273

01:12:13.320 --> 01:12:16.120 higher frequencies and noise, I don't know.

NOTE Confidence: 0.842704062727273

01:12:16.120 --> 01:12:18.280 So again, this is me just trying

NOTE Confidence: 0.842704062727273

01:12:18.280 --> 01:12:19.850 to digest this new material.

NOTE Confidence: 0.842704062727273

01:12:19.850 --> 01:12:21.400 Like what is the signal?

NOTE Confidence: 0.842704062727273

01:12:21.400 --> 01:12:22.873 What is noise?

NOTE Confidence: 0.842704062727273

01:12:22.873 --> 01:12:24.837 Are there any criteria?

NOTE Confidence: 0.842704062727273

01:12:24.840 --> 01:12:26.615 And what is the danger

NOTE Confidence: 0.842704062727273

01:12:26.615 --> 01:12:27.680 of overheating generally?

NOTE Confidence: 0.781565643333333

01:12:28.920 --> 01:12:31.398 Yeah. So talking about frequency first,

NOTE Confidence: 0.781565643333333

01:12:31.400 --> 01:12:33.880 a high frequencies are not always the noise

NOTE Confidence: 0.58775102

01:12:34.200 --> 01:12:36.440 exactly. Yeah, right. But in

NOTE Confidence: 0.83228996

01:12:36.440 --> 01:12:39.632 most of the cases, most of the cases

NOTE Confidence: 0.83228996

01:12:39.632 --> 01:12:42.417 usually you don't need high frequency

NOTE Confidence: 0.83228996

01:12:42.417 --> 01:12:46.164 because in general the date the in the

NOTE Confidence: 0.83228996

01:12:46.164 --> 01:12:50.720 natural data is smooth in nature seizure.

NOTE Confidence: 0.83228996

01:12:50.720 --> 01:12:52.240 Yeah, Yeah. So the best one is not,

NOTE Confidence: 0.83228996

01:12:52.240 --> 01:12:54.544 not, not all the time,

NOTE Confidence: 0.83228996

01:12:54.544 --> 01:12:56.200 Yeah, not all the time,

NOTE Confidence: 0.83228996

01:12:56.200 --> 01:12:57.560 but they're important sometimes.

NOTE Confidence: 0.83228996

01:12:57.560 --> 01:13:00.836 And over fitting is like sometimes you

NOTE Confidence: 0.83228996

01:13:00.836 --> 01:13:04.860 can kind of take so many scales here

NOTE Confidence: 0.83228996

01:13:04.860 --> 01:13:08.773 and so many kind of so many levels

NOTE Confidence: 0.83228996

01:13:08.773 --> 01:13:11.359 of decomposition and that kind of

NOTE Confidence: 0.737157356666667

01:13:13.440 --> 01:13:16.278 that that can like create issues.

NOTE Confidence: 0.737157356666667

01:13:16.280 --> 01:13:18.352 If you kind of take more and more

NOTE Confidence: 0.737157356666667

01:13:18.352 --> 01:13:20.557 scales and make it more complicated

NOTE Confidence: 0.737157356666667

01:13:20.557 --> 01:13:22.993 representation in this particular part here.

NOTE Confidence: 0.737157356666667

01:13:23.000 --> 01:13:24.775 Then again that that has

NOTE Confidence: 0.737157356666667

01:13:24.775 --> 01:13:26.484 another kind of a set.

NOTE Confidence: 0.737157356666667

01:13:26.484 --> 01:13:28.756 It can the set back can happen again.

NOTE Confidence: 0.737157356666667

01:13:28.760 --> 01:13:29.680 But but The thing is,

NOTE Confidence: 0.737157356666667

01:13:29.680 --> 01:13:32.928 if you're not able to kind of find out

NOTE Confidence: 0.737157356666667

01:13:32.928 --> 01:13:35.040 the nice patterns and all in the data,

NOTE Confidence: 0.737157356666667

01:13:35.040 --> 01:13:38.215 if you take two computationally

NOTE Confidence: 0.737157356666667

01:13:38.215 --> 01:13:40.120 complex methodology here,

NOTE Confidence: 0.737157356666667

01:13:40.120 --> 01:13:41.836 then you can always reduce it.

NOTE Confidence: 0.632596745

01:13:43.760 --> 01:13:45.902 Still let me yeah, let me try

NOTE Confidence: 0.632596745

01:13:45.902 --> 01:13:47.800 ask this question different way.

NOTE Confidence: 0.632596745

01:13:47.800 --> 01:13:51.824 So if I do something basic as principle

NOTE Confidence: 0.632596745

01:13:51.824 --> 01:13:54.154 component analysis on imaging data,

NOTE Confidence: 0.632596745

01:13:54.160 --> 01:13:58.859 I will always expect that first

NOTE Confidence: 0.632596745

01:13:58.859 --> 01:14:01.554 largest components, maybe three or

NOTE Confidence: 0.632596745

01:14:01.554 --> 01:14:05.015 four are going to be noise, right?

NOTE Confidence: 0.632596745

01:14:05.015 --> 01:14:07.640 So this is my approach to denoising

NOTE Confidence: 0.632596745

01:14:07.640 --> 01:14:11.994 my data and I'm just curious if

NOTE Confidence: 0.62216018

01:14:15.120 --> 01:14:18.168 there is any built in intuitional

NOTE Confidence: 0.62216018

01:14:18.168 --> 01:14:21.264 logic except for high frequencies

NOTE Confidence: 0.62216018

01:14:21.264 --> 01:14:23.440 typically noise to denoising,

NOTE Confidence: 0.62216018

01:14:23.440 --> 01:14:26.080 we are dealing with incredibly noisy

NOTE Confidence: 0.62216018

01:14:26.080 --> 01:14:28.798 data when we do for instance fMRI.

NOTE Confidence: 0.62216018

01:14:28.800 --> 01:14:30.515 So we know that noise is there.

NOTE Confidence: 0.62216018

01:14:30.520 --> 01:14:34.280 So how do you so sometimes noise in

NOTE Confidence: 0.851337192142857

01:14:34.600 --> 01:14:35.920 this scattering transforms.

NOTE Confidence: 0.851337192142857

01:14:35.920 --> 01:14:38.120 There are some learnable parameters

NOTE Confidence: 0.851337192142857

01:14:38.120 --> 01:14:40.757 also they have people have introduced,

NOTE Confidence: 0.851337192142857

01:14:40.760 --> 01:14:43.000 I think Dhananjay was one of the

NOTE Confidence: 0.851337192142857

01:14:43.000 --> 01:14:44.554 contributors in the, in those,

NOTE Confidence: 0.851337192142857

01:14:44.554 --> 01:14:46.608 in those in that work where you

NOTE Confidence: 0.851337192142857

01:14:46.608 --> 01:14:48.596 can kind of learn from the data,

NOTE Confidence: 0.851337192142857

01:14:48.600 --> 01:14:49.760 how much of high frequency

NOTE Confidence: 0.851337192142857

01:14:49.760 --> 01:14:50.920 component I'm going to use,

NOTE Confidence: 0.851337192142857

01:14:50.920 --> 01:14:52.220 how much of low frequency

NOTE Confidence: 0.851337192142857

01:14:52.220 --> 01:14:53.520 component I'm going to use.

NOTE Confidence: 0.851337192142857

01:14:53.520 --> 01:14:56.120 So it's not a hard for hard and

NOTE Confidence: 0.851337192142857

01:14:56.120 --> 01:14:58.453 fast deterministic set of values or

NOTE Confidence: 0.851337192142857

01:14:58.453 --> 01:15:00.473 deterministic set of frequency windows.

NOTE Confidence: 0.851337192142857

01:15:00.480 --> 01:15:02.460 But you can make it learnable

NOTE Confidence: 0.851337192142857

01:15:02.460 --> 01:15:04.504 as well based on the data.

NOTE Confidence: 0.851337192142857

01:15:04.504 --> 01:15:06.234 So it can be data-driven

NOTE Confidence: 0.851337192142857

01:15:06.234 --> 01:15:08.158 learnable representation as well.

NOTE Confidence: 0.885135092222222

01:15:09.680 --> 01:15:11.876 Maybe I can add a little bit to that.

NOTE Confidence: 0.885135092222222

01:15:11.880 --> 01:15:15.848 So the denoising is built into all the

NOTE Confidence: 0.885135092222222

01:15:15.848 --> 01:15:18.092 steps really in the pipeline, right.

NOTE Confidence: 0.885135092222222

01:15:18.092 --> 01:15:20.365 So the, the way first step has to do

NOTE Confidence: 0.885135092222222

01:15:20.365 --> 01:15:22.634 with how we construct the graph and how

NOTE Confidence: 0.885135092222222

01:15:22.634 --> 01:15:24.960 we define like a signal on the graph.

NOTE Confidence: 0.703059965

01:15:25.200 --> 01:15:26.760 So you denoise at that level,

NOTE Confidence: 0.78617978

01:15:27.240 --> 01:15:29.448 there is denoising built into that

NOTE Confidence: 0.78617978

01:15:29.448 --> 01:15:31.864 level in the sense for example in this

NOTE Confidence: 0.78617978

01:15:31.864 --> 01:15:33.720 in this picture when we build a graph,

NOTE Confidence: 0.78617978

01:15:33.720 --> 01:15:35.754 we are our our nodes ourselves

NOTE Confidence: 0.78617978

01:15:35.754 --> 01:15:37.957 and our edges are cells that

NOTE Confidence: 0.78617978

01:15:37.957 --> 01:15:39.877 are adjacent to each other.

NOTE Confidence: 0.78617978

01:15:39.880 --> 01:15:41.301 And so when we define the time

NOTE Confidence: 0.78617978

01:15:41.301 --> 01:15:42.400 lapse signal on the graph,

NOTE Confidence: 0.78617978

01:15:42.400 --> 01:15:44.619 we are averaging the signal over each

NOTE Confidence: 0.78617978

01:15:44.619 --> 01:15:47.198 cell like the segmentation of the cell.

NOTE Confidence: 0.78617978

01:15:47.200 --> 01:15:50.476 When we build a graph from brain,

NOTE Confidence: 0.78617978

01:15:50.480 --> 01:15:52.640 then we consider a parcellation of

NOTE Confidence: 0.78617978

01:15:52.640 --> 01:15:55.080 the brain based off of some Atlas.

NOTE Confidence: 0.78617978

01:15:55.080 --> 01:15:56.599 And so to define a time lapse

NOTE Confidence: 0.78617978

01:15:56.599 --> 01:15:57.600 signal on that graph,

NOTE Confidence: 0.78617978

01:15:57.600 --> 01:15:59.958 we are averaging within each parcel.

NOTE Confidence: 0.78617978

01:15:59.960 --> 01:16:02.440 So there's some averaging happening

NOTE Confidence: 0.78617978

01:16:02.440 --> 01:16:04.939 in the walk cells belonging to each

NOTE Confidence: 0.78617978

01:16:04.939 --> 01:16:08.184 parcel and that has some small amount of

NOTE Confidence: 0.78617978

01:16:08.184 --> 01:16:10.920 denoising built into that averaging process.

NOTE Confidence: 0.78617978

01:16:10.920 --> 01:16:13.920 And then in the graph signal processing here,

NOTE Confidence: 0.78617978

01:16:13.920 --> 01:16:15.732 the wavelets that we are using

NOTE Confidence: 0.78617978

01:16:15.732 --> 01:16:18.022 to kind of capture the signal at

NOTE Confidence: 0.78617978

01:16:18.022 --> 01:16:19.717 different scales on the graph,

NOTE Confidence: 0.78617978

01:16:19.720 --> 01:16:22.123 the shape of the wavelet and,

NOTE Confidence: 0.78617978

01:16:22.123 --> 01:16:24.707 and how that wavelet is being applied to

NOTE Confidence: 0.78617978

01:16:24.707 --> 01:16:28.199 the data that does some amount of denoising.

NOTE Confidence: 0.78617978

01:16:28.200 --> 01:16:28.518 And so that,

NOTE Confidence: 0.810794172857143

01:16:28.680 --> 01:16:30.479 that's, that's what I was wondering actually.

NOTE Confidence: 0.810794172857143

01:16:30.480 --> 01:16:32.958 I was wondering if you have some

NOTE Confidence: 0.810794172857143

01:16:32.958 --> 01:16:34.948 wavelets that typically more more

NOTE Confidence: 0.810794172857143

01:16:34.948 --> 01:16:36.560 commonly represent noise that

NOTE Confidence: 0.810794172857143

01:16:36.560 --> 01:16:38.480 because of their shape or not.

NOTE Confidence: 0.810794172857143

01:16:38.480 --> 01:16:41.368 Like I have my toolbox and I know

NOTE Confidence: 0.810794172857143

01:16:41.368 --> 01:16:44.764 that in this corner I have noisier

NOTE Confidence: 0.810794172857143

01:16:44.764 --> 01:16:46.664 base functions like is it or,

NOTE Confidence: 0.810794172857143

01:16:46.664 --> 01:16:47.960 or it doesn't work like that.

NOTE Confidence: 0.794906938888889

01:16:48.720 --> 01:16:51.393 So when you pick a wavelet scale that's very,

NOTE Confidence: 0.794906938888889

01:16:51.400 --> 01:16:54.136 very small, then the frequency of

NOTE Confidence: 0.794906938888889

01:16:54.136 --> 01:16:57.000 the wavelet is much higher and,

NOTE Confidence: 0.794906938888889

01:16:57.000 --> 01:17:01.116 and that will correspond to very noisy.

NOTE Confidence: 0.794906938888889

01:17:01.120 --> 01:17:02.596 That's going to capture noise in,

NOTE Confidence: 0.794906938888889

01:17:02.600 --> 01:17:06.398 in the, in the data set on the graph.

NOTE Confidence: 0.794906938888889

01:17:06.400 --> 01:17:09.760 Again, depends on what what scale becomes

NOTE Confidence: 0.794906938888889

01:17:09.760 --> 01:17:12.918 noise versus what scale is still like,

NOTE Confidence: 0.794906938888889

01:17:12.920 --> 01:17:14.760 you know, an epileptic seizure,

NOTE Confidence: 0.794906938888889

01:17:14.760 --> 01:17:16.096 which is, you know,

NOTE Confidence: 0.794906938888889

01:17:16.096 --> 01:17:18.799 going to be rapidly wading across the brain.

NOTE Confidence: 0.794906938888889

01:17:18.800 --> 01:17:21.134 So, so there's denoising kind of

NOTE Confidence: 0.794906938888889

01:17:21.134 --> 01:17:23.436 in the spatial domain that's built

NOTE Confidence: 0.794906938888889

01:17:23.436 --> 01:17:25.796 into step one and Step 2 through

NOTE Confidence: 0.794906938888889

01:17:25.796 --> 01:17:27.932 averaging in step one and through

NOTE Confidence: 0.794906938888889

01:17:27.932 --> 01:17:30.559 the scales of the wavelets in Step 2.

NOTE Confidence: 0.794906938888889

01:17:30.560 --> 01:17:32.640 But that doesn't address

NOTE Confidence: 0.794906938888889

01:17:32.640 --> 01:17:35.240 denoising in the temporal domain.

NOTE Confidence: 0.794906938888889

01:17:35.240 --> 01:17:37.396 And so the denoising in the temporal

NOTE Confidence: 0.794906938888889

01:17:37.396 --> 01:17:39.910 domain is built into step three has to

NOTE Confidence: 0.794906938888889

01:17:39.910 --> 01:17:42.115 do with how that trajectory itself gets

NOTE Confidence: 0.794906938888889

01:17:42.115 --> 01:17:44.096 computed from all of these numerical

NOTE Confidence: 0.794906938888889

01:17:44.096 --> 01:17:46.159 features that are derived in Step 2.

NOTE Confidence: 0.794906938888889

01:17:46.160 --> 01:17:48.716 So the temporal denoising happens there.

NOTE Confidence: 0.794906938888889

01:17:48.720 --> 01:17:50.160 And then there's another denoising

NOTE Confidence: 0.794906938888889

01:17:50.160 --> 01:17:52.413 step in Step 4 where when we

NOTE Confidence: 0.794906938888889

01:17:52.413 --> 01:17:53.917 compute the topological signatures,

NOTE Confidence: 0.794906938888889

01:17:53.920 --> 01:17:55.225 we ignore topology,

NOTE Confidence: 0.794906938888889

01:17:55.225 --> 01:17:57.400 topology close to the diagonal.

NOTE Confidence: 0.794906938888889

01:17:57.400 --> 01:17:59.464 And so we are those are kind of

NOTE Confidence: 0.794906938888889

01:17:59.464 --> 01:18:00.862 the smaller topological features

NOTE Confidence: 0.794906938888889

01:18:00.862 --> 01:18:02.917 that are not as persistent.

NOTE Confidence: 0.794906938888889

01:18:02.920 --> 01:18:04.340 So there's again denoising

NOTE Confidence: 0.794906938888889

01:18:04.340 --> 01:18:05.760 happening at that level,

NOTE Confidence: 0.794906938888889

01:18:05.760 --> 01:18:06.628 but yeah,

NOTE Confidence: 0.794906938888889

01:18:06.628 --> 01:18:08.798 there's multiple levels of denoising.

NOTE Confidence: 0.794906938888889

01:18:08.800 --> 01:18:10.560 I'd say in step one and Step 2,

NOTE Confidence: 0.794906938888889

01:18:10.560 --> 01:18:12.558 the denoising operates on the spatial

NOTE Confidence: 0.794906938888889

01:18:12.558 --> 01:18:15.143 axis and step three and step four it

NOTE Confidence: 0.794906938888889

01:18:15.143 --> 01:18:17.318 operates on the temporal axis of the data.

NOTE Confidence: 0.958471965

01:18:18.920 --> 01:18:19.240 Thank you.

NOTE Confidence: 0.8437771175

01:18:21.440 --> 01:18:22.310 Very exciting. Looking

NOTE Confidence: 0.8437771175

01:18:22.310 --> 01:18:23.760 forward to the next talk.

NOTE Confidence: 0.878853604545455

01:18:23.760 --> 01:18:25.770 Yeah, thanks. Wanted to thank also

NOTE Confidence: 0.878853604545455

01:18:25.770 --> 01:18:27.560 the audience for sticking around.

NOTE Confidence: 0.878853604545455

01:18:27.560 --> 01:18:29.359 I know it was quite technical today,

NOTE Confidence: 0.878853604545455

01:18:29.360 --> 01:18:31.600 but it has really built in some

NOTE Confidence: 0.878853604545455

01:18:31.600 --> 01:18:33.276 vocabulary and some tools that

NOTE Confidence: 0.878853604545455

01:18:33.276 --> 01:18:35.232 would be very helpful in kind

NOTE Confidence: 0.878853604545455

01:18:35.232 --> 01:18:37.078 of bringing this all together.

NOTE Confidence: 0.878853604545455

01:18:37.080 --> 01:18:38.520 In our next workshop,

NOTE Confidence: 0.878853604545455

01:18:38.520 --> 01:18:40.320 we'll bring it all together,

NOTE Confidence: 0.878853604545455

01:18:40.320 --> 01:18:42.245 but we'll also show you lots and

NOTE Confidence: 0.878853604545455

01:18:42.245 --> 01:18:44.039 lots of examples of applying this

NOTE Confidence: 0.878853604545455

01:18:44.039 --> 01:18:45.857 technique to real data and build

NOTE Confidence: 0.878853604545455

01:18:45.857 --> 01:18:48.085 even more intuition around what

NOTE Confidence: 0.878853604545455

01:18:48.085 --> 01:18:52.195 kind of dynamics GSTH is capturing.

NOTE Confidence: 0.878853604545455

01:18:52.200 --> 01:18:53.550 And then we'll give you access

NOTE Confidence: 0.878853604545455

01:18:53.550 --> 01:18:55.425 to the tool and have you work on

NOTE Confidence: 0.878853604545455

01:18:55.425 --> 01:18:56.805 some data sets yourself in the

NOTE Confidence: 0.878853604545455

01:18:56.858 --> 01:18:58.278 final workshop in the series.

NOTE Confidence: 0.878853604545455

01:18:58.280 --> 01:18:59.240 So thanks for coming.

NOTE Confidence: 0.925540525

01:18:59.360 --> 01:19:00.280 Thanks for sticking around.

NOTE Confidence: 0.9690436425

01:19:01.040 --> 01:19:01.960 Thank you so much.

NOTE Confidence: 0.885253648571429

01:19:03.200 --> 01:19:06.000 Yes, thank you. See you next week.