WEBVTT

1 00:00:00.280 --> 00:00:01.760 <v Man>Good afternoon, everybody.</v>

2 00:00:01.760 --> 00:00:03.410 Good morning, Professor Holbrook.

3 00:00:04.520 --> 00:00:07.850 Today I'm honored to introduce Professor Andrew Holbrook.

4 00:00:07.850 --> 00:00:11.460 So professor Holbrook earned his bachelor's from UC Berkeley

5 00:00:11.460 --> 00:00:14.033 and a statistics masters and PhD from UC Irvine.

6 00:00:15.170 --> 00:00:16.930 His research touches a number of areas

7 00:00:16.930 --> 00:00:18.460 of biomedical interests,

8 00:00:18.460 --> 00:00:20.973 including Alzheimer's and epidemiology.

9 00:00:22.180 --> 00:00:23.690 He's currently an assistant professor

10 00:00:23.690 --> 00:00:27.280 of biostatistics at UCLA, where he teaches their advanced

11 00:00:27.280 --> 00:00:28.610 basic computer course.

12 00:00:28.610 --> 00:00:30.000 And he's the author of several pieces

13 00:00:30.000 --> 00:00:32.290 of scientific software.

14 00:00:32.290 --> 00:00:37.090 All of it, I think, is he's very fond of parallelization,

15 00:00:37.090 --> 00:00:40.330 and he also has a package including one on studying

16 00:00:40.330 --> 00:00:43.880 Hawkes processes, which he's going to tell us...

17 00:00:43.880 --> 00:00:45.990 Well, he's gonna tell us about the biological phenomenon

18 00:00:45.990 --> 00:00:48.012 and what's going on today.

19 00:00:48.012 --> 00:00:50.493 So Professor Holbrook, thank you so much.

20 00:00:51.500 --> 00:00:52.761 <v ->Okay, great.</v>

21 00:00:52.761 --> 00:00:57.170 Thank you so much for the kind invitation,

22 00:00:57.170 --> 00:01:01.990 and thanks for having me this morning slash afternoon.

23 00:01:01.990 --> 00:01:05.894 So today I'm actually gonna be kind of trying to present

24 00:01:05.894 --> 00:01:10.270 more of a high level talk that's gonna just focus on

25 00:01:10.270 --> 00:01:13.610 a couple of different problems that have

26 00:01:13.610 --> 00:01:18.140 come up when modeling Hawkes processes

27 00:01:18.140 --> 00:01:20.700 for public health data, and in particular

28 00:01:20.700 --> 00:01:22.563 for large scale public health data.

29 00:01:23.920 --> 00:01:27.630 So, today I'm interested in spatiotemporal data

30 00:01:27.630 --> 00:01:29.880 in public health, and this can take a number

31 00:01:29.880 --> 00:01:31.053 of different forms.

32 00:01:32.680 --> 00:01:37.680 So a great example of this is in Washington D.C.

33 00:01:38.500 --> 00:01:41.950 Here, I've got about 4,000 gunshots.

34 00:01:41.950 --> 00:01:43.530 You'll see this figure again,

35 00:01:43.530 --> 00:01:46.160 and I'll explain the colors to you

36 00:01:46.160 --> 00:01:48.640 and everything like that.

37 00:01:48.640 --> 00:01:52.930 But I just want you to see that in the year 2018 alone,

38 00:01:52.930 --> 00:01:56.890 there were 4,000 gunshots recorded in Washington DC.

39 00:01:56.890 --> 00:02:01.350 And this is just one example of really a gun violence

40 00:02:01.350 --> 00:02:03.923 problem in the U S of epidemic proportions.

41 00:02:07.483 --> 00:02:09.510 But spatiotemporal public health data

42 00:02:09.510 --> 00:02:11.210 can take on many forms.

43 00:02:11.210 --> 00:02:16.210 So here, for example, I have almost almost 3000 wildfires

44 00:02:18.290 --> 00:02:22.543 in Alaska between the years, 2015 and 2019.

45 00:02:23.810 --> 00:02:28.720 And this is actually just one piece of a larger

46 00:02:30.070 --> 00:02:32.363 trend that's going on in the American west.

47 00:02:34.810 --> 00:02:39.400 And then finally, another example spatiotemporal public

48 00:02:39.400 --> 00:02:43.720 health data is, and I believe that we don't need to spend

49 00:02:43.720 --> 00:02:45.650 too much time on this motivation,

50 00:02:45.650 --> 00:02:48.180 but it's the global spread of viruses.

51 00:02:48.180 --> 00:02:52.220 So for example, here, I've got 5,000 influenza cases

52 00:02:52.220 --> 00:02:56.290 recorded throughout, through 2000 to 2012.

53 00:02:57.750 --> 00:02:59.590 So if I want to model this data,

54 00:02:59.590 --> 00:03:02.210 what I'm doing is I'm modeling event data.

55 00:03:02.210 --> 00:03:06.417 And one of the classic models for doing so

56 00:03:06.417 --> 00:03:11.417 is really the canonical stochastic process here,

57 00:03:11.720 --> 00:03:14.240 in this context is, is the Poisson process.

58 00:03:14.240 --> 00:03:17.840 And I hope that you'll bear with me if we do just a little

59 00:03:17.840 --> 00:03:21.410 bit of review for our probability 101.

60 00:03:21.410 --> 00:03:23.810 But we say that accounting process

61 00:03:23.810 --> 00:03:27.659 is a homogeneous Poisson process, point process

62 00:03:27.659 --> 00:03:32.030 with rate parameter, excuse me, parameter lambda,

63 00:03:32.030 --> 00:03:34.160 which is greater than zero.

64 00:03:34.160 --> 00:03:37.823 If this process is always equal to zero at zero,

65 00:03:38.700 --> 00:03:42.780 if it's independent increments, excuse me,

66 00:03:42.780 --> 00:03:46.103 if it's increment over non-overlapping intervals

67 00:03:47.720 --> 00:03:49.880 are independent random variables.

68 00:03:49.880 --> 00:03:52.088 And then finally, if it's increments

69 00:03:52.088 --> 00:03:57.020 are Poisson distributed with mean given

70 00:03:57.020 --> 00:03:59.510 by that rate parameter lambda,

71 00:03:59.510 --> 00:04:01.943 and then the difference in the times.

72 00:04:03.890 --> 00:04:05.230 So we can make this model

73 00:04:06.866 --> 00:04:09.370 just a very little bit more complex.

74 00:04:09.370 --> 00:04:13.450 We can create an inhomogeneous Poisson point process,

75 00:04:13.450 --> 00:04:15.810 simply by saying that that rate parameter

76 00:04:15.810 --> 00:04:19.930 is no longer fixed, but itself is a function

77 00:04:19.930 --> 00:04:21.850 over the positive real line.

78 00:04:21.850 --> 00:04:24.010 And here everything is the exact same,

79 00:04:24.010 --> 00:04:27.620 except now we're saying that it's increments,

80 00:04:27.620 --> 00:04:29.930 it's differences over two different time periods

81 00:04:29.930 --> 00:04:34.930 are Poisson distributed, where now the mean is simply given

82 00:04:35.270 --> 00:04:39.840 by the definite integral over that interval.

83 00:04:39.840 --> 00:04:41.903 So we just integrate that rate function.

84 00:04:44.370 --> 00:04:45.499 Okay.

85 00:04:45.499 --> 00:04:47.800 So then how do we choose our rate function for the problems

86 00:04:47.800 --> 00:04:49.370 that we're interested in?

87 00:04:49.370 --> 00:04:53.410 Well, if we return to say the gun violence example,

88 00:04:53.410 --> 00:04:58.220 then it is plausible that at least sometimes some gun

89 00:04:58.220 --> 00:05:02.630 violence might precipitate more gun violence.

90 00:05:02.630 --> 00:05:07.513 So here we would say that having observed an event,

91 00:05:09.050 --> 00:05:11.730 having observed gunshots at a certain location

92 00:05:11.730 --> 00:05:14.820 at a certain time, we might expect that the probability

93 00:05:14.820 --> 00:05:19.820 of observing gunshots nearby and soon after is elevated,

94 00:05:23.480 --> 00:05:27.830 and the same could plausibly go for wildfires as well.

95 00:05:27.830 --> 00:05:32.830 It's that having observed a wildfire in a certain location,

96 00:05:33.400 --> 00:05:38.077 this could directly contribute to the existence

97 00:05:39.000 --> 00:05:42.310 or to the observation of other wildfires.

98 00:05:42.310 --> 00:05:45.350 So for example, this could happen by natural means.

99 00:05:45.350 --> 00:05:48.763 So we could have embers that are blown by the wind,

100 00:05:51.052 --> 00:05:53.850 or there could be a human that is in fact

101 00:05:53.850 --> 00:05:57.423 causing these wildfires, which is also quite common.

102 00:06:00.620 --> 00:06:02.900 And then it's not a stretch at all

103 00:06:02.900 --> 00:06:07.540 to believe that viral observation,

104 00:06:07.540 --> 00:06:11.870 so a child sick with influenza could precipitate

105 00:06:11.870 --> 00:06:16.190 another child that becomes sick with influenza

106 00:06:16.190 --> 00:06:18.993 in the same classroom and perhaps on the next day.

107 00:06:22.778 --> 00:06:27.440 So then, the solution to building this kind of dynamic into

108 00:06:27.440 --> 00:06:32.440 an in homogeneous Poisson process is simply to craft

109 00:06:32.790 --> 00:06:36.700 the rate function in a way that is asymmetric in time.

110 00:06:36.700 --> 00:06:40.523 So here is just a regular temporal Hawkes process.

111 00:06:43.418 --> 00:06:48.010 And what we do is we divide this rate function, lambda T,

112 00:06:48.010 --> 00:06:50.810 which I'm showing you in the bottom of the equation,

113 00:06:50.810 --> 00:06:55.430 into a background portion which is here.

114 00:06:55.430 --> 00:06:58.923 I denote nu, and this nu can be a function itself.

115 00:07:00.030 --> 00:07:04.090 And then we also have this self excitatory component C of T.

116 00:07:04.090 --> 00:07:08.110 And this self excitatory component for time T,

117 00:07:08.110 --> 00:07:13.110 it depends exclusively on observations

118 00:07:13.160 --> 00:07:15.880 that occur before time T.

119 00:07:16.832 --> 00:07:21.832 So each tn, where tn is less than T,

120 00:07:22.020 --> 00:07:25.000 are able to contribute information

121 00:07:25.000 --> 00:07:27.083 in some way to this process.

122 00:07:28.550 --> 00:07:31.970 And typically G is our triggering function.

123 00:07:31.970 --> 00:07:34.803 G is non increasing.

124 00:07:36.961 --> 00:07:39.520 And then the only other thing that we ask

125 00:07:39.520 --> 00:07:42.330 is that the different events contribute

126 00:07:42.330 --> 00:07:44.960 in an additive manner to the rate.

127 00:07:44.960 --> 00:07:48.840 So here, we've got the background rate in this picture,

128 00:07:48.840 --> 00:07:50.480 We have observation T1.

129 00:07:50.480 --> 00:07:51.993 The rate increases.

130 00:07:53.020 --> 00:07:54.640 It slowly decreases.

131 00:07:54.640 --> 00:07:57.350 We have another observation, the rate increases.

132 00:07:57.350 --> 00:07:59.900 And what you see is actually that after T1,

133 00:07:59.900 --> 00:08:04.330 we have a nice little bit of self excitation as it's termed,

134 00:08:04.330 --> 00:08:07.063 where we observe more observations.

135 00:08:08.940 --> 00:08:12.730 This model itself can be made just a little bit more complex

136 00:08:12.730 --> 00:08:14.430 if we add a spatial component.

137 00:08:14.430 --> 00:08:18.350 So here now, is the spatiotemporal Hawkes process

138 00:08:18.350 --> 00:08:22.440 where I'm simply showing you the background process,

139 00:08:22.440 --> 00:08:25.600 which now I'm allowing to be described

140 00:08:25.600 --> 00:08:29.000 by a rate function over space.

141 00:08:29.000 --> 00:08:32.260 And then, we also have the self excitatory component,

142 00:08:32.260 --> 00:08:34.890 which again, although it also involves

143 00:08:34.890 --> 00:08:36.780 a spatial component in it,

144 00:08:36.780 --> 00:08:39.550 it still has this asymmetry in time.

145 00:08:39.550 --> 00:08:42.140 So in this picture, we have these,

146 00:08:42.140 --> 00:08:44.410 what are often called immigrant events

147 00:08:44.410 --> 00:08:47.543 or parent events in black.

148 00:08:48.590 --> 00:08:50.250 And then we have the child events,

149 00:08:50.250 --> 00:08:53.230 the offspring from these events described in blue.

150 00:08:53.230 --> 00:08:58.090 So this appears to a pretty good stochastic process model,

151 00:08:58.090 --> 00:09:02.210 which is not overly complex, but is simply complex enough

152 00:09:02.210 --> 00:09:04.783 to capture contagion dynamics.

153 00:09:08.240 --> 00:09:11.440 So for this talk, I'm gonna be talking about some major

154 00:09:11.440 --> 00:09:16.440 challenges that are confronting the really data analysis

155 00:09:16.540 --> 00:09:18.820 using the Hawkes process.

156 00:09:18.820 --> 00:09:22.920 So very applied in nature, and these challenges persist

157 00:09:22.920 --> 00:09:25.760 despite the use of a very simple model.

158 00:09:25.760 --> 00:09:28.840 So basically, all the models that I'm showing you today

159 00:09:28.840 --> 00:09:32.640 are variations on this extremely simple model,

160 00:09:32.640 --> 00:09:35.360 as far as the Hawkes process literature goes.

161 00:09:35.360 --> 00:09:39.810 So we assume an exponential decay triggering function.

162 00:09:39.810 --> 00:09:41.930 So here in this self excitatory component,

163 00:09:41.930 --> 00:09:46.892 what this looks like is the triggering function

164 00:09:46.892 --> 00:09:51.892 is simply the exponentiation of negative omega,

165 00:09:52.470 --> 00:09:56.210 where one over omega is some sort of length scale.

166 00:09:56.210 --> 00:09:58.210 And then we've got T minus tn.

167 00:09:58.210 --> 00:10:01.050 Again, that difference between a T

168 00:10:01.050 --> 00:10:04.720 and preceding event times.

169 00:10:04.720 --> 00:10:06.500 And then we're also assuming Gaussian kernel

170 00:10:06.500 --> 00:10:08.550 spatial smoothers, very simple.

171 00:10:08.550 --> 00:10:11.700 And then finally, another simplifying assumption

172 00:10:11.700 --> 00:10:14.170 that we're making is separability.

173 00:10:14.170 --> 00:10:19.170 So, in these individual components of the rate function,

174 00:10:20.061 --> 00:10:24.770 we always have separation between the temporal component.

175 00:10:24.770 --> 00:10:27.760 So here on the left, and then the spatial component

176 00:10:27.760 --> 00:10:30.473 on the right, and this is a simplifying assumption.

177 00:10:34.230 --> 00:10:37.300 So what are the challenges that I'm gonna present today?

178 00:10:37.300 --> 00:10:42.300 The first challenge is big data because when we are modeling

179 00:10:42.480 --> 00:10:46.430 many events, what we see is the computational complexity

180 00:10:46.430 --> 00:10:48.423 of actually carrying out inference,

181 00:10:50.909 --> 00:10:53.900 whether using maximum likelihood or using say,

182 00:10:53.900 --> 00:10:55.550 Markov chain Monte Carlo,

183 00:10:55.550 --> 00:10:57.430 well, that's actually gonna explode quickly,

184 00:10:57.430 --> 00:10:59.320 the computational complexity.

185 00:10:59.320 --> 00:11:01.760 Something else is the spatial data precision.

186 00:11:01.760 --> 00:11:04.323 And this is actually related to big data.

187 00:11:06.060 --> 00:11:07.990 As we accrue more data,

188 00:11:07.990 --> 00:11:10.930 it's harder to guarantee data quality,

189 00:11:10.930 --> 00:11:14.770 but then also the tools that I'm gonna offer up to actually

190 00:11:14.770 --> 00:11:18.440 deal with poor spatial data precision are actually

191 00:11:18.440 --> 00:11:21.350 gonna also suffer under a big data setting.

192 00:11:21.350 --> 00:11:24.060 And then finally, big models.

193 00:11:24.060 --> 00:11:26.670 So, you know, when we're trying to draw very specific

194 00:11:26.670 --> 00:11:30.750 scientific conclusions from our model, then what happens?

195 00:11:30.750 --> 00:11:32.580 And all these data, excuse me,

196 00:11:32.580 --> 00:11:34.380 all these challenges are intertwined,

197 00:11:34.380 --> 00:11:35.830 and I'll try to express that.

198 00:11:38.990 --> 00:11:43.100 Finally today, I am interested in scientifically

199 00:11:43.100 --> 00:11:46.300 interpretable inference.

200 00:11:46.300 --> 00:11:48.470 So, I'm not gonna talk about prediction,

201 00:11:48.470 --> 00:11:50.720 but if you have questions about prediction,

202 00:11:50.720 --> 00:11:52.530 then we can talk about that afterward.

203 00:11:52.530 --> 00:11:53.363 I'm happy too.

204 00:11:57.450 --> 00:11:58.283 Okay.

205 00:11:58.283 --> 00:11:59.840 So I've shown you this figure before,

206 00:11:59.840 --> 00:12:01.760 and it's not the last time that you'll see it.

207 00:12:01.760 --> 00:12:04.780 But again, this is 4,000 gunshots in 2018.

208 00:12:04.780 --> 00:12:07.420 This is part of a larger dataset that's made available

209 00:12:07.420 --> 00:12:10.693 by the Washington DC Police Department.

210 00:12:11.680 --> 00:12:14.930 And in fact, from 2006 to 2018,

211 00:12:14.930 --> 00:12:19.560 we have over 85,000 potential gunshots recorded.

212 00:12:19.560 --> 00:12:20.580 How are they recorded?

213 00:12:20.580 --> 00:12:23.820 They're recorded using the help of an acoustic gunshot

214 00:12:23.820 --> 00:12:27.910 locator system that uses the actual acoustics

215 00:12:27.910 --> 00:12:31.570 to triangulate the time and the location

216 00:12:31.570 --> 00:12:33.913 of the individual gunshots.

217 00:12:35.030 --> 00:12:39.730 So in a 2018 paper, Charles Loeffler and Seth Flaxman,

218 00:12:39.730 --> 00:12:44.217 they used a subset of this data in a paper entitled

219 00:12:44.217 --> 00:12:45.690 ”Is Gun Violence Contagious?”

220 00:12:45.690 --> 00:12:48.730 And they in fact apply to Hawkes process model

221 00:12:48.730 --> 00:12:50.700 to try to determine their question,

222 00:12:50.700 --> 00:12:52.150 the answer to their question.

223 00:12:53.170 --> 00:12:54.800 But in order to do, though,

224 00:12:54.800 --> 00:12:56.870 they had to significantly subset.

225 00:12:56.870 --> 00:12:59.690 They took roughly 10% of the data.

226 00:12:59.690 --> 00:13:01.990 So the question is whether their conclusions,

227 00:13:01.990 --> 00:13:06.070 which in fact work yes to the affirmative,

228 00:13:06.070 --> 00:13:10.800 they were able to detect this kind of contagion dynamics.

229 00:13:10.800 --> 00:13:14.000 But the question is, do their results hold

230 00:13:14.000 --> 00:13:16.137 when we analyze the complete data set?

231 00:13:18.130 --> 00:13:20.450 So for likelihood based inference,

232 00:13:20.450 --> 00:13:25.340 which we're going to need to use in order to learn,

233 00:13:25.340 --> 00:13:28.543 in order to apply the Hawkes process to real-world data,

234 00:13:30.350 --> 00:13:34.200 for the first thing to see is that the likelihood

235 00:13:34.200 --> 00:13:38.670 takes on the form of an integral term on the left.

236 00:13:38.670 --> 00:13:42.950 And then we have a simple product of the rate function

237 00:13:42.950 --> 00:13:47.943 evaluated at our individual events, observed events.

238 00:13:49.880 --> 00:13:52.780 And when we consider the log likelihood,

239 00:13:52.780 --> 00:13:57.780 then it in fact will involve this term that I'm showing you

240 00:13:58.010 --> 00:14:00.410 on the bottom line, where it's the sum

241 00:14:00.410 --> 00:14:04.030 of the log of the, again, the rate function evaluated

242 00:14:04.030 --> 00:14:07.430 at the individual events. (background ringing)

243 00:14:07.430 --> 00:14:08.263 I'm sorry.

244 00:14:08.263 --> 00:14:10.260 You might be hearing a little bit of the sounds

245 00:14:10.260 --> 00:14:14.110 of Los Angeles in the background, and there's very little

246 00:14:14.110 --> 00:14:16.460 that I can do about Los Angeles.

247 00:14:16.460 --> 00:14:18.740 So moving on.

248 00:14:18.740 --> 00:14:23.740 So this summation in the log likelihood occurs.

249 00:14:24.800 --> 00:14:27.640 It actually involves a double summation.

250 00:14:27.640 --> 00:14:31.500 So it is the sum over all of our observations,

251 00:14:31.500 --> 00:14:33.970 of the log of the rate function.

252 00:14:33.970 --> 00:14:36.900 And then, again, the rate function because of the very

253 00:14:36.900 --> 00:14:40.840 specific form taken by the self excitatory component

254 00:14:40.840 --> 00:14:43.973 is also gonna involve this summation.

255 00:14:44.920 --> 00:14:48.543 So the upshot is that we actually need to evaluate.

256 00:14:49.380 --> 00:14:52.500 Every time we evaluate the log likelihood,

257 00:14:52.500 --> 00:14:57.500 we're going to need to evaluate N choose two,

258 00:14:59.110 --> 00:15:00.730 where N is the number of data points.

259 00:15:00.730 --> 00:15:05.580 N choose two terms, in this summation right here,

260 00:15:05.580 --> 00:15:07.930 and then we're gonna need to sum them together.

261 00:15:09.300 --> 00:15:13.223 And then the gradient also features this,

262 00:15:15.790 --> 00:15:18.133 quadratic computational complexity.

263 00:15:20.700 --> 00:15:23.120 So the solution, the first solution that I'm gonna offer up

264 00:15:23.120 --> 00:15:25.060 is not a statistical solution.

265 00:15:25.060 --> 00:15:26.960 It's a parallel computing solution.

266 00:15:26.960 --> 00:15:31.240 And the basic idea is, well, all of these terms that we need

267 00:15:31.240 --> 00:15:36.100 to sum over, evaluate and sum over, let's do it all at once

268 00:15:36.100 --> 00:15:38.083 and thereby speed up our inference.

269 00:15:40.730 --> 00:15:44.490 I do so, using multiple computational tools.

270 00:15:44.490 --> 00:15:49.490 So the first one is I use CP, they're just multicore CPUs.

271 00:15:50.380 --> 00:15:54.350 These can have anywhere from two to 100 cores.

272 00:15:54.350 --> 00:15:58.360 And then I combine this with something called SIMD,

273 00:15:58.360 --> 00:16:02.440 single instruction multiple data, which is vectorization.

274 00:16:02.440 --> 00:16:07.440 So the idea, the basic idea is that I can apply a function,

275 00:16:08.960 --> 00:16:13.420 the same function, the same instruction set to an extended

276 00:16:13.420 --> 00:16:18.420 register or vector of input data, and thereby speed up

277 00:16:19.950 --> 00:16:24.340 my computing by a factor that is proportional

278 00:16:24.340 --> 00:16:27.380 to the size of the vector that I'm evaluating

279 00:16:27.380 --> 00:16:29.170 my function over.

280 00:16:29.170 --> 00:16:32.970 And then, I actually can do something better than this.

281 00:16:32.970 --> 00:16:35.030 I can use a graphics processing unit,

282 00:16:35.030 --> 00:16:38.610 which instead of hundreds cores, has thousands of cores.

283 00:16:38.610 --> 00:16:42.160 And instead of SIMD, or it can be interpreted as SIMD,

284 00:16:42.160 --> 00:16:45.420 but Nvidia likes to call it a single instruction

285 00:16:45.420 --> 00:16:47.380 multiple threads or SIMT.

286 00:16:47.380 --> 00:16:50.040 And here, what the major difference

287 00:16:50.040 --> 00:16:52.143 is the scale at which it's occurring.

288 00:16:54.180 --> 00:16:58.320 And then, the other difference is that actually

289 00:16:58.320 --> 00:17:01.090 individual threads or small working groups of threads

290 00:17:01.090 --> 00:17:03.210 on my GPU can work together.

291 00:17:03.210 --> 00:17:06.720 So actually the tools that I have available are very complex

292 00:17:06.720 --> 00:17:09.540 and a lot of need for care.

293 00:17:09.540 --> 00:17:13.430 There's a lot of need to carefully code this up.

294 00:17:13.430 --> 00:17:17.760 The solution is not statistical, but it's very much

295 00:17:17.760 --> 00:17:19.400 an engineering solution.

296 00:17:19.400 --> 00:17:23.640 But the results are really, really impressive

297 00:17:23.640 --> 00:17:26.660 from my standpoint, because if I compare.

298 00:17:26.660 --> 00:17:31.660 So on the left, I'm comparing relative speed ups against

299 00:17:31.880 --> 00:17:36.880 a very fast single core SIMD implementation on the left.

300 00:17:39.780 --> 00:17:43.233 So my baseline right here is the bottom of this blue curve.

301 00:17:44.220 --> 00:17:47.520 The X axis is giving me the number of CPU threads

302 00:17:47.520 --> 00:17:50.593 that I'm using, between one and 18.

303 00:17:51.930 --> 00:17:54.760 And then, the top line is not using CPU threads.

304 00:17:54.760 --> 00:17:58.110 So I just create a top-line that's flat.

305 00:17:58.110 --> 00:18:00.680 This is the GPU results.

306 00:18:00.680 --> 00:18:03.560 If I don't use SIMD, if I use non vectorized

307 00:18:03.560 --> 00:18:05.680 single core computing, of course, this is still

308 00:18:05.680 --> 00:18:08.180 pre-compiled C++ implementation.

309 00:18:08.180 --> 00:18:10.950 So it's fast or at least faster than R,

310 00:18:10.950 --> 00:18:13.150 and I'll show you that on the next slide.

311 00:18:13.150 --> 00:18:17.380 If I do that, then AVX is twice as fast.

312 00:18:17.380 --> 00:18:19.593 As I increased the number of cores,

313 00:18:20.815 --> 00:18:24.310 my relative speed up increases,

314 00:18:24.310 --> 00:18:26.523 but I also suffer diminishing returns.

315 00:18:28.160 --> 00:18:31.230 And then that is actually all these simulations

316 00:18:31.230 --> 00:18:32.540 on the left-hand plot.

317 00:18:32.540 --> 00:18:34.380 That's for a fixed amount of data.

318 00:18:34.380 --> 00:18:38.420 That's 75,000 randomly generated data points

319 00:18:38.420 --> 00:18:41.520 at each iteration of my simulation.

320 00:18:41.520 --> 00:18:45.320 But I can also just look at the seconds per evaluation.

321 00:18:45.320 --> 00:18:48.630 So that's my Y axis on the right-hand side.

322 00:18:48.630 --> 00:18:52.910 So ideally I want this to be as low as possible.

323 00:18:52.910 --> 00:18:55.520 And then I'm increasing the number of data points

324 00:18:55.520 --> 00:18:58.353 on the Y axis, on the X axis, excuse me.

325 00:19:00.140 --> 00:19:03.020 And then as the number of threads that I use,

326 00:19:03.020 --> 00:19:04.890 as I increased the number of threads,

327 00:19:04.890 --> 00:19:08.000 then my implementation is much faster.

328 00:19:08.000 --> 00:19:11.600 But again, you're seeing this quadratic computational

329 00:19:11.600 --> 00:19:14.010 complexity at play, right.

330 00:19:14.010 --> 00:19:16.953 All of these lines are looking rather parabolic.

331 00:19:18.100 --> 00:19:20.880 Finally, I go down all the way to the bottom,

332 00:19:20.880 --> 00:19:22.400 where I've got my GPU curve,

333 00:19:22.400 --> 00:19:24.670 again, suffering, computational complexity,

334 00:19:24.670 --> 00:19:27.230 which the quadratic computational complexity,

335 00:19:27.230 --> 00:19:30.560 which we can't get past, but doing a much better job

336 00:19:30.560 --> 00:19:32.450 than the CPU computing.

337 00:19:32.450 --> 00:19:34.520 Now you might ask, well, you might say,

338 00:19:34.520 --> 00:19:37.870 well, a 100 fold speed up is not that great.

339 00:19:37.870 --> 00:19:40.890 So I'd put this in perspective and say, well,

340 00:19:40.890 --> 00:19:45.450 what does this mean for R, which I use every day?

341 00:19:45.450 --> 00:19:48.920 Well, what it amounts to,

342 00:19:48.920 --> 00:19:51.230 and here, I'll just focus on the relative speed up

343 00:19:51.230 --> 00:19:55.360 over our implementation on the right.

344 00:19:55.360 --> 00:19:59.423 The GPU is reliably over 1000 times faster.

345 00:20:03.680 --> 00:20:08.680 So the way that Charles Loeffler and Seth Flaxman

346 00:20:12.420 --> 00:20:16.170 obtained a subset of their data was actually

347 00:20:16.170 --> 00:20:17.993 by thinning the data.

348 00:20:21.260 --> 00:20:23.840 They needed to do so because of the sheer computational

349 00:20:23.840 --> 00:20:27.150 complexity of using the Hawkes model.

350 00:20:27.150 --> 00:20:30.170 So, I'm not criticizing this in any way,

351 00:20:30.170 --> 00:20:33.910 but I'm simply pointing out why our results

352 00:20:33.910 --> 00:20:36.470 using the full data set, differ.

353 00:20:36.470 --> 00:20:39.538 So on the left, on the top left,

354 00:20:39.538 --> 00:20:43.600 we have the posterior density for the spatial length scale

355 00:20:43.600 --> 00:20:45.600 of the self excitatory component.

356 00:20:45.600 --> 00:20:47.600 And when we use the full data set,

357 00:20:47.600 --> 00:20:51.000 then we believe that we're operating more at around 70

358 00:20:51.000 --> 00:20:56.000 meters instead of the 126 inferred in the original paper.

359 00:20:56.480 --> 00:21:00.900 So one thing that you might notice is our posterior

360 00:21:00.900 --> 00:21:05.477 densities are much more concentrated than in blue,

361 00:21:07.930 --> 00:21:12.150 than the original analysis in Salmon.

362 00:21:12.150 --> 00:21:13.970 And this of course makes sense.

363 00:21:13.970 --> 00:21:16.673 We're using 10 times the amount of the data.

364 00:21:17.610 --> 00:21:20.360 Our temporal length scale is also meant,

365 00:21:20.360 --> 00:21:24.070 is also, we believe, much smaller, in fact.

366 00:21:24.070 --> 00:21:27.550 So now it's down to one minute instead of 10 minutes.

367 00:21:27.550 --> 00:21:29.070 Again, this could be interpreted

368 00:21:29.070 --> 00:21:31.540 as the simple result of thinning.

369 00:21:31.540 --> 00:21:34.618 And then finally, I just want to focus on this on

370 00:21:34.618 --> 00:21:38.733 the green posterior density.

371 00:21:40.972 --> 00:21:43.772 This is the proportion of events that we're interpreting

372 00:21:44.760 --> 00:21:49.760 that arise from self excitation or contagion dynamics.

373 00:21:49.890 --> 00:21:54.890 Experts believe that anywhere between 10 and 18% of gun

374 00:21:56.010 --> 00:21:59.380 violence events are retaliatory in nature.

375 00:21:59.380 --> 00:22:04.380 So actually our inference is kind of agreeing with,

376 00:22:06.960 --> 00:22:11.783 it safely within the band suggested by the experts.

377 00:22:15.030 --> 00:22:17.590 Actually, another thing that we can do,

378 00:22:17.590 --> 00:22:21.510 and that also requires a pretty computationally.

379 00:22:21.510 --> 00:22:26.510 So this is also quadratic computational complexity.

380 00:22:26.940 --> 00:22:30.110 Again, is post-processing.

381 00:22:30.110 --> 00:22:32.410 So if, for example, for individual events,

382 00:22:32.410 --> 00:22:36.370 we want to know the probability that the event arose

383 00:22:38.203 --> 00:22:41.050 from retaliatory gun violence,

384 00:22:41.050 --> 00:22:46.050 then we could look at the self excitatory component

385 00:22:46.210 --> 00:22:49.150 of the rate function divided by the total rate function.

386 00:22:49.150 --> 00:22:51.220 And then we can just look at the posterior

387 00:22:51.220 --> 00:22:54.970 distribution of this statistic.

388 00:22:54.970 --> 00:22:58.415 And this will give us our posterior probability

389 00:22:58.415 --> 00:23:03.415 that the event arose from contagion dynamics at least.

390 00:23:03.930 --> 00:23:05.790 And you can see that we can actually observe

391 00:23:05.790 --> 00:23:09.157 a very wide variety of values.

392 00:23:22.740 --> 00:23:27.740 So the issue of big data is actually not gonna go away,

393 00:23:28.450 --> 00:23:32.163 as we move on to discussing spatial data precision.

394 00:23:33.290 --> 00:23:37.760 Now, I'll tell you a little bit more about this data.

395 00:23:37.760 --> 00:23:42.100 All the data that we access is freely accessible online,

396 00:23:42.100 --> 00:23:47.100 is rounded to the nearest 100 meters

397 00:23:47.930 --> 00:23:51.470 by the DC Police Department.

398 00:23:51.470 --> 00:23:56.313 And the reason that they do this is for reasons of privacy.

399 00:23:57.740 --> 00:24:00.820 So one immediate question that we can ask is, well,

400 00:24:00.820 --> 00:24:05.483 how does this rounding actually affect our inference?

401 00:24:09.890 --> 00:24:12.590 Now we actually observed wildfires

402 00:24:12.590 --> 00:24:14.863 of wildly different sizes.

403 00:24:15.800 --> 00:24:18.770 And the question is, well, how does...

404 00:24:23.220 --> 00:24:27.520 If we want to model the spread of wildfires,

405 00:24:27.520 --> 00:24:29.810 then it would be useful to know

406 00:24:29.810 --> 00:24:32.263 where the actual ignition site,

407 00:24:33.460 --> 00:24:35.483 the site of ignition was.

408 00:24:37.020 --> 00:24:41.090 Where did the fire occur originally?

409 00:24:41.090 --> 00:24:44.380 And many of these fires are actually discovered

410 00:24:44.380 --> 00:24:47.610 out in the wild, far away from humans.

411 00:24:47.610 --> 00:24:50.000 And there's a lot of uncertainty.

412 00:24:50.000 --> 00:24:54.133 There's actually a large swaths of land that are involved.

413 00:24:57.010 --> 00:25:00.030 Finally, this, this global influenza data

414 00:25:00.030 --> 00:25:02.620 is very nice for certain reasons.

415 00:25:02.620 --> 00:25:06.730 For example, it features all of the observations,

416 00:25:06.730 --> 00:25:09.720 actually provide a viral genome data.

417 00:25:09.720 --> 00:25:12.370 So we can perform other more complex

418 00:25:12.370 --> 00:25:13.610 analyses on the data.

419 00:25:13.610 --> 00:25:16.120 And in fact, I'll do that in the third section

420 00:25:17.360 --> 00:25:18.563 for related data.

421 00:25:20.849 --> 00:25:24.900 But the actual spatial precision for this data is very poor.

422 00:25:24.900 --> 00:25:28.550 So, for some of these viral cases,

423 00:25:28.550 --> 00:25:31.890 we know the city in which it occurred.

424 00:25:31.890 --> 00:25:33.750 For some of them, we know the region

425 00:25:33.750 --> 00:25:35.200 or the state in which it occurred.

426 00:25:35.200 --> 00:25:37.100 And for some of them, we know the country

427 00:25:37.100 --> 00:25:38.150 in which it occurred.

428 00:25:40.230 --> 00:25:42.050 So I'm gonna start with the easy problem,

429 00:25:42.050 --> 00:25:47.050 which is analyzing the DC gun violence, the DC gunshot data.

430 00:25:47.740 --> 00:25:50.440 And here again, the police department rounds the data

431 00:25:50.440 --> 00:25:52.150 to the nearest hundred meters.

432 00:25:52.150 --> 00:25:53.260 So what do we do?

433 00:25:53.260 --> 00:25:56.510 We take that at face value and we simply use,

434 00:25:56.510 --> 00:26:00.950 place a uniform prior over the 10,000 meters square

435 00:26:03.650 --> 00:26:06.260 that is centered at each one of our observations.

436 00:26:06.260 --> 00:26:10.270 So here I'm denoting our actual data,

437 00:26:10.270 --> 00:26:14.500 our observed data with this kind of Gothic X,

438 00:26:14.500 --> 00:26:16.930 and then I'm placing a prior over the location

439 00:26:16.930 --> 00:26:18.990 at which the gunshot actually occurred.

440 00:26:18.990 --> 00:26:23.120 And this is a uniform prior over a box centered at my data.

441 00:26:23.120 --> 00:26:28.050 And using this prior actually has another interpretation

442 00:26:28.050 --> 00:26:32.740 similar to some other concepts

443 00:26:32.740 --> 00:26:35.770 from the missing data literature.

444 00:26:35.770 --> 00:26:40.470 And use of this prior actually corresponds to using

445 00:26:40.470 --> 00:26:43.010 something called the group data likelihood.

446 00:26:43.010 --> 00:26:48.010 And it's akin to the expected, complete data likelihood

447 00:26:48.429 --> 00:26:52.543 if you're familiar with the missing data literature.

448 00:26:53.460 --> 00:26:56.980 So what we do, and I'm not gonna get too much into

449 00:26:56.980 --> 00:27:00.130 the inference at this point, but we actually use MCMC

450 00:27:00.130 --> 00:27:03.890 to simultaneously infer the locations,

451 00:27:03.890 --> 00:27:07.680 and the Hawkes model parameters,

452 00:27:07.680 --> 00:27:10.203 the rate function parameters at the same time.

453 00:27:12.310 --> 00:27:14.690 So here, I'm just showing you a couple of examples

454 00:27:14.690 --> 00:27:16.470 of what this looks like.

455 00:27:16.470 --> 00:27:19.620 For each one of our observations colored yellow,

456 00:27:19.620 --> 00:27:22.283 we then have 100 posterior samples.

457 00:27:24.540 --> 00:27:28.110 So these dynamics can take on different forms

458 00:27:28.110 --> 00:27:32.000 and they take on different forms in very complex ways,

459 00:27:32.000 --> 00:27:36.340 simply because what we're essentially doing when we're...

460 00:27:38.190 --> 00:27:40.950 I'm going to loosely use the word impute.

461 00:27:40.950 --> 00:27:44.180 When we're imputing this data, when we're actually inferring

462 00:27:44.180 --> 00:27:47.370 these locations, we're basically simulating

463 00:27:47.370 --> 00:27:50.653 from a very complex n-body problem.

464 00:27:52.920 --> 00:27:57.120 So on the left, how can we interpret this?

465 00:27:57.120 --> 00:28:00.760 Well, we've got these four points and the model believes

466 00:28:00.760 --> 00:28:02.430 that actually they are farther away

467 00:28:02.430 --> 00:28:03.990 from each other than observed.

468 00:28:03.990 --> 00:28:05.110 Why is that?

469 00:28:05.110 --> 00:28:08.960 Well, right in the middle here, we have a shopping center,

470 00:28:08.960 --> 00:28:12.980 where there's actually many less gunshots.

471 00:28:12.980 --> 00:28:14.750 And then we've got residential areas

472 00:28:14.750 --> 00:28:18.070 where there are many more gunshots on the outside.

473 00:28:18.070 --> 00:28:21.513 And the bottom right, we actually have all of these,

474 00:28:25.550 --> 00:28:30.090 we believe that the actual locations of these gunshots

475 00:28:30.090 --> 00:28:34.180 collect closer together, kind of toward a very high

476 00:28:34.180 --> 00:28:36.883 intensity region in Washington, DC.

477 00:28:39.400 --> 00:28:40.930 And then we can just think about

478 00:28:40.930 --> 00:28:43.910 the general posterior displacement.

479 00:28:43.910 --> 00:28:45.670 So the mean posterior displacement.

480 00:28:45.670 --> 00:28:48.443 So in general, are there certain points that,

481 00:28:49.941 --> 00:28:53.420 where the model believes that the gunshots occurred

482 00:28:53.420 --> 00:28:57.510 further away from the observed events?

483 00:28:57.510 --> 00:28:59.923 And in general, there's not really.

484 00:29:01.380 --> 00:29:04.190 It's hard to come up with any steadfast rules.

485 00:29:04.190 --> 00:29:07.860 For example, in the bottom, right, we have some shots,

486 00:29:07.860 --> 00:29:12.700 some gunshots that show a very large posterior displacement,

487 00:29:12.700 --> 00:29:15.380 and they're in a very high density region.

488 00:29:15.380 --> 00:29:18.590 Whereas on the top, we also get large displacement

489 00:29:18.590 --> 00:29:21.210 and we're not surrounded by very many gunshots at all.

490 00:29:21.210 --> 00:29:24.250 So it is a very complex n-body problem

491 00:29:24.250 --> 00:29:25.803 that we're solving.

492 00:29:27.330 --> 00:29:29.500 And the good news is, for this problem,

493 00:29:29.500 --> 00:29:31.750 it doesn't matter much anyway.

494 00:29:31.750 --> 00:29:34.563 The results that we get are pretty much the same.

495 00:29:37.410 --> 00:29:42.250 I mean, so from the standpoint of statistical significance,

496 00:29:42.250 --> 00:29:44.920 we do get some statistically significant results.

497 00:29:44.920 --> 00:29:47.390 So in this figure, on the top,

498 00:29:47.390 --> 00:29:50.560 I'm showing you 95% credible intervals,

499 00:29:50.560 --> 00:29:55.560 and this is the self excitatory spatial length scale.

500 00:29:55.560 --> 00:29:57.040 We believe that it's smaller,

501 00:29:57.040 --> 00:30:00.550 but from a practical standpoint, it's not much smaller.

502 00:30:00.550 --> 00:30:02.840 It's a difference between 60 meters

503 00:30:02.840 --> 00:30:06.823 and maybe it's at 73 meters, 72 meters.

504 00:30:12.500 --> 00:30:15.550 But we shouldn't take too much comfort

505 00:30:15.550 --> 00:30:18.910 because actually as we increase the spatial prec-

506 00:30:18.910 --> 00:30:21.840 excuse me, as we decrease the spatial precision,

507 00:30:21.840 --> 00:30:25.210 we find that the model that does not take account

508 00:30:26.120 --> 00:30:28.780 of the rounding, performs much worse.

509 00:30:28.780 --> 00:30:32.760 So for example, if you look in the table,

510 00:30:32.760 --> 00:30:36.373 then we have the fixed locations model,

511 00:30:37.310 --> 00:30:40.050 where I'm not actually inferring the locations.

512 00:30:40.050 --> 00:30:44.590 And I just want to see, what's the empirical coverage

513 00:30:44.590 --> 00:30:47.003 of the 95% credible intervals?

514 00:30:48.010 --> 00:30:52.580 And let's just focus on the 95%

515 00:30:52.580 --> 00:30:54.900 credible intervals, specifically,

516 00:30:54.900 --> 00:30:58.670 simply because actually the other intervals,

517 00:30:58.670 --> 00:31:03.230 the 50% credible interval, the 80% credible interval,

518 00:31:03.230 --> 00:31:07.263 they showed the similar dynamic, which is that as we,

519 00:31:09.520 --> 00:31:12.500 so if we start on the right-hand side,

520 00:31:12.500 --> 00:31:16.260 we have precision down to down to 0.1.

521 00:31:16.260 --> 00:31:19.370 This is a unit list example.

522 00:31:19.370 --> 00:31:21.940 So we have higher precision, actually.

523 00:31:21.940 --> 00:31:24.160 Then we see that we have very good coverage,

524 00:31:24.160 --> 00:31:27.957 even if we don't take this locational

525 00:31:30.550 --> 00:31:32.303 coarsening into account.

526 00:31:33.160 --> 00:31:38.020 But as we increase the size of our error box,

527 00:31:38.020 --> 00:31:40.960 then we actually lose coverage,

528 00:31:40.960 --> 00:31:43.720 and we deviate from that 95% coverage.

529 00:31:43.720 --> 00:31:46.290 And then finally, if we increase too much,

530 00:31:46.290 --> 00:31:48.770 then we're never actually going to be

531 00:31:50.800 --> 00:31:55.563 capturing the true spatial length scale,

532 00:31:56.740 --> 00:31:59.040 whereas if we actually do sample the locations,

533 00:31:59.040 --> 00:32:00.970 we perform surprisingly well,

534 00:32:00.970 --> 00:32:05.893 even when we have a very high amount of spatial coarsening.

535 00:32:08.010 --> 00:32:10.550 Well, how else can we break the model?

536 00:32:10.550 --> 00:32:12.750 Another way that we can break this model,

537 00:32:12.750 --> 00:32:15.690 and by break the model, I mean, my naive model

538 00:32:15.690 --> 00:32:18.320 where I'm not inferring the locations.

539 00:32:18.320 --> 00:32:21.710 Another way that we can break this model

540 00:32:21.710 --> 00:32:24.380 is simply by considering data

541 00:32:24.380 --> 00:32:28.400 where we have variable spatial coarsening.

542 00:32:28.400 --> 00:32:30.860 That is where different data points

543 00:32:31.710 --> 00:32:34.270 are coarsened different amounts,

544 00:32:34.270 --> 00:32:36.683 so we have a variable precision.

545 00:32:40.290 --> 00:32:42.850 So considering the wildfire data,

546 00:32:42.850 --> 00:32:47.850 we actually see something with the naive approach

547 00:32:48.480 --> 00:32:51.010 where we're not inferring the locations.

548 00:32:51.010 --> 00:32:55.960 We actually see something that is actually recorded

549 00:32:55.960 --> 00:33:00.370 elsewhere in the Hawkes process literature.

550 00:33:00.370 --> 00:33:04.560 And that is that when we try to use a flexible

551 00:33:04.560 --> 00:33:07.360 background function, as we are trying to do,

552 00:33:07.360 --> 00:33:11.933 then we get this multimodal posterior distribution.

553 00:33:12.780 --> 00:33:14.350 And that's fine.

554 00:33:14.350 --> 00:33:17.410 We can also talk about it in a frequentist,

555 00:33:17.410 --> 00:33:18.710 from the frequency standpoint,

556 00:33:18.710 --> 00:33:21.360 because it's observed there as well

557 00:33:21.360 --> 00:33:24.910 in the maximum likelihood context, which is,

558 00:33:24.910 --> 00:33:27.560 we still see this multimodality.

559 00:33:28.740 --> 00:33:32.253 What specific form does this multimodality take?

560 00:33:33.710 --> 00:33:38.710 So what we see is that we get modes around the places

561 00:33:39.970 --> 00:33:44.970 where the background rate parameters,

562 00:33:46.750 --> 00:33:49.600 the background length scale parameters are equal

563 00:33:49.600 --> 00:33:52.950 to the temporal, excuse me, the self excitatory

564 00:33:54.040 --> 00:33:55.830 length scale parameters.

565 00:33:55.830 --> 00:33:59.190 So for the naive model, it's mode A,

566 00:34:00.280 --> 00:34:02.560 it believes that the spatial length scale

567 00:34:02.560 --> 00:34:07.410 is about 24 kilometers, and that the spatial length scale

568 00:34:07.410 --> 00:34:09.160 of the self excitatory dynamics

569 00:34:09.160 --> 00:34:13.930 are also roughly 24 kilometers.

570 00:34:13.930 --> 00:34:15.330 And then for the other mode,

571 00:34:16.180 --> 00:34:19.970 we get equal temporal length scales.

572 00:34:19.970 --> 00:34:23.930 So here, it believes 10 days, and 10 days

573 00:34:23.930 --> 00:34:27.320 for the self excitatory in the background component.

574 00:34:27.320 --> 00:34:29.010 And this can be very bad indeed.

575 00:34:29.010 --> 00:34:31.430 So for example, for mode A,

576 00:34:31.430 --> 00:34:35.910 it completely, the Hawkes model completely fails

577 00:34:35.910 --> 00:34:40.400 to capture seasonal dynamics, which is the first thing

578 00:34:40.400 --> 00:34:42.910 that you would want it to pick up on.

579 00:34:42.910 --> 00:34:46.690 The first thing that you would want it to understand

580 00:34:46.690 --> 00:34:49.060 is that wildfires...

581 00:34:49.060 --> 00:34:50.650 Okay, I need to be careful here

582 00:34:50.650 --> 00:34:52.643 because I'm not an expert on wildfires.

583 00:34:54.610 --> 00:34:55.830 I'll go out on a limb and say,

584 00:34:55.830 --> 00:34:59.983 wildfires don't happen in Alaska during the winter.

585 00:35:02.920 --> 00:35:05.060 On the other hand, when we use the full model

586 00:35:05.060 --> 00:35:08.450 and we're actually simultaneously inferring the locations,

587 00:35:08.450 --> 00:35:10.950 then we get this kind of Goldilocks effect,

588 00:35:10.950 --> 00:35:14.400 where here, the spatial length scale

589 00:35:14.400 --> 00:35:17.010 is somewhere around 35 kilometers,

590 00:35:17.010 --> 00:35:20.840 which is between the 23 kilometers and 63 kilometers

591 00:35:20.840 --> 00:35:25.840 for mode modes A and B, and we see that reliably.

592 00:35:33.160 --> 00:35:36.843 I can stop for some questions because I'm making good time.

593 00:35:44.025 --> 00:35:49.025 <v Man>Does anybody have any questions, if you want to ask?</v>

594 00:35:52.120 --> 00:35:53.430 <v Student>What's the interpretation</v>

595 00:35:53.430 --> 00:35:56.180 of the spatial length scale and the temporal length scale?

596 00:35:56.180 --> 00:35:58.910 What do those numbers actually mean?

597 00:35:58.910 --> 00:36:02.180 <v ->Yeah, thank you.</v>

598 00:36:02.180 --> 00:36:06.230 So, the interpretation of the...

599 00:36:06.230 --> 00:36:10.660 I think that the most useful interpretation,

600 00:36:10.660 --> 00:36:14.910 so just to give you an idea of how they can be interpreted.

601 00:36:14.910 --> 00:36:19.770 So for example, for the self excitatory component, right,

602 00:36:19.770 --> 00:36:22.283 that's describing the contagion dynamics.

603 00:36:23.420 --> 00:36:28.420 What this is saying is that if we see a wildfire,

604 00:36:29.110 --> 00:36:32.400 then we expect to observe another wildfire

605 00:36:34.020 --> 00:36:38.193 with mean distribution of one day.

606 00:36:40.750 --> 00:36:45.750 So the temporal length scale is in units days.

607 00:36:46.120 --> 00:36:49.740 So in the full model, after observing the wildfire,

608 00:36:49.740 --> 00:36:53.520 we expect to see another wildfire with mean, you know,

609 00:36:53.520 --> 00:36:55.143 on average, the next day.

610 00:36:56.390 --> 00:37:01.390 And this of course, you know, we have this model

611 00:37:01.620 --> 00:37:05.250 that's taking space and time into account.

612 00:37:05.250 --> 00:37:10.020 So the idea though, is that because of the separability

613 00:37:10.020 --> 00:37:12.200 in our model, we're basically simply

614 00:37:12.200 --> 00:37:14.343 expecting to see it somewhere.

615 00:37:18.920 --> 00:37:19.987 <v Student>Thank you.</v>

616 00:37:23.960 --> 00:37:25.960 <v Man>Any other questions?</v>

617 00:37:25.960 --> 00:37:29.627 (man speaking indistinctly)

618 00:37:30.620 --> 00:37:32.620 <v Student>Hi, can I have one question?</v>

619 00:37:34.520 --> 00:37:35.890 <v ->Go head.</v>

620 00:37:35.890 --> 00:37:37.573 <v Student>Okay.</v>

621 00:37:37.573 --> 00:37:38.406 I'm curious.

622 00:37:38.406 --> 00:37:39.277 What is a main difference between

623 00:37:39.277 --> 00:37:42.850 the naive model A and the naive model B?

624 00:37:42.850 --> 00:37:43.683 <v ->Okay.</v>

625 00:37:43.683 --> 00:37:44.761 So, sorry.

626 00:37:44.761 --> 00:37:45.594 This is...

627 00:37:46.860 --> 00:37:49.260 I think I could have presented

628 00:37:49.260 --> 00:37:52.070 this aspect better within the table itself.

629 00:37:52.070 --> 00:37:55.263 So this is the same exact model.

630 00:37:57.680 --> 00:38:00.520 But all that I'm doing is I'm applying

631 00:38:00.520 --> 00:38:02.840 the model multiple times.

632 00:38:02.840 --> 00:38:05.893 So in this case, I'm using Markov chain Monte Carlo.

633 00:38:07.490 --> 00:38:09.850 So one question that you might ask is,

634 00:38:09.850 --> 00:38:14.850 well, what happens when I run MCMC multiple times?

635 00:38:16.490 --> 00:38:20.060 Sometimes I get trapped in one mode.

636 00:38:20.060 --> 00:38:22.370 Sometimes I get trapped in another mode.

637 00:38:22.370 --> 00:38:25.050 You can just for, you know, a mental cartoon,

638 00:38:25.050 --> 00:38:27.090 we can think of like a (indistinct)

639 00:38:27.090 --> 00:38:29.680 a mixture of Gaussian distribution, right.

640 00:38:29.680 --> 00:38:33.720 Sometimes I can get trapped in this Gaussian component.

641 00:38:33.720 --> 00:38:36.570 Sometimes I could get trapped in this Gaussian component.

642 00:38:38.290 --> 00:38:43.290 So there's nothing intrinsically wrong with multimodality.

643 00:38:43.760 --> 00:38:47.490 We prefer to avoid it as best we can simply because it makes

644 00:38:47.490 --> 00:38:49.963 interpretation much more difficult.

645 00:38:52.040 --> 00:38:56.010 In this case, if I only perform inference

646 00:38:56.010 --> 00:38:59.560 and only see mode A, then I'm never actually gonna be

647 00:38:59.560 --> 00:39:04.560 picking up on seasonal dynamics.

648 00:39:07.320 --> 00:39:08.470 Does that (indistinct)?

649 00:39:09.760 --> 00:39:11.900 <v Woman>Yeah, it's clear.</v>

650 00:39:11.900 --> 00:39:13.080 <v Instructor>Okay.</v>

651 00:39:13.080 --> 00:39:15.510 <v Woman>Okay, and I also (indistinct).</v>

652 00:39:15.510 --> 00:39:18.030 So for the full model, you can capture

653 00:39:18.030 --> 00:39:20.820 the spatial dynamic property.

654 00:39:20.820 --> 00:39:22.740 So how to do that?

655 00:39:22.740 --> 00:39:25.437 So I know you need the Hawkes process that sees,

656 00:39:25.437 --> 00:39:28.150 clarifies the baseline.

657 00:39:28.150 --> 00:39:31.600 So how do you estimate a baseline part?

658 00:39:31.600 --> 00:39:32.777 <v ->Oh, okay, great.</v>

659 00:39:34.574 --> 00:39:35.743 In the exact same way.

660 00:39:37.280 --> 00:39:39.130 <v Student>Okay, I see.</v>

661 00:39:39.130 --> 00:39:44.130 <v ->So I'm jointly, simultaneously performing inference</v>

662 00:39:44.610 --> 00:39:47.380 over all of the model parameters.

663 00:39:47.380 --> 00:39:50.993 And I can go all the way back.

664 00:39:53.320 --> 00:39:54.419 Right.

665 00:39:54.419 --> 00:39:56.519 'Cause it's actually a very similar model.

666 00:39:57.960 --> 00:39:58.793 Yes.

667 00:39:58.793 --> 00:40:01.560 So this is my baseline.

668 00:40:01.560 --> 00:40:05.720 And so, for example, when we're talking about that temporal

669 00:40:05.720 --> 00:40:09.050 smooth that you saw on that last figure,

670 00:40:09.050 --> 00:40:13.390 where I'm supposed to be capturing seasonal dynamics.

671 00:40:13.390 --> 00:40:17.790 Well, if tau T, which I'm just calling

672 00:40:17.790 --> 00:40:21.728 my temporal length scale, if that is too large,

673 00:40:21.728 --> 00:40:24.310 then I'm never going to be capturing

674 00:40:24.310 --> 00:40:28.430 those seasonal dynamics, which I would be hoping to capture

675 00:40:28.430 --> 00:40:30.943 precisely using this background smoother.

676 00:40:33.080 --> 00:40:34.040 <v Student>Okay, I see.</v>

677 00:40:34.040 --> 00:40:37.850 So it looks like they assume the formula for the baseline,

678 00:40:37.850 --> 00:40:41.910 and then you estimates some parameters in these formulas.

679 00:40:41.910 --> 00:40:43.210 <v ->Yes.</v>

680 00:40:43.210 --> 00:40:44.190 <v Student>In my understanding,</v>

681 00:40:44.190 --> 00:40:47.060 in the current Hawkes literature,

682 00:40:47.060 --> 00:40:48.680 somebody uses (indistinct) function

683 00:40:48.680 --> 00:40:51.500 to approximate baseline also.

684 00:40:51.500 --> 00:40:52.333 <v ->Yes.</v>

685 00:40:52.333 --> 00:40:53.906 <v Student>This is also interesting.</v>

686 00:40:53.906 --> 00:40:54.895 Thank you. <v ->Yes.</v>

687 00:40:54.895 --> 00:40:55.728 Okay, okay, great.

688 00:40:55.728 --> 00:40:59.080 I'm happy to show another, you know.

689 00:40:59.080 --> 00:41:00.380 And of course I did not invent this.

690 00:41:00.380 --> 00:41:03.030 This is just another tact that you can take.

691 00:41:03.030 --> 00:41:03.880 <v Student>Yeah, yeah, yeah, yeah.</v>

692 00:41:03.880 --> 00:41:04.713 That's interesting.

693 00:41:04.713 --> 00:41:05.546 Thanks

694 00:41:05.546 --> 00:41:06.379 <v ->Yup.</v>

695 00:41:09.810 --> 00:41:11.810 <v Student>As just a quick follow up on</v>

696 00:41:12.860 --> 00:41:16.140 when you were showing the naive model,

697 00:41:16.140 --> 00:41:18.513 and this maybe a naive question on my part.

698 00:41:19.920 --> 00:41:23.980 Did you choose naive model A to be the one

699 00:41:23.980 --> 00:41:26.680 that does the type seasonality or is that approach

700 00:41:26.680 --> 00:41:31.013 just not (indistinct) seasonality?

701 00:41:32.950 --> 00:41:36.780 <v ->So I think that the point</v>

702 00:41:38.030 --> 00:41:41.650 is that sometimes based on, you know,

703 00:41:41.650 --> 00:41:43.550 I'm doing MCMC.

704 00:41:43.550 --> 00:41:46.320 It's random in nature, right.

705 00:41:46.320 --> 00:41:49.070 So just sometimes when I do that,

706 00:41:49.070 --> 00:41:52.550 I get trapped in that mode A,

707 00:41:52.550 --> 00:41:54.943 and sometimes I get trapped in that mode B.

708 00:41:59.560 --> 00:42:03.660 The label that I apply to it is just arbitrary,

709 00:42:03.660 --> 00:42:06.113 but maybe I'm not getting your question.

710 00:42:10.880 --> 00:42:13.830 <v Student>No, I think you did.</v>

711 00:42:13.830 --> 00:42:16.820 So, it's possible that we detect it.

712 00:42:16.820 --> 00:42:18.430 It's possible that we don't.

713 00:42:20.045 --> 00:42:20.878 <v ->Exactly.</v>

714 00:42:20.878 --> 00:42:22.000 And that's, you know,

715 00:42:22.000 --> 00:42:23.263 <v Student>That's what it is.</v>

716 00:42:23.263 --> 00:42:24.760 <v ->multimodality.</v>

717 00:42:24.760 --> 00:42:26.830 So this is kind of nice though,

718 00:42:26.830 --> 00:42:29.970 that this can actually give you,

719 00:42:29.970 --> 00:42:32.973 that actually inferring the locations can somehow,

720 00:42:34.560 --> 00:42:37.330 at least in this case, right,

721 00:42:37.330 --> 00:42:40.000 I mean, this is a case study, really,

722 00:42:40.000 --> 00:42:43.260 that this can help resolve that multimodality.

723 00:42:46.640 --> 00:42:48.315 <v Student>Thank you.</v>

724 00:42:48.315 --> 00:42:49.148 Yeah.

725 00:42:49.148 --> 00:42:54.148 <v Student>So back to the comparison between CPU and GPU.</v>

726 00:42:54.820 --> 00:42:59.700 Let's say, if we increase the thread of CPU,

727 00:42:59.700 --> 00:43:04.700 say like to infinity, will it be possible that the speed

728 00:43:05.737 --> 00:43:09.033 of CPU match the speed up of GPU?

729 00:43:11.810 --> 00:43:12.643 <v ->So.</v>

730 00:43:15.170 --> 00:43:16.760 You're saying if we increase.

731 00:43:16.760 --> 00:43:18.590 So, can I ask you one more time?

732 00:43:18.590 --> 00:43:21.190 Can I just ask for clarification?

733 00:43:21.190 --> 00:43:23.733 You're saying if we increase what to infinity?

734 00:43:24.640 --> 00:43:26.187 <v Student>The thread of CPU.</v>

735 00:43:27.560 --> 00:43:31.520 I think in the graph you're increasing the threads

736 00:43:31.520 --> 00:43:34.203 of CPU from like one to 80.

737 00:43:35.380 --> 00:43:39.030 And the speed up increase as the number

738 00:43:39.030 --> 00:43:41.770 of threats increasing.

739 00:43:41.770 --> 00:43:44.860 So just say like, let's say the threads of CPU

740 00:43:44.860 --> 00:43:49.860 increase to infinity, will the speed up match,

741 00:43:50.540 --> 00:43:53.690 because GPU with like (indistinct).

742 00:43:53.690 --> 00:43:55.843 Very high, right. <v ->Yeah, yeah.</v>

743 00:43:57.080 --> 00:43:59.510 Let me show you another figure,

744 00:43:59.510 --> 00:44:01.603 and then we can return to that.

745 00:44:02.747 --> 00:44:05.363 I think it's a good segue into the next section.

746 00:44:06.960 --> 00:44:09.060 So, let me answer that in a couple slides.

747 00:44:10.171 --> 00:44:11.740 <v Student>Okay, sounds good.</v>

748 00:44:11.740 --> 00:44:12.573 <v ->Okay.</v>

749 00:44:12.573 --> 00:44:15.180 So, questions about.

750 00:44:15.180 --> 00:44:17.630 I've gotten some good questions about how do we interpret

751 00:44:17.630 --> 00:44:22.630 the length scales and then this makes me think about,

752 00:44:23.380 --> 00:44:25.970 well, if all that we're doing is interpreting

753 00:44:25.970 --> 00:44:29.200 the length scales, how much is that telling us about

754 00:44:29.200 --> 00:44:32.130 the phenomenon that we're interested in?

755 00:44:32.130 --> 00:44:36.540 And can we actually craft more complex hierarchical models

756 00:44:36.540 --> 00:44:40.500 so that we can actually learn something perhaps

757 00:44:40.500 --> 00:44:42.750 even biologically interpretable?

758 00:44:42.750 --> 00:44:46.650 So here, I'm looking at 2014, 2016

759 00:44:46.650 --> 00:44:49.650 Ebola virus outbreak data.

760 00:44:49.650 --> 00:44:53.870 This is over almost 22,000 cases.

761 00:44:53.870 --> 00:44:58.697 And of these cases, we have about 1600

762 00:45:00.320 --> 00:45:04.993 that are providing us genome data.

763 00:45:07.630 --> 00:45:12.110 And then of those 1600, we have a smaller subset

764 00:45:12.110 --> 00:45:17.110 that provide us genome data, as well as spatiotemporal data.

765 00:45:19.630 --> 00:45:24.630 So often people use genome data, say RNA sequences in order

766 00:45:26.640 --> 00:45:29.100 to try to infer the way that different viral cases

767 00:45:29.100 --> 00:45:31.140 are related to each other.

768 00:45:31.140 --> 00:45:34.030 And the question is, can we pull together sequenced

769 00:45:34.030 --> 00:45:36.233 and unsequenced data at the same time?

770 00:45:38.990 --> 00:45:42.170 So what I'm doing here is, again,

771 00:45:42.170 --> 00:45:44.090 I'm not inventing this.

772 00:45:44.090 --> 00:45:46.870 This is something that already exists.

773 00:45:46.870 --> 00:45:51.870 So all that I'm doing is modifying my triggering function G,

774 00:45:52.160 --> 00:45:53.670 and giving it this little N,

775 00:45:53.670 --> 00:45:57.310 this little subscript right there,

776 00:45:57.310 --> 00:46:01.480 which is denoting the fact that I'm allowing different viral

777 00:46:01.480 --> 00:46:04.660 observations to contribute to the rate function

778 00:46:04.660 --> 00:46:05.993 in different manners.

779 00:46:07.180 --> 00:46:09.240 And the exact form that that's gonna take on

780 00:46:09.240 --> 00:46:12.350 for my specific simple model that I'm using,

781 00:46:12.350 --> 00:46:16.560 is I'm going to give this this data N.

782 00:46:16.560 --> 00:46:19.890 And I'm gonna include this data N parameter

783 00:46:19.890 --> 00:46:22.350 in my self excitatory component.

784 00:46:22.350 --> 00:46:26.563 And this data N is restricted to be greater than zero.

785 00:46:27.680 --> 00:46:30.380 So if it is greater than one,

786 00:46:30.380 --> 00:46:33.690 I'm gonna assume that actually, this self excite,

787 00:46:33.690 --> 00:46:37.350 excuse me, that this particular observation,

788 00:46:37.350 --> 00:46:40.820 little N is somehow more contagious.

789 00:46:40.820 --> 00:46:42.660 And if data is less than one,

790 00:46:42.660 --> 00:46:45.333 then I'm going to assume that it's less contagious.

791 00:46:47.870 --> 00:46:51.610 And this is an entirely unsatisfactory part of my talk,

792 00:46:51.610 --> 00:46:56.610 where I'm gonna gloss over a massive part of my model.

793 00:46:57.930 --> 00:47:00.570 And all that I'm gonna say is that

794 00:47:02.030 --> 00:47:05.360 this Phylogenetic Hawkes process, which I'm gonna be telling

795 00:47:05.360 --> 00:47:08.423 you about in the context of big modeling,

796 00:47:09.270 --> 00:47:13.040 and that challenge is that we start

797 00:47:13.040 --> 00:47:16.170 with the phylogenetic tree, which is simply the family tree

798 00:47:16.170 --> 00:47:21.170 that is uniting my 1600 sequenced cases.

799 00:47:21.520 --> 00:47:25.220 And then based on that, actually conditioned on that tree,

800 00:47:25.220 --> 00:47:28.350 we're gonna allow that tree to inform the larger

801 00:47:28.350 --> 00:47:33.350 co-variants of my model parameters, which are then going to

802 00:47:33.390 --> 00:47:36.870 contribute to the overall Hawkes rate function

803 00:47:36.870 --> 00:47:40.043 in a differential manner, although it's still additive.

804 00:47:44.670 --> 00:47:48.560 Now, let's see.

805 00:47:48.560 --> 00:47:51.633 Do I get to go till 10 or 9:50?

806 00:47:56.560 --> 00:47:58.540 <v Man>So you can go till 10.</v>

807 00:47:58.540 --> 00:47:59.770 <v ->Okay, great.</v>

808 00:47:59.770 --> 00:48:04.770 So then, I'll quickly say that if I'm inferring

809 00:48:05.680 --> 00:48:10.197 all of these rates, then I'm inferring over 1300 rates.

810 00:48:12.670 --> 00:48:15.270 So that is actually the dimensionality

811 00:48:15.270 --> 00:48:17.583 of my posterior distribution.

812 00:48:21.270 --> 00:48:23.140 So a tool that I can use,

813 00:48:23.140 --> 00:48:26.150 a classic tool over 50 years old at this point,

814 00:48:26.150 --> 00:48:29.290 that I can use, is I can use the random walk metropolis

815 00:48:29.290 --> 00:48:32.420 algorithm, which is actually going to sample

816 00:48:32.420 --> 00:48:35.830 from the posterior distribution of these rates.

817 00:48:35.830 --> 00:48:40.040 And it's gonna do so in a manner that is effective

818 00:48:40.040 --> 00:48:45.040 in low dimensions, but not effective in high dimensions.

819 00:48:45.950 --> 00:48:47.390 And the way that it works is say,

820 00:48:47.390 --> 00:48:49.230 we start at negative three, negative three.

821 00:48:49.230 --> 00:48:52.380 What we want to do is we want to explore this high density

822 00:48:52.380 --> 00:48:55.320 region of this bi-variate Gaussian,

823 00:48:55.320 --> 00:49:00.233 and we slowly amble forward, and eventually we get there.

824 00:49:02.780 --> 00:49:06.530 But this algorithm breaks down in moderate dimensions.

825 00:49:06.530 --> 00:49:07.363 So.

826 00:49:11.390 --> 00:49:14.060 An algorithm that I think many of us are aware of

827 00:49:14.060 --> 00:49:16.040 at this point, that is kind of a workhorse

828 00:49:16.040 --> 00:49:17.800 in high dimensional Bayesian inference

829 00:49:17.800 --> 00:49:19.880 is Hamiltonian Monte Carlo.

830 00:49:19.880 --> 00:49:23.900 And this works by using actual gradient information about

831 00:49:23.900 --> 00:49:27.520 our log posterior in order to intelligently guide

832 00:49:27.520 --> 00:49:32.140 the MCMC proposals that we're making.

833 00:49:32.140 --> 00:49:34.230 So, again, let's just pretend that we start

834 00:49:34.230 --> 00:49:35.770 at negative three, negative three,

835 00:49:35.770 --> 00:49:37.640 but within a small number of steps,

836 00:49:37.640 --> 00:49:40.110 we're actually effectively exploring

837 00:49:40.110 --> 00:49:43.520 that high density region, and we're doing so

838 00:49:44.550 --> 00:49:47.060 because we're using that gradient information

839 00:49:47.060 --> 00:49:48.403 of the log posterior.

840 00:49:51.230 --> 00:49:55.930 I'm not going to go too deep right now into the formulation

841 00:49:55.930 --> 00:49:59.690 of Hamiltonian Monte Carlo, for the sake of time.

842 00:49:59.690 --> 00:50:04.220 But what I would like to point out,

843 00:50:04.220 --> 00:50:09.220 is that after constructing this kind of physical system

844 00:50:13.462 --> 00:50:18.462 that is based on our target distribution

845 00:50:19.610 --> 00:50:22.423 on the posterior distribution, in some manner,

846 00:50:23.520 --> 00:50:28.520 we actually obtain our proposals within the MCMC.

847 00:50:29.900 --> 00:50:34.900 We obtain the proposals by simulating, by forward simulating

848 00:50:35.130 --> 00:50:39.263 the physical system, according to Hamilton's equations.

849 00:50:40.400 --> 00:50:41.233 Now,

850 00:50:43.400 --> 00:50:48.210 what this simulation involves is a massive number

851 00:50:48.210 --> 00:50:51.323 of repeated gradient evaluations.

852 00:50:53.470 --> 00:50:58.470 Moreover, if the posterior distribution is an ugly one,

853 00:50:59.770 --> 00:51:03.963 that is if it is still conditioned, which we interpret as,

854 00:51:05.670 --> 00:51:09.090 the log posterior Hessian has eigenvalues

855 00:51:09.090 --> 00:51:11.526 that are all over the place.

856 00:51:11.526 --> 00:51:16.526 Then we can also use a mass matrix, M, which is gonna allow

857 00:51:16.828 --> 00:51:21.828 us to condition our dynamics, and make sure that we are

858 00:51:23.610 --> 00:51:27.023 exploring all the dimensions of our model in an even manner.

859 00:51:29.120 --> 00:51:32.100 So the benefit of Hamiltonian Monte-Carlo is that it scales

860 00:51:32.100 --> 00:51:34.030 to tens of thousands of parameters.

861 00:51:34.030 --> 00:51:38.130 But the challenge is that that HMC necessitates repeated

862 00:51:38.130 --> 00:51:39.973 computation at the log likelihood,

863 00:51:42.433 --> 00:51:44.957 it's gradient and then preconditioning.

864 00:51:46.010 --> 00:51:49.330 And the best way that I know to precondition actually

865 00:51:49.330 --> 00:51:53.343 involves evaluating the log likelihood Hessian as well.

866 00:51:54.840 --> 00:51:57.110 And I told you that the challenges that I'm talking about

867 00:51:57.110 --> 00:51:58.340 today are intertwined.

868 00:51:58.340 --> 00:52:00.973 So what does this look like in a big data setting?

869 00:52:02.290 --> 00:52:06.370 Well, we've already managed to speed up the log likelihood

870 00:52:06.370 --> 00:52:09.913 computations that are quadratic in computational complexity.

871 00:52:11.120 --> 00:52:14.080 Well, it turns out that the log likelihood gradient

872 00:52:14.080 --> 00:52:16.760 and the log likelihood Hessian

873 00:52:16.760 --> 00:52:20.830 are all quadratic and computational complexity.

874 00:52:20.830 --> 00:52:24.410 So this means that as the size of our data set grows,

875 00:52:24.410 --> 00:52:25.760 we're going to...

876 00:52:26.720 --> 00:52:31.000 HMC, which is good at scaling to high dimensional models

877 00:52:31.000 --> 00:52:35.250 is going to break down because it's just gonna take too long

878 00:52:35.250 --> 00:52:38.513 to evaluate the quantities that we need to evaluate.

879 00:52:42.510 --> 00:52:45.080 To show you exactly how these parallel

880 00:52:45.080 --> 00:52:47.603 gradient calculations can work.

881 00:52:50.630 --> 00:52:53.290 So, what am I gonna do?

882 00:52:53.290 --> 00:52:55.476 I'm gonna parallelize again on a GPU

883 00:52:55.476 --> 00:53:00.260 or a multi-core CPU implementation,

884 00:53:00.260 --> 00:53:04.350 and I'm interested in evaluating or obtaining

885 00:53:04.350 --> 00:53:06.350 the quantities in the red box.

886 00:53:06.350 --> 00:53:08.670 These are simply the gradient of the log likelihood

887 00:53:08.670 --> 00:53:11.263 with respect to the individual rate parameters.

888 00:53:12.810 --> 00:53:16.780 And because of the summation that it involves,

889 00:53:16.780 --> 00:53:20.520 we actually obtain in the left, top left,

890 00:53:20.520 --> 00:53:24.930 we have the contribution of the first observation

891 00:53:24.930 --> 00:53:28.010 to that gradient term.

892 00:53:28.010 --> 00:53:30.780 Then we have the contribution of the second observation

893 00:53:30.780 --> 00:53:34.730 all the way up to the big int observation,

894 00:53:34.730 --> 00:53:37.090 that contribution to the gradient term.

895 00:53:37.090 --> 00:53:40.970 And these all need to be evaluated and summed over.

896 00:53:40.970 --> 00:53:42.010 So what do we do?

897 00:53:42.010 --> 00:53:44.710 We just do a running total, very simple.

898 00:53:44.710 --> 00:53:47.823 We start by getting the first contribution.

899 00:53:48.790 --> 00:53:51.593 We keep that stored in place.

900 00:53:52.850 --> 00:53:55.560 We evaluate the second contribution,

901 00:53:55.560 --> 00:53:57.380 all at the same time in parallel,

902 00:53:57.380 --> 00:54:01.360 and we simply increment our total observat-

903 00:54:01.360 --> 00:54:04.820 excuse me, our total gradient by that value.

904 00:54:04.820 --> 00:54:05.810 Very simple.

905 00:54:05.810 --> 00:54:07.373 We do this again and again.

906 00:54:08.340 --> 00:54:10.810 Kind of complicated to program, to be honest.

907 00:54:10.810 --> 00:54:11.763 But it's simple.

908 00:54:15.812 --> 00:54:16.645 It's simple when you think about it from the high level.

909 00:54:19.210 --> 00:54:21.370 So I showed you this figure before.

910 00:54:21.370 --> 00:54:24.060 And well, a similar figure before,

911 00:54:24.060 --> 00:54:25.630 and the interpretations are the same,

912 00:54:25.630 --> 00:54:29.910 but here I'll just focus on the question that I received.

913 00:54:29.910 --> 00:54:32.060 In the top left, we have the gradient.

914 00:54:32.060 --> 00:54:33.870 In the bottom left, excuse me,

915 00:54:33.870 --> 00:54:35.160 top row, we have the gradient.

916 00:54:35.160 --> 00:54:36.810 Bottom row, we have the Hessian,

917 00:54:36.810 --> 00:54:41.810 and here I'm increasing to 104 cores.

918 00:54:41.810 --> 00:54:45.970 So this is not infinite cores, right.

919 00:54:45.970 --> 00:54:47.320 It's 104.

920 00:54:47.320 --> 00:54:50.233 But I do want you to see that there's diminishing returns.

921 00:54:54.260 --> 00:54:57.480 And to give a little bit more technical

922 00:54:57.480 --> 00:54:59.093 response to that question,

923 00:55:01.530 --> 00:55:03.940 the thing to bear in mind is that

924 00:55:03.940 --> 00:55:07.700 it's not just about the number of threads that we use.

925 00:55:07.700 --> 00:55:12.170 It's having a lot of RAM very close

926 00:55:12.170 --> 00:55:15.110 to where the computing is being done.

927 00:55:15.110 --> 00:55:18.230 And that is something that GPUs,

928 00:55:18.230 --> 00:55:21.683 modern gigantic GPS do very well.

929 00:55:25.510 --> 00:55:28.470 So why is it important to do all this parallelization?

930 00:55:28.470 --> 00:55:31.890 Well, this is really, I want to kind of communicate

931 00:55:31.890 --> 00:55:34.453 this fact because it is so important.

932 00:55:36.227 --> 00:55:39.710 This slide underlines almost the entire challenge

933 00:55:39.710 --> 00:55:44.210 of big modeling using the spatiotemporal Hawkes process.

934 00:55:44.210 --> 00:55:49.210 The computing to apply this model to the 20,000 plus

935 00:55:49.420 --> 00:55:53.010 data points took about a month

936 00:55:53.950 --> 00:55:57.973 using a very large Nvidia GV100 GPU.

937 00:55:59.930 --> 00:56:01.102 Why?

938 00:56:01.102 --> 00:56:04.410 Because we had to generate 100 million Markov chain states

939 00:56:04.410 --> 00:56:07.993 at a rate of roughly three and a half million each day.

940 00:56:10.890 --> 00:56:14.940 After 100 million Markov chain states,

941 00:56:14.940 --> 00:56:18.303 after generating 100 million Markov chain states,

942 00:56:20.210 --> 00:56:22.740 this is the empirical distribution on the left

943 00:56:22.740 --> 00:56:25.633 of the effective sample sizes across,

944 00:56:27.910 --> 00:56:31.350 across all of the individual rates that we're inferring,

945 00:56:31.350 --> 00:56:33.050 actually all the model parameters.

946 00:56:34.130 --> 00:56:38.710 The minimum is 222, and that's right above my typical

947 00:56:38.710 --> 00:56:42.860 threshold of 200, because in general, we want the effective

948 00:56:42.860 --> 00:56:45.143 sample size to be as large as possible.

949 00:56:47.810 --> 00:56:50.350 Well, why was it so difficult?

950 00:56:50.350 --> 00:56:53.240 Well, a lot of the posterior,

951 00:56:53.240 --> 00:56:55.330 a lot of the marginal posteriors

952 00:56:55.330 --> 00:57:00.330 for our different parameters were very complex.

953 00:57:00.650 --> 00:57:04.950 So for example, here, I just have one individual rate,

954 00:57:04.950 --> 00:57:07.970 and this is the posterior that we learned from it.

955 00:57:07.970 --> 00:57:08.893 It's bi-modal.

956 00:57:09.960 --> 00:57:11.290 And not only is it bi-modal,

957 00:57:11.290 --> 00:57:14.113 but the modes exist on very different scales.

958 00:57:15.640 --> 00:57:19.300 Well, why else is it a difficult posterior to sample from?

959 00:57:19.300 --> 00:57:21.640 Well, because actually, as you might imagine,

960 00:57:21.640 --> 00:57:25.403 these rates have a very complex correlation in structure.

961 00:57:27.753 --> 00:57:29.880 This is kind of repeating something that I said earlier

962 00:57:29.880 --> 00:57:32.623 when we were actually inferring locations,

963 00:57:33.470 --> 00:57:36.330 which is that what this amounts to is really simulating

964 00:57:36.330 --> 00:57:38.593 a very large n-body problem.

965 00:57:43.750 --> 00:57:45.040 But what's the upshot?

966 00:57:45.040 --> 00:57:50.040 Well, we can actually capture these individual rates,

967 00:57:50.730 --> 00:57:55.150 which could give us hints at where to look for certain

968 00:57:55.150 --> 00:58:00.150 mutations that are allowing, say in this example,

969 00:58:00.600 --> 00:58:03.203 the Ebola virus to spread more effectively.

970 00:58:04.560 --> 00:58:08.790 And here, red is generally the highest,

971 00:58:08.790 --> 00:58:10.683 whereas blue is the lowest.

972 00:58:13.270 --> 00:58:14.990 We can get credible intervals,

973 00:58:14.990 --> 00:58:17.740 which can give us another way of thinking about, you know,

974 00:58:17.740 --> 00:58:19.010 where should I be looking

975 00:58:22.258 --> 00:58:26.347 in this collection of viral samples, for the next big one?

976 00:58:28.643 --> 00:58:31.890 And then I can also ask, well, how do these rates actually

977 00:58:31.890 --> 00:58:36.890 distribute along the phylogenetic tree?

978 00:58:37.170 --> 00:58:41.010 So I can look for clades or groups of branches

979 00:58:41.010 --> 00:58:45.577 that are in general, more red in this case than others.

980 00:58:53.270 --> 00:58:55.080 So, something that I...

981 00:58:55.080 --> 00:58:58.143 Okay, so it's 10 o'clock, and I will finish in one slide.

982 00:59:02.610 --> 00:59:04.980 The challenges that I'm talking about today,

983 00:59:04.980 --> 00:59:07.700 they're complex and they're intertwined,

984 00:59:07.700 --> 00:59:09.720 but they're not the only challenges.

985 00:59:09.720 --> 00:59:14.000 There are many challenges in the application

986 00:59:14.000 --> 00:59:16.370 of spatiotemporal Hawkes models,

987 00:59:16.370 --> 00:59:18.673 and there's actually a very large literature.

988 00:59:21.100 --> 00:59:24.560 So some other challenges that we might consider,

989 00:59:24.560 --> 00:59:29.560 and that will also be extremely challenging to overcome

990 00:59:31.270 --> 00:59:32.350 in a big data setting.

991 00:59:32.350 --> 00:59:37.253 So, kind of the first challenge is flexible modeling.

992 00:59:38.150 --> 00:59:40.860 So here, we want to use as flexible

993 00:59:40.860 --> 00:59:43.940 of a Hawkes model as possible.

994 00:59:43.940 --> 00:59:48.940 And this challenge kind of encapsulates one of the great

995 00:59:49.460 --> 00:59:54.210 ironies of model-based nonparametrics, which is that,

996 00:59:55.300 --> 00:59:58.020 the precise time that we actually want to use

997 00:59:58.020 --> 01:00:01.203 a flexible model, is the big data setting.

998 01:00:03.410 --> 01:00:07.200 I mean, I don't know if you recall my earlier slide

999 01:00:07.200 --> 01:00:09.600 where I was showing the posterior distribution

1000 01:00:09.600 --> 01:00:12.850 of some of the length scales associated with

1001 01:00:12.850 --> 01:00:17.673 the Washington DC data, and they're extremely tight.

1002 01:00:19.190 --> 01:00:23.620 But this is actually exactly where we'd want to be able

1003 01:00:23.620 --> 01:00:28.180 to use a flexible model, because no matter what,

1004 01:00:28.180 --> 01:00:31.640 if I apply my model to 85,000 data points,

1005 01:00:31.640 --> 01:00:36.240 I'm going to be very certain in my conclusion,

1006 01:00:36.240 --> 01:00:38.823 conditioned on the specific model that I'm using.

1007 01:00:40.520 --> 01:00:43.000 There's also boundary issues, right.

1008 01:00:43.000 --> 01:00:44.640 This is a huge, a huge thing.

1009 01:00:44.640 --> 01:00:47.030 So for those of you that are aware

1010 01:00:47.030 --> 01:00:50.703 of the survival literature, which I'm sure many of you are,

1011 01:00:51.720 --> 01:00:53.940 you know, they're censoring.

1012 01:00:53.940 --> 01:00:56.740 So what about gunshots that occurred right outside

1013 01:00:56.740 --> 01:01:00.700 of the border of Washington DC, and it occurred as a result

1014 01:01:00.700 --> 01:01:03.280 of gunshots that occurred within the border?

1015 01:01:03.280 --> 01:01:05.330 And then we can flip that on its head.

1016 01:01:05.330 --> 01:01:10.100 What about parent events outside of Washington DC

1017 01:01:10.100 --> 01:01:13.450 that precipitated gun violence within Washington DC.

1018 01:01:13.450 --> 01:01:15.710 And then finally, sticking with the same example,

1019 01:01:15.710 --> 01:01:16.810 differential sampling.

1020 01:01:20.120 --> 01:01:25.120 You can be certain that those acoustic gunshot locators,

1021 01:01:26.880 --> 01:01:30.320 location system sensors are not planted

1022 01:01:30.320 --> 01:01:32.343 all over Washington DC.

1023 01:01:34.210 --> 01:01:36.603 And how does their distribution affect things?

1024 01:01:41.010 --> 01:01:41.843 Okay.

1025 01:01:41.843 --> 01:01:44.550 This is joint work with Mark Suchard at UCLA, also at UCLA.

1026 01:01:44.550 --> 01:01:46.530 And then my very good friend,

1027 01:01:46.530 --> 01:01:49.990 my very dear friend, Xiang Ji at Tulane.

1028 01:01:49.990 --> 01:01:54.240 It's funded by the K-Award Big Data Predictive Phylogenetics

1029 01:01:54.240 --> 01:01:57.840 with Bayesian learning, funded by the NIH.

1030 01:01:57.840 --> 01:01:58.920 line:15% And that's it.

1031 01:01:58.920 --> 01:01:59.753 line:15% Thank you.

1032 01:02:05.640 --> 01:02:06.685 <v Man>All right.</v>

1033 01:02:06.685 --> 01:02:07.950 Thank you so much, Professor Holbrook.

1034 01:02:07.950 --> 01:02:11.349 Does anybody have any other questions?

1035 01:02:11.349 --> 01:02:15.266 (people speaking indistinctly)

1036 01:02:18.070 --> 01:02:18.903 Yeah.

1037 01:02:20.970 --> 01:02:25.316 Any other questions from the room here, or from Zoom?

1038 01:02:25.316 --> 01:02:27.375 (people speaking indistinctly)