WEBVTT

1 00:00:04.280 --> 00:00:06.020 - So, hi everyone.

2 00:00:06.020 --> 00:00:08.267 Since we're still waiting for people to join,

3 00:00:08.267 --> 00:00:10.770 I will first give a brief introduction

4 00:00:10.770 --> 00:00:11.913 to Matthew here.

5 00:00:13.490 --> 00:00:16.080 First, it's my honor to introduce Dr. Matthew Stevens

6 00:00:16.080 --> 00:00:18.210 as our seminar speaker today.

7 00:00:18.210 --> 00:00:21.190 And Matthew is a professor from human genetics

8 00:00:21.190 --> 00:00:24.320 and the statistics at University of Chicago.

9 00:00:24.320 --> 00:00:26.640 And in the past, his research mainly focused

10 00:00:26.640 --> 00:00:28.550 on developing new statistical methods

11 00:00:28.550 --> 00:00:31.670 for especially genetic applications.

12 00:00:31.670 --> 00:00:33.230 Including, for example,

13 00:00:33.230 --> 00:00:35.590 GWAS association studies and fine mapping

14 00:00:35.590 --> 00:00:37.750 and populations genetic variants.

15 00:00:37.750 --> 00:00:39.580 And today, he will give a talk

16 00:00:39.580 --> 00:00:43.490 on some recently developed empirical Bayse methods

17 00:00:43.490 --> 00:00:45.707 for the estimation for normal mean models

18 00:00:45.707 --> 00:00:48.140 that will introduce the (indistinct) properties

19 00:00:48.140 --> 00:00:49.433 such as shrinkage,

20 00:00:49.433 --> 00:00:51.180 sparsity or smoothness.

21 00:00:51.180 --> 00:00:53.910 And he will also discuss how to apply these methods

22 00:00:53.910 --> 00:00:56.203 to a range of practical applications.

23 00:00:58.560 --> 00:01:02.110 Okay, so let's wait for another minute

24 00:01:02.110 --> 00:01:04.150 and then I will hand it over to Matthew.

25 00:01:58.814 --> 00:02:02.030 So I will hand it over to Matthew from here.

26 00:02:02.030 --> 00:02:03.007 Let's welcome him.

27 00:02:04.650 --> 00:02:05.483 - Thank you very much.

28 00:02:05.483 --> 00:02:07.370 It's a great pleasure to be here

29 00:02:07.370 --> 00:02:08.660 and to get the opportunity

30 00:02:08.660 --> 00:02:11.320 to present my work to you today.

31 00:02:11.320 --> 00:02:13.830 So I guess just a little bit of background.

32 00:02:13.830 --> 00:02:16.290 So few years ago...

33 00:02:18.010 --> 00:02:20.180 Well, I guess, I've been teaching sparsity

34 00:02:20.180 --> 00:02:22.380 and shrinkage for a while,

35 00:02:22.380 --> 00:02:24.120 and it struck me that, in practice,

36 00:02:24.120 --> 00:02:28.790 people don't really use many of these ideas directly...

37 00:02:28.790 --> 00:02:30.430 At least not the empirical Bayse versions

38 00:02:30.430 --> 00:02:31.860 of these ideas

39 00:02:31.860 --> 00:02:33.630 directly in applications.

40 00:02:33.630 --> 00:02:37.023 And so, I'm wondering why that is.

41 00:02:37.023 --> 00:02:40.209 And partly, it's the lack of...

42 00:02:40.209 --> 00:02:42.900 User-friendly, convenient methods

43 00:02:42.900 --> 00:02:44.090 for applying these ideas.

44 00:02:44.090 --> 00:02:46.080 So I've been trying to think about

45 00:02:46.080 --> 00:02:50.680 how we can make these powerful ideas and methods

46 00:02:50.680 --> 00:02:53.440 more generally applicable or easily applicable

47 00:02:53.440 --> 00:02:54.470 in applications.

48 00:02:54.470 --> 00:02:57.153 These ideas have been around quite some time.

49 00:02:57.153 --> 00:02:59.180 But I think we've made some progress

50 00:02:59.180 --> 00:03:01.860 on actually just making them a bit simpler maybe

51 00:03:01.860 --> 00:03:03.560 and simpler to apply in practice,

52 00:03:03.560 --> 00:03:05.513 so I'm gonna tell you about those today.

53 00:03:07.740 --> 00:03:08.573 Oh, sorry.

54 00:03:08.573 --> 00:03:10.100 It's not advancing, let me see.

55 00:03:10.100 --> 00:03:13.360 Okay, so yeah, kind of related to that,

56 00:03:13.360 --> 00:03:15.120 the normal means problem is

57 00:03:15.120 --> 00:03:18.153 something we teach quite frequently.

58 00:03:18.153 --> 00:03:19.940 It’s not hard to teach

59 00:03:19.940 --> 00:03:22.755 but it always struck me whenever I was taught it

60 00:03:22.755 --> 00:03:24.020 that it looked like a kind of

61 00:03:24.020 --> 00:03:28.373 a toy model that statisticians kind of think up

62 00:03:28.373 --> 00:03:30.530 to teach students things

63 00:03:30.530 --> 00:03:32.220 but never actually use.

64 00:03:32.220 --> 00:03:35.160 And then suddenly, I had an epiphany

65 00:03:35.160 --> 00:03:37.280 and realized that it’s super useful.

66 00:03:37.280 --> 00:03:38.403 And so now, I’m trying to...

67 00:03:38.403 --> 00:03:41.580 I’m not the only one but I’m trying to convince people

68 00:03:41.580 --> 00:03:44.100 that actually, this is a super useful thing

69 00:03:44.100 --> 00:03:45.820 that we should be using in practice.

70 00:03:45.820 --> 00:03:47.470 So here’s the normal means model.

71 00:03:48.510 --> 00:03:49.700 The idea is that you’ve got

72 00:03:49.700 --> 00:03:51.990 a bunch of observations, XJ,

73 00:03:51.990 --> 00:03:54.200 that you can think of as noisy observations

74 00:03:54.200 --> 00:03:55.331 of theta J

75 00:03:55.331 --> 00:03:57.792 and they have some variance.

76 00:03:57.792 --> 00:04:00.263 I’m going to allow each variance to be different.

77 00:04:02.270 --> 00:04:04.370 The simplest version would be to assume

78 00:04:04.370 --> 00:04:05.990 that the variances are all the same

79 00:04:05.990 --> 00:04:07.923 but I’m going to allow them to be different.

80 00:04:09.793 --> 00:04:11.260 But an important point is

81 00:04:11.260 --> 00:04:13.560 that we’re going to assume that the variance is unknown,

82 00:04:13.560 --> 00:04:15.000 which sounds a bit weird

83 00:04:15.000 --> 00:04:17.320 but in applications, we’ll see that there are reasons

84 00:04:17.320 --> 00:04:20.950 why we might think that’s an okay assumption

85 00:04:20.950 --> 00:04:22.468 in some applications.

86 00:04:22.468 --> 00:04:25.040 Okay, so the basic idea is

87 00:04:25.040 --> 00:04:26.140 you've got a bunch of measurements

88 00:04:26.140 --> 00:04:27.350 that are noisy measurements

89 00:04:27.350 --> 00:04:29.180 of sum theta J

90 00:04:29.180 --> 00:04:30.980 and they have known variance,

91 00:04:30.980 --> 00:04:33.460 so they have known precision essentially,

92 00:04:33.460 --> 00:04:36.050 and you want to estimate the Theta Js.

93 00:04:36.050 --> 00:04:37.580 And, of course, the MLE is just

94 00:04:37.580 --> 00:04:38.890 to estimate see theta J

95 00:04:38.890 --> 00:04:42.750 by its corresponding measurement, XJ.

96 00:04:42.750 --> 00:04:45.670 And really, it was a big surprise, I think.

97 00:04:45.670 --> 00:04:46.790 I wasn't around at the time

98 00:04:46.790 --> 00:04:49.420 but I believe it was a big surprise in 1956

99 00:04:49.420 --> 00:04:52.320 when Stein showed that you can do better

100 00:04:52.320 --> 00:04:54.150 than the MLE, at least in terms of

101 00:04:55.185 --> 00:04:58.083 average squared error expected square there.

102 00:05:01.160 --> 00:05:05.240 And so, there are many different ways to motivate or...

103 00:05:05.240 --> 00:05:06.942 To motivate this result.

104 00:05:06.942 --> 00:05:11.180 And I think many of them end up not being that intuitive.

105 00:05:11.180 --> 00:05:14.512 It is quite a surprising result in generality

106 00:05:14.512 --> 00:05:15.910 but I think...

107 00:05:15.910 --> 00:05:17.790 So the way I like to think about the intuition

108 00:05:17.790 --> 00:05:19.240 for why this might be true,

109 00:05:19.240 --> 00:05:20.290 it's not the only intuition

110 00:05:20.290 --> 00:05:22.390 but it's one intuition for why this might be true,

111 00:05:22.390 --> 00:05:26.500 is to have an empirical Bayse thinking to the problem.

112 00:05:26.500 --> 00:05:29.500 And so, to illustrate this idea,

113 00:05:29.500 --> 00:05:33.320 I use a well-worn device, at this point,

114 00:05:33.320 --> 00:05:35.953 which is baseball batting averages.

115 00:05:37.920 --> 00:05:40.040 Efron certainly has used this example before

116 00:05:40.040 --> 00:05:42.310 to motivate empirical Bayse ideas.

117 00:05:42.310 --> 00:05:44.210 This particular example comes from...

118 00:05:44.210 --> 00:05:46.030 The data come from this block here,

119 00:05:46.030 --> 00:05:47.650 that I referenced at the bottom,

120 00:05:47.650 --> 00:05:49.200 which I quite like as an explanation

121 00:05:49.200 --> 00:05:51.970 of basic ideas behind empirical Bayse.

122 00:05:51.970 --> 00:05:54.410 So this histogram here shows a bunch

123 00:05:54.410 --> 00:05:56.530 of basic baseball batting averages

124 00:05:56.530 --> 00:05:59.220 for a particular season in 1900.

125 00:05:59.220 --> 00:06:00.940 You don't need to know very much about baseball

126 00:06:00.940 --> 00:06:02.340 to know what's going on here.

127 00:06:02.340 --> 00:06:05.550 Essentially, in baseball, you go and try and hit a ball

128 00:06:05.550 --> 00:06:06.520 and your batting average is

129 00:06:06.520 --> 00:06:08.200 what proportion of the time

130 00:06:08.200 --> 00:06:11.760 you as a bat person end up hitting the ball.

131 00:06:11.760 --> 00:06:15.613 And a good baseball batting average is around 0.3 or so.

132 00:06:16.740 --> 00:06:19.090 And in a professional baseball,

133 00:06:19.090 --> 00:06:21.330 no one's really going to have a batting average of zero

134 00:06:21.330 --> 00:06:22.863 'cause they wouldn't survive.

135 00:06:25.362 --> 00:06:28.390 But empirically, there were some individuals in this season

136 00:06:28.390 --> 00:06:29.680 who had a batting average of zero,

137 00:06:29.680 --> 00:06:32.940 that is they completely failed to hit the ball every time

138 00:06:32.940 --> 00:06:34.260 they went up to bat.

139 00:06:34.260 --> 00:06:35.830 And there were some people

140 00:06:35.830 --> 00:06:38.370 who had a batting average of above 0.4,

141 00:06:38.370 --> 00:06:42.755 which is also completely unheard of in baseball.

142 00:06:42.755 --> 00:06:45.260 Nobody has a batting average that high,

143 00:06:45.260 --> 00:06:46.961 so what’s going on here?

144 00:06:46.961 --> 00:06:48.720 Well, it’s a simple explanation is

145 00:06:48.720 --> 00:06:51.650 that these individuals at the tails are individuals

146 00:06:51.650 --> 00:06:53.270 who just had a few at-bats.

147 00:06:53.270 --> 00:06:54.990 They only went and attempted

148 00:06:54.990 --> 00:06:57.470 to hit the ball a small number of times.

149 00:06:57.470 --> 00:07:00.650 And so, maybe these individuals only had two bats

150 00:07:00.650 --> 00:07:02.370 and they missed it both times,

151 00:07:02.370 --> 00:07:04.300 they got injured or they weren’t selected

152 00:07:04.300 --> 00:07:05.690 or, for whatever reason, they didn’t hit

153 00:07:05.690 --> 00:07:06.810 the ball many times...

154 00:07:06.810 --> 00:07:08.600 They didn’t go to at bat many times

155 00:07:08.600 --> 00:07:12.255 and so, their batting average was empirically zero.

156 00:07:12.255 --> 00:07:15.490 Think of that as the maximum likelihood estimate.

157 00:07:15.490 --> 00:07:16.900 But if you wanted to predict

158 00:07:16.900 --> 00:07:19.000 what they would do, say next season,

159 00:07:19.000 --> 00:07:20.930 if you gave them more at bats

160 00:07:20.930 --> 00:07:21.980 in the long run,

161 00:07:21.980 --> 00:07:26.753 zero would be a bad estimate for obvious reasons.

162 00:07:26.753 --> 00:07:29.520 And the same applies to these individuals up here

163 00:07:29.520 --> 00:07:30.790 with very big batting averages.

164 00:07:30.790 --> 00:07:33.687 They also had relatively few at-bats

165 00:07:33.687 --> 00:07:38.300 and they just happened to hit it above 0.4 of the time

166 00:07:38.300 --> 00:07:39.490 out of the at-bats.

167 00:07:39.490 --> 00:07:41.370 And the individuals who had lots of at-bats are

168 00:07:41.370 --> 00:07:43.770 all in the middle here.

169 00:07:43.770 --> 00:07:45.487 So these are binomial observations, basically,

170 00:07:45.487 --> 00:07:47.193 and the ones who have small N are

171 00:07:48.690 --> 00:07:50.000 more likely to be in the tails

172 00:07:50.000 --> 00:07:50.930 and the ones we're big N are

173 00:07:50.930 --> 00:07:53.470 all going to be around in the middle here.

174 00:07:53.470 --> 00:07:54.303 So what would we do?

175 00:07:54.303 --> 00:07:55.450 What would we want to do

176 00:07:55.450 --> 00:07:57.230 if we wanted to estimate,

177 00:07:57.230 --> 00:07:59.340 for example, for this individual,

178 00:07:59.340 --> 00:08:01.190 their batting average for next season?

179 00:08:01.190 --> 00:08:04.344 If we were gonna predict what they were gonna get.

180 00:08:04.344 --> 00:08:08.100 Well, we would definitely want to estimate

181 00:08:08.100 --> 00:08:11.683 something closer to the average batting average than 04.

182 00:08:12.840 --> 00:08:14.220 That's the intuition.

183 00:08:14.220 --> 00:08:17.620 And one way to frame that problem is that...

184 00:08:17.620 --> 00:08:18.453 So sorry.

185 00:08:18.453 --> 00:08:20.420 So this is the basic idea of shrinkage.

186 00:08:20.420 --> 00:08:22.680 We would want to shrink these estimates towards,

187 00:08:22.680 --> 00:08:24.230 in this case, towards the mean.

188 00:08:25.300 --> 00:08:26.823 So how are we gonna do that?

189 00:08:26.823 --> 00:08:30.748 Well, one way to think about it is...

190 00:08:30.748 --> 00:08:33.300 Sorry, let me just...

191 00:08:33.300 --> 00:08:34.133 Yes.

192 00:08:35.630 --> 00:08:37.330 Sorry, just getting my slides.

193 00:08:37.330 --> 00:08:40.230 Okay, so here, the red line represents

194 00:08:40.230 --> 00:08:43.650 some underlying distribution

195 00:08:43.650 --> 00:08:45.710 of actual batting averages.

196 00:08:45.710 --> 00:08:47.280 So conceptually, some distribution

197 00:08:47.280 --> 00:08:50.350 of actual batting averages among individuals

198 00:08:52.540 --> 00:08:53.950 in this kind of league.

199 00:08:53.950 --> 00:08:57.360 So the red line, in a Bayesean point of view,

200 00:08:57.360 --> 00:08:59.830 kind of represent a sensible prior

201 00:08:59.830 --> 00:09:01.570 for any given individual's batting average

202 00:09:01.570 --> 00:09:03.321 before we saw that data.

203 00:09:03.321 --> 00:09:05.160 So think of the red line as representing

204 00:09:05.160 --> 00:09:07.900 the variation or the distribution

205 00:09:07.900 --> 00:09:12.410 of actual batting averages among players.

206 00:09:12.410 --> 00:09:17.110 And in fact, what we've done here is estimate

207 00:09:17.110 --> 00:09:20.560 that red line from the data.

208 00:09:20.560 --> 00:09:22.580 That's the empirical Bayse part

209 00:09:22.580 --> 00:09:24.640 of the empirical Bayse.

210 00:09:24.640 --> 00:09:26.860 The empirical part of empirical Bayse is that

211 00:09:26.860 --> 00:09:28.170 the red line which we're going to use

212 00:09:28.170 --> 00:09:30.230 as a prior for any given player was

213 00:09:30.230 --> 00:09:32.590 actually estimated from all the data.

214 00:09:32.590 --> 00:09:33.890 And the basic idea is

215 00:09:33.890 --> 00:09:36.260 because we know what the variance

216 00:09:36.260 --> 00:09:38.220 of a binomial distribution is,

217 00:09:38.220 --> 00:09:40.080 we can kind of estimate

218 00:09:40.080 --> 00:09:41.870 what the overall distribution

219 00:09:41.870 --> 00:09:46.159 of the underlying piece in this binomial look like,

220 00:09:46.159 --> 00:09:49.120 taking account of the fact that the histogram is

221 00:09:49.120 --> 00:09:53.370 a noisy observations of that underlying P.

222 00:09:53.370 --> 00:09:54.203 Every bat...

223 00:09:54.203 --> 00:09:57.620 Basically, every every estimated batting average is

224 00:09:57.620 --> 00:09:59.980 a noisy estimate of the true batting average

225 00:09:59.980 --> 00:10:00.920 with the noise depending on

226 00:10:00.920 --> 00:10:02.980 how many at-bats they have.

227 00:10:02.980 --> 00:10:05.133 So once we've estimated that red line,

228 00:10:06.570 --> 00:10:07.580 that prior,

229 00:10:07.580 --> 00:10:09.960 we can compute the posterior

230 00:10:09.960 --> 00:10:12.350 for each individual based on that prior.

231 00:10:12.350 --> 00:10:13.183 And when we do that,

232 00:10:13.183 --> 00:10:16.110 this is a histogram of the posterior means.

233 00:10:16.110 --> 00:10:17.710 So these are, if you like,

234 00:10:17.710 --> 00:10:20.340 shrunken estimates of the batting average

235 00:10:20.340 --> 00:10:21.173 for each individual.

236 00:10:21.173 --> 00:10:22.250 And you can see that the individuals

237 00:10:22.250 --> 00:10:24.900 who had zero at-bats got shrunk

238 00:10:24.900 --> 00:10:27.120 all the way over somewhere here.

239 00:10:27.120 --> 00:10:29.340 And that's because their data really...

240 00:10:29.340 --> 00:10:31.800 Although, the point estimate was zero,

241 00:10:31.800 --> 00:10:32.830 they had very few at bats.

242 00:10:32.830 --> 00:10:37.830 So the information in that data are very slim,

243 00:10:38.010 --> 00:10:39.790 very little information.

244 00:10:39.790 --> 00:10:41.210 And so, the prior dominates

245 00:10:41.210 --> 00:10:43.230 when you're looking at the posterior distribution

246 00:10:43.230 --> 00:10:44.480 for these individuals.

247 00:10:44.480 --> 00:10:45.720 Whereas individuals in the middle

248 00:10:45.720 --> 00:10:46.553 who have more at-bats,

249 00:10:46.553 --> 00:10:51.553 will have the estimate that is less shrunken.

250 00:10:51.710 --> 00:10:54.203 So that's gonna be a theme we'll come back to later.

251 00:10:55.370 --> 00:10:57.490 So how do we form...

252 00:10:57.490 --> 00:10:58.720 That's a picture.

253 00:10:58.720 --> 00:11:00.380 How do we formulate that?

254 00:11:00.380 --> 00:11:02.570 So those were binomial data,

255 00:11:02.570 --> 00:11:04.790 I'm gonna talk about normal data.

256 00:11:04.790 --> 00:11:07.700 So don't get confused by that.

257 00:11:07.700 --> 00:11:09.730 I'm just going to assume normality could do

258 00:11:09.730 --> 00:11:11.950 the same thing for a binomial,

259 00:11:11.950 --> 00:11:16.390 but I think the normals a more generally useful

260 00:11:16.390 --> 00:11:18.833 and convenient way to go.

261 00:11:19.960 --> 00:11:23.060 So here's a normal means model again

262 00:11:23.060 --> 00:11:24.530 and the idea is that that

263 00:11:24.530 --> 00:11:26.370 we're going to assume that thetas come

264 00:11:26.370 --> 00:11:28.530 from some prior distribution, G,

265 00:11:28.530 --> 00:11:30.760 that was the red line in my example,

266 00:11:30.760 --> 00:11:32.580 and we're going to estimate G

267 00:11:32.580 --> 00:11:34.240 by maximum likelihood essentially.

268 00:11:34.240 --> 00:11:36.333 So we're going to use all the X's,

269 00:11:36.333 --> 00:11:37.910 integrating out theta

270 00:11:37.910 --> 00:11:40.430 to obtain a maximum likelihood estimate for G.

271 00:11:40.430 --> 00:11:42.180 That's stage one,

272 00:11:42.180 --> 00:11:43.380 that's estimating that red line.

273 00:11:43.380 --> 00:11:44.450 And then stage two is

274 00:11:44.450 --> 00:11:46.160 to compute the posterior distribution

275 00:11:46.160 --> 00:11:48.110 for each batting average,

276 00:11:48.110 --> 00:11:50.720 or whatever theta J we're interested in,

277 00:11:50.720 --> 00:11:52.950 taking into account that estimated prior

278 00:11:52.950 --> 00:11:55.904 and the data on the individual J.

279 00:11:55.904 --> 00:12:00.560 So that's the formalization of these ideas.

280 00:12:00.560 --> 00:12:02.636 And these posterior distributions are gonna be shrunk

281 00:12:02.636 --> 00:12:05.553 towards the prior or the primary.

282 00:12:07.380 --> 00:12:09.200 So what kind of...

283 00:12:09.200 --> 00:12:12.340 So I guess I've left unspecified here,

284 00:12:12.340 --> 00:12:15.823 what family of priors should we consider for G?

285 00:12:17.370 --> 00:12:20.610 So a commonly used prior distribution is

286 00:12:20.610 --> 00:12:22.610 this so-called point-normal,

287 00:12:22.610 --> 00:12:26.593 or sometimes called spike and slab prior distribution.

288 00:12:27.520 --> 00:12:28.353 And these are...

289 00:12:28.353 --> 00:12:29.186 Sorry, I should say,

290 00:12:29.186 --> 00:12:30.630 I'm going to be thinking a lot about problems

291 00:12:30.630 --> 00:12:32.810 where we want to induce sparsity.

292 00:12:32.810 --> 00:12:36.560 So in baseball, we were shrinking towards the mean

293 00:12:36.560 --> 00:12:38.170 but in many applications,

294 00:12:38.170 --> 00:12:39.890 the natural point towards

295 00:12:39.890 --> 00:12:42.294 natural prime mean, if you like, is zero

296 00:12:42.294 --> 00:12:45.530 in situations where we expect effects

297 00:12:45.530 --> 00:12:47.200 to be sparse, for example.

298 00:12:47.200 --> 00:12:49.580 So I'm gonna be talking mostly about that situation,

299 00:12:49.580 --> 00:12:51.260 although the ideas are more general.

300 00:12:51.260 --> 00:12:52.260 And so, I'm going to be focusing

301 00:12:52.260 --> 00:12:56.270 on the sparsity inducing choices of prior family.

302 00:12:56.270 --> 00:12:59.210 And so, one commonly used one is this point normal

303 00:12:59.210 --> 00:13:02.400 where there's some mass pi zero

304 00:13:02.400 --> 00:13:03.750 exactly at zero,

305 00:13:03.750 --> 00:13:05.810 and then the rest of the mass is

306 00:13:05.810 --> 00:13:08.073 normally distributed about zero.

307 00:13:09.020 --> 00:13:11.450 So the commonly used one.

308 00:13:11.450 --> 00:13:12.670 In fact, it turns out,

309 00:13:12.670 --> 00:13:14.610 and this is kind of interesting I think,

310 00:13:14.610 --> 00:13:17.150 that it can be easier to do the computations

311 00:13:17.150 --> 00:13:19.644 for more general families.

312 00:13:19.644 --> 00:13:22.010 So for example,

313 00:13:22.010 --> 00:13:23.900 just take the non-parametric family

314 00:13:23.900 --> 00:13:26.590 that's the zero-centered scale mixture of normal,

315 00:13:26.590 --> 00:13:28.160 so we'll see that in it,

316 00:13:28.160 --> 00:13:31.410 which includes all these distributions of special cases.

317 00:13:31.410 --> 00:13:32.510 It's nonparametric.

318 00:13:32.510 --> 00:13:34.970 It includes a point-normal here.

319 00:13:34.970 --> 00:13:36.447 It also includes the T-distribution,

320 00:13:36.447 --> 00:13:37.680 the Laplace distribution,

321 00:13:37.680 --> 00:13:39.543 the horseshoe prior, if you've come across that,

322 00:13:39.543 --> 00:13:41.800 this zero-centered scale mixture of normals

323 00:13:41.800 --> 00:13:44.160 and the surprise is that it turns out

324 00:13:45.070 --> 00:13:47.120 to be easier, in some sense,

325 00:13:47.120 --> 00:13:49.080 to do the calculations for this family,

326 00:13:49.080 --> 00:13:50.060 this more general family,

327 00:13:50.060 --> 00:13:51.700 than this narrow family,

328 00:13:51.700 --> 00:13:53.280 partly because of the convex family.

329 00:13:53.280 --> 00:13:56.920 So you can think of this as a kind of a convex relaxation

330 00:13:56.920 --> 00:13:57.753 of the problem.

331 00:13:57.753 --> 00:13:59.010 So all the computations become...

332 00:13:59.010 --> 00:14:00.930 The optimizations you have to do in the simplest case

333 00:14:00.930 --> 00:14:04.589 become convex when you use this family.

334 00:14:04.589 --> 00:14:07.470 So let me say a bit more about that

335 00:14:07.470 --> 00:14:08.730 for the non-parametric.

336 00:14:08.730 --> 00:14:11.870 How do we actually do these non-parametric computations?

337 00:14:11.870 --> 00:14:13.700 Well, we actually approximate

338 00:14:13.700 --> 00:14:17.470 the non-parametric computation using a grid idea.

339 00:14:17.470 --> 00:14:19.640 So here's the idea.

340 00:14:19.640 --> 00:14:21.620 We modeled G, our prior,

341 00:14:21.620 --> 00:14:23.052 as a mixture of...

342 00:14:23.052 --> 00:14:25.260 I like to think of this K as being big.

343 00:14:25.260 --> 00:14:28.000 A large number of normal distributions.

344 00:14:28.000 --> 00:14:30.690 All of these normal distributions are centered at zero,

345 00:14:30.690 --> 00:14:31.910 that's this zero here,

346 00:14:31.910 --> 00:14:33.880 and they have a different variance.

347 00:14:33.880 --> 00:14:36.000 Some of them have very small variances,

348 00:14:36.000 --> 00:14:37.840 perhaps even one of them has a zero variance,

349 00:14:37.840 --> 00:14:39.440 so that's the point mass at zero.

350 00:14:39.440 --> 00:14:40.810 And the variance is sigma...

351 00:14:40.810 --> 00:14:43.480 Think of Sigma squared K getting gradually bigger

352 00:14:43.480 --> 00:14:46.910 until the last Sigma squared K is very big.

353 00:14:46.910 --> 00:14:48.450 So we're just gonna use a lot of them.

354 00:14:48.450 --> 00:14:50.360 Think of K as being, let's say 100

355 00:14:50.360 --> 00:14:52.130 or 1,000 for the...

356 00:14:52.130 --> 00:14:53.810 In practice, we find 20 is enough

357 00:14:53.810 --> 00:14:56.560 but just think of it as being big

358 00:14:56.560 --> 00:14:58.620 and spanning a lot of different variances,

359 00:14:58.620 --> 00:14:59.780 going from very, very small,

360 00:14:59.780 --> 00:15:01.429 to very, very big.

361 00:15:01.429 --> 00:15:04.960 And then, estimating G just comes down

362 00:15:04.960 --> 00:15:06.390 to estimating these pis,

363 00:15:06.390 --> 00:15:07.913 these mixture proportions.

364 00:15:08.770 --> 00:15:10.660 And that, then of course,

365 00:15:10.660 --> 00:15:13.200 is a finite dimensional optimization problem

366 00:15:13.200 --> 00:15:14.910 and in the normal means model,

367 00:15:14.910 --> 00:15:15.930 it's a convex...

368 00:15:15.930 --> 00:15:17.610 Well actually, for any mixture,

369 00:15:17.610 --> 00:15:19.120 it's a convex problem,

370 00:15:19.120 --> 00:15:21.981 and so there are efficient ways to find

371 00:15:21.981 --> 00:15:26.981 the MLE for pi, given the grid of variances.

372 00:15:27.470 --> 00:15:29.840 So let's just illustrate what's going on here.

373 00:15:29.840 --> 00:15:33.100 Here's a grid of just three normals.

374 00:15:33.100 --> 00:15:34.777 The one in the middle has the smallest variance,

375 00:15:34.777 --> 00:15:36.767 the one over here has the biggest variance.

376 00:15:36.767 --> 00:15:38.880 And we can get a mixture of those,

377 00:15:38.880 --> 00:15:39.927 looks like that.

378 00:15:39.927 --> 00:15:42.440 So you can see this is kind of a spiky distribution

379 00:15:42.440 --> 00:15:43.690 but also with a long tail,

380 00:15:43.690 --> 00:15:47.200 even with just a mixture of three distributions.

381 00:15:47.200 --> 00:15:49.120 And so, the idea is that you can get

382 00:15:49.120 --> 00:15:50.770 quite a flex...

383 00:15:50.770 --> 00:15:51.730 It's a flexible family

384 00:15:51.730 --> 00:15:55.921 by using a larger number of variances than three.

385 00:15:55.921 --> 00:15:58.860 You can imagine you can get distributions

386 00:15:58.860 --> 00:16:01.210 that have all sorts of spikiness

387 00:16:01.210 --> 00:16:03.123 and long-tailed behavior.

388 00:16:06.284 --> 00:16:09.250 So maybe just to fill in the details here;

389 00:16:09.250 --> 00:16:12.500 with that prior as a mixture of normals,

390 00:16:12.500 --> 00:16:14.746 the marginal distribution, P of X,

391 00:16:14.746 --> 00:16:17.930 integrating out theta is analytic

392 00:16:17.930 --> 00:16:20.500 because the sum of normals is normal.

393 00:16:20.500 --> 00:16:23.160 So if you have a normally distributed variable

394 00:16:23.160 --> 00:16:24.620 and then you have another variable

395 00:16:24.620 --> 00:16:26.843 that's a normal error on top of that,

396 00:16:26.843 --> 00:16:28.420 you get a normal.

397 00:16:28.420 --> 00:16:31.910 So the marginal is a mixture of normals

398 00:16:31.910 --> 00:16:34.380 that's very simple to work with

399 00:16:34.380 --> 00:16:38.150 and estimating pi is a convex optimization problem.

400 00:16:38.150 --> 00:16:38.983 You can do it.

401 00:16:38.983 --> 00:16:39.950 You can do an EM algorithm

402 00:16:39.950 --> 00:16:41.310 but convex methods,

403 00:16:41.310 --> 00:16:43.330 as pointed out by Koenker and Mizera,

404 00:16:43.330 --> 00:16:45.513 can be a lot more reliable and faster.

405 00:16:49.350 --> 00:16:52.380 Okay, so let's just illustrate those ideas again.

406 00:16:52.380 --> 00:16:56.200 Here's a potential prior distribution

407 00:16:57.660 --> 00:16:59.780 and here's a likelihood.

408 00:16:59.780 --> 00:17:01.680 So this is like a likelihood from a normal...

409 00:17:01.680 --> 00:17:03.673 This is an estimate...

410 00:17:03.673 --> 00:17:05.310 Think of this as a likelihood

411 00:17:05.310 --> 00:17:08.170 for theta J in a normal means model.

412 00:17:08.170 --> 00:17:10.870 So maybe XJ was one and a half or something

413 00:17:10.870 --> 00:17:12.210 and SJ was, I don't know,

414 00:17:12.210 --> 00:17:13.766 something like a half or something

415 00:17:13.766 --> 00:17:14.666 or a half squared.

416 00:17:17.092 --> 00:17:19.420 So this is meant to represent the likelihood.

417 00:17:19.420 --> 00:17:20.840 So what does the posterior look like

418 00:17:20.840 --> 00:17:23.010 when we combine this prior,

419 00:17:23.010 --> 00:17:23.843 the black line,

420 00:17:23.843 --> 00:17:25.810 with this likelihood, the red line?

421 00:17:25.810 --> 00:17:27.564 it looks like this green line here.

422 00:17:27.564 --> 00:17:30.680 So what you can see is going on here is

423 00:17:30.680 --> 00:17:34.410 that you get shrinkage towards the mean, right?

424 00:17:34.410 --> 00:17:37.266 But because the black line is long-tailed

425 00:17:37.266 --> 00:17:39.900 because of the prior in this case has a long tail,

426 00:17:39.900 --> 00:17:40.998 and because the red line...

427 00:17:40.998 --> 00:17:44.364 The likelihood lies quite a ways in the tail,

428 00:17:44.364 --> 00:17:47.810 the spiky bit at zero doesn't have very much impact

429 00:17:47.810 --> 00:17:48.790 because it's completely...

430 00:17:48.790 --> 00:17:51.780 Zero is basically inconsistent with the data

431 00:17:51.780 --> 00:17:54.300 and so the posterior looks approximately normal.

432 00:17:54.300 --> 00:17:55.780 It's actually a mixture of normals

433 00:17:55.780 --> 00:17:58.220 but it looks approximately normal 'cause of weight

434 00:17:58.220 --> 00:18:00.433 and there, zero is very, very small.

435 00:18:02.390 --> 00:18:04.370 Whereas if a...

436 00:18:04.370 --> 00:18:05.360 Here's a different example,

437 00:18:05.360 --> 00:18:07.860 the black line is covered

438 00:18:07.860 --> 00:18:09.300 by the green line this time because it's...

439 00:18:09.300 --> 00:18:11.610 So I plotted all three lines on the same plot here.

440 00:18:11.610 --> 00:18:12.560 The black line is...

441 00:18:12.560 --> 00:18:14.350 Think of it as pretty much the green line.

442 00:18:14.350 --> 00:18:16.050 It's still the same spiky prior

443 00:18:16.050 --> 00:18:18.270 but now the likelihood is much flatter.

444 00:18:18.270 --> 00:18:19.780 The XJ is the same.

445 00:18:19.780 --> 00:18:20.760 Actually, it's one and a half

446 00:18:20.760 --> 00:18:23.030 but we have an SJ that's much bigger.

447 00:18:23.030 --> 00:18:25.210 So what happens here is that

448 00:18:25.210 --> 00:18:27.170 the prior dominates because the likelihood's

449 00:18:27.170 --> 00:18:28.990 relatively flat,

450 00:18:28.990 --> 00:18:31.430 and so the posterior looks pretty much like the prior

451 00:18:31.430 --> 00:18:33.010 and you get very strong shrinkage.

452 00:18:33.010 --> 00:18:35.460 So think of this as corresponding

453 00:18:35.460 --> 00:18:37.960 to those individuals who had very few at-bats,

454 00:18:37.960 --> 00:18:40.580 their data are very imprecise,

455 00:18:40.580 --> 00:18:42.850 and so their posterior, the green line,

456 00:18:42.850 --> 00:18:45.800 looks very like the prior, the black line.

457 00:18:45.800 --> 00:18:49.770 Okay, so we're gonna shrink those observations more.

458 00:18:49.770 --> 00:18:51.900 So the key point here, I guess,

459 00:18:51.900 --> 00:18:54.870 is that the observations with larger standard error,

460 00:18:54.870 --> 00:18:56.133 larger SJ,

461 00:18:57.837 --> 00:19:00.047 get shrunk more.

462 00:19:00.047 --> 00:19:03.986 I should say "larger standard deviation" get shrunk more.

463 00:19:03.986 --> 00:19:06.730 Here's another intermediate example

464 00:19:06.730 --> 00:19:07.750 where the red line...

465 00:19:07.750 --> 00:19:10.930 The likelihood's kind of not quite enough.

466 00:19:10.930 --> 00:19:14.920 It illustrates the idea that the posterior could be bimodal

467 00:19:14.920 --> 00:19:18.240 because the prior and the likelihood are indifferent,

468 00:19:18.240 --> 00:19:19.970 have weight in different places.

469 00:19:19.970 --> 00:19:23.010 So you can get different kinds of shrinkage depending on

470 00:19:23.010 --> 00:19:24.170 how spiky the prior is,

471 00:19:24.170 --> 00:19:25.290 how long-tailed the prior is,

472 00:19:25.290 --> 00:19:27.163 how flat the likelihood is etc.

473 00:19:34.620 --> 00:19:35.830 So obviously the shrinkage,

474 00:19:35.830 --> 00:19:37.120 the amount of shrinkage you get,

475 00:19:37.120 --> 00:19:39.060 depends on the prior, G,

476 00:19:39.060 --> 00:19:40.810 which you're gonna estimate from the data.

477 00:19:40.810 --> 00:19:43.030 It also depends on the standard error

478 00:19:43.030 --> 00:19:45.720 or the standard deviation, SJ.

479 00:19:45.720 --> 00:19:48.330 And one way to summarize this kind of the behavior,

480 00:19:48.330 --> 00:19:49.810 the shrinkage behavior,

481 00:19:49.810 --> 00:19:54.810 is to focus on how the posterior mean changes with X.

482 00:19:54.840 --> 00:19:56.720 So we can define this operator here,

483 00:19:56.720 --> 00:19:59.300 S-G-S of X,

484 00:19:59.300 --> 00:20:03.770 as the X posterior mean of theta J,

485 00:20:03.770 --> 00:20:08.250 given the prior and its variance or standard deviation

486 00:20:08.250 --> 00:20:13.250 and that we observed XJ is equal to X.

487 00:20:14.441 --> 00:20:16.720 I'm gonna call this the shrinkage operator

488 00:20:16.720 --> 00:20:18.460 for the prior, G,

489 00:20:18.460 --> 00:20:21.620 and variance, S for standard deviation, S.

490 00:20:21.620 --> 00:20:23.910 Okay, so we could just plot

491 00:20:23.910 --> 00:20:25.300 some of these shrinkage operators.

492 00:20:25.300 --> 00:20:26.480 So the idea here is...

493 00:20:26.480 --> 00:20:30.170 Sorry, this slide has B instead of X.

494 00:20:30.170 --> 00:20:33.610 Sometimes I use B and sometimes I use X.

495 00:20:33.610 --> 00:20:35.110 I've got them mixed up here, sorry.

496 00:20:35.110 --> 00:20:37.400 So think of this as X

497 00:20:37.400 --> 00:20:39.070 and this is S of X.

498 00:20:39.070 --> 00:20:42.830 So these different lines here correspond

499 00:20:42.830 --> 00:20:44.320 to different priors.

500 00:20:44.320 --> 00:20:47.360 So the idea is that by using different priors,

501 00:20:47.360 --> 00:20:50.350 we can get different types of shrinkage behavior.

502 00:20:50.350 --> 00:20:53.590 So this prior here shrinks very strongly to zero.

503 00:20:53.590 --> 00:20:57.000 This green line shrinks very strongly to zero

504 00:20:57.000 --> 00:21:00.900 until B exceeds some value around five,

505 00:21:00.900 --> 00:21:02.900 at which point it hardly shrinks at all.

506 00:21:04.010 --> 00:21:06.580 So this is kind of a prior that has

507 00:21:06.580 --> 00:21:08.783 kind of a big spike near zero.

508 00:21:09.790 --> 00:21:11.333 But also a long tail,

509 00:21:11.333 --> 00:21:14.880 such that when you get far enough in the tail,

510 00:21:14.880 --> 00:21:16.780 you start to be convinced

511 00:21:16.780 --> 00:21:18.100 that there's a real signal here.

512 00:21:18.100 --> 00:21:18.990 So you can think of that

513 00:21:18.990 --> 00:21:20.010 as this kind of...

514 00:21:20.010 --> 00:21:22.105 This is sometimes called...

515 00:21:22.105 --> 00:21:24.210 This is local shrinkage

516 00:21:24.210 --> 00:21:25.509 and this is global.

517 00:21:25.509 --> 00:21:28.690 So you get very strong local shrinkage towards zero

518 00:21:28.690 --> 00:21:30.860 but very little shrinkage

519 00:21:30.860 --> 00:21:32.600 if the signal is strong enough.

520 00:21:32.600 --> 00:21:34.090 That kind of thing.

521 00:21:34.090 --> 00:21:35.180 But the real point here is that

522 00:21:35.180 --> 00:21:36.980 by using different priors,

523 00:21:36.980 --> 00:21:39.440 these different scale mixture of normal priors,

524 00:21:39.440 --> 00:21:43.460 you can get very different looking shrinkage behaviors.

525 00:21:43.460 --> 00:21:45.390 Ones that shrink very strongly to zero

526 00:21:45.390 --> 00:21:46.420 and then stop shrinking

527 00:21:46.420 --> 00:21:50.313 or ones that shrink a little bit all the way, etc.

528 00:21:51.910 --> 00:21:54.610 And so, if you're familiar with other ways

529 00:21:54.610 --> 00:21:56.270 of doing shrinkage analysis,

530 00:21:56.270 --> 00:21:57.800 and this is one of them,

531 00:21:57.800 --> 00:21:59.010 or shrinkage,

532 00:21:59.010 --> 00:22:00.960 is to use a penalized likelihood.

533 00:22:00.960 --> 00:22:03.720 Then you can try and draw a parallel

534 00:22:03.720 --> 00:22:04.760 and that's what I'm trying to do here.

535 00:22:04.760 --> 00:22:07.500 Draw a parallel between the Bayesean method

536 00:22:07.500 --> 00:22:10.800 and the penalized likelihood-based approaches

537 00:22:10.800 --> 00:22:12.943 to inducing shrinkage or sparsity.

538 00:22:14.690 --> 00:22:18.270 Another way to induce shrinkage is to essentially...

539 00:22:18.270 --> 00:22:20.880 This is the kind of normal log likelihood here

540 00:22:20.880 --> 00:22:23.470 and this is a penalty here

541 00:22:23.470 --> 00:22:24.960 that you add for this.

542 00:22:24.960 --> 00:22:26.120 This could be an L1 penalty

543 00:22:26.120 --> 00:22:28.370 or an L2 penalty or an L0 penalty,

544 00:22:28.370 --> 00:22:30.170 or some other kind of penalty.

545 00:22:30.170 --> 00:22:32.172 So there's a penalty function here.

546 00:22:32.172 --> 00:22:34.240 And you define the estimate

547 00:22:34.240 --> 00:22:35.810 as the value that minimizes

548 00:22:35.810 --> 00:22:37.851 this penalized log likelihood.

549 00:22:37.851 --> 00:22:40.520 Sorry, yeah, this is a negative log likelihood.

550 00:22:40.520 --> 00:22:42.870 Penalized least squares, I guess this would be.

551 00:22:44.590 --> 00:22:48.978 Okay, so now eight is a penalty function here

552 00:22:48.978 --> 00:22:51.140 and Lambda is a tuning parameter

553 00:22:51.140 --> 00:22:55.266 that says how strong, in some sense, the penalty is.

554 00:22:55.266 --> 00:22:57.790 And these are also widely used

555 00:22:57.790 --> 00:22:58.900 to induce shrinkage,

556 00:22:58.900 --> 00:23:02.111 especially in regression contexts.

557 00:23:02.111 --> 00:23:06.450 And so, here are some commonly used shrinkage operators,

558 00:23:06.450 --> 00:23:09.020 corresponding to different penalty functions.

559 00:23:09.020 --> 00:23:12.210 So this green line is what's called

560 00:23:12.210 --> 00:23:16.240 the hard thresholding,

561 00:23:16.240 --> 00:23:19.029 which corresponds to an L0 penalty.

562 00:23:19.029 --> 00:23:21.060 If you don't know what that means, don't worry.

563 00:23:21.060 --> 00:23:23.623 But if you do, you make that connection.

564 00:23:24.830 --> 00:23:27.550 At the red line here is L1 penalty

565 00:23:27.550 --> 00:23:29.310 or soft thresholding.

566 00:23:29.310 --> 00:23:33.060 And these two other ones here are particular instances

567 00:23:33.060 --> 00:23:35.670 of some non-convex penalties that are used

568 00:23:35.670 --> 00:23:36.740 in regression context,

569 00:23:36.740 --> 00:23:38.740 particularly in practice.

570 00:23:38.740 --> 00:23:42.230 And I guess that the point here is

571 00:23:42.230 --> 00:23:45.170 that, essentially, different prior distributions

572 00:23:45.170 --> 00:23:47.730 in the normal means model can lead

573 00:23:47.730 --> 00:23:51.070 to shrinkage operators, shrinkage behavior

574 00:23:51.070 --> 00:23:52.856 that looks kind of similar

575 00:23:52.856 --> 00:23:57.856 to each of these different types of penalty.

576 00:23:58.310 --> 00:24:02.636 So you can't actually mimic the behavior exactly.

577 00:24:02.636 --> 00:24:04.580 I've just...

578 00:24:04.580 --> 00:24:06.850 Or actually, my student, (indistinct) Kim,

579 00:24:06.850 --> 00:24:10.690 chose the priors to visually closely match these

580 00:24:10.690 --> 00:24:11.680 but you can't get...

581 00:24:11.680 --> 00:24:13.110 Some of these have kinks and stuff

582 00:24:13.110 --> 00:24:17.710 that you can't actually, formally, exactly mimic

583 00:24:17.710 --> 00:24:21.020 but you can get qualitatively similar shrinkage behavior

584 00:24:21.020 --> 00:24:22.680 from different priors

585 00:24:22.680 --> 00:24:24.210 as different penalty functions.

586 00:24:24.210 --> 00:24:25.870 So you should think about the different priors

587 00:24:25.870 --> 00:24:29.143 as being analogous to different penalty functions.

588 00:24:30.280 --> 00:24:32.990 And so, the key...

589 00:24:32.990 --> 00:24:35.480 How does EB, empirical Bayse shrinkage,

590 00:24:35.480 --> 00:24:40.050 differ from, say, these kinds of penalty-based approaches,

591 00:24:40.050 --> 00:24:41.290 which I should say are maybe

592 00:24:41.290 --> 00:24:44.303 more widely used in practice?

593 00:24:44.303 --> 00:24:49.176 Well, so shrinkage is determined by the prior, G,

594 00:24:49.176 --> 00:24:51.980 which we estimate in an empirical Bayse context

595 00:24:51.980 --> 00:24:53.390 by maximum likelihood.

596 00:24:53.390 --> 00:24:56.609 Whereas in typical shrinkage...

597 00:24:56.609 --> 00:24:59.930 Sorry, typical penalty-based analyses,

598 00:24:59.930 --> 00:25:03.140 people use cross validation to estimate parameters.

599 00:25:03.140 --> 00:25:06.790 And the result is that cross-validation is fine

600 00:25:06.790 --> 00:25:08.010 for estimating one parameter

601 00:25:08.010 --> 00:25:09.740 but it becomes quite cumbersome

602 00:25:09.740 --> 00:25:11.710 to estimate two parameters,

603 00:25:11.710 --> 00:25:14.380 and really tricky to estimate three or four parameters

604 00:25:14.380 --> 00:25:17.100 'cause you have to go and do a grid of different values

605 00:25:17.100 --> 00:25:17.933 and do a lot of cross-validations

606 00:25:17.933 --> 00:25:21.590 and start estimating all these different parameters.

607 00:25:21.590 --> 00:25:22.910 So the point here is really

608 00:25:22.910 --> 00:25:25.570 because we estimate G by maximum likelihood,

609 00:25:25.570 --> 00:25:29.460 we can actually have a much more flexible family in practice

610 00:25:29.460 --> 00:25:32.647 that we can optimize over more easily.

611 00:25:32.647 --> 00:25:33.900 It's very flexible,

612 00:25:33.900 --> 00:25:35.670 you can mimic a range of penalty functions

613 00:25:35.670 --> 00:25:36.990 so you don't have to choose

614 00:25:36.990 --> 00:25:39.850 whether to use L1 or L2 or L0.

615 00:25:39.850 --> 00:25:41.910 You can essentially estimate

616 00:25:41.910 --> 00:25:44.059 over these non-parametric prior families.

617 00:25:44.059 --> 00:25:46.870 Think of that as kind of deciding automatically

618 00:25:46.870 --> 00:25:48.930 whether to use L0, L1, L2

619 00:25:48.930 --> 00:25:51.270 or some kind of non-convex penalty,

620 00:25:51.270 --> 00:25:53.220 or something in between.

621 00:25:53.220 --> 00:25:57.650 And the posterior distribution, of course then,

622 00:25:57.650 --> 00:25:59.340 another nice thing is that it gives not

623 00:25:59.340 --> 00:26:00.630 only the point estimates

624 00:26:00.630 --> 00:26:04.400 but, if you like, it also gives shrunken interval estimates

625 00:26:04.400 --> 00:26:07.770 which are not yielded by a penalty-based approach.

626 00:26:07.770 --> 00:26:09.430 So I guess I'm trying to say

627 00:26:09.430 --> 00:26:11.120 that there are potential advantages

628 00:26:11.120 --> 00:26:12.600 of the empirical Bayse approach

629 00:26:12.600 --> 00:26:15.750 over the penalty-based approach.

630 00:26:15.750 --> 00:26:18.600 And yeah, although I think,

631 00:26:18.600 --> 00:26:21.330 people have tried, particularly Efron has highlighted

632 00:26:21.330 --> 00:26:22.840 the potential for empirical Bayse

633 00:26:22.840 --> 00:26:24.614 to be used in practical applications,

634 00:26:24.614 --> 00:26:26.790 largely in the practical application.

635 00:26:26.790 --> 00:26:29.060 So I've seen empirical Bayse shrinkage hasn't

636 00:26:29.060 --> 00:26:31.293 really been used very, very much.

637 00:26:32.460 --> 00:26:34.240 So that's the goal,

638 00:26:34.240 --> 00:26:35.653 is to change that.

639 00:26:36.550 --> 00:26:39.130 So before I talk about examples,

640 00:26:39.130 --> 00:26:41.340 I guess I will pause for a moment

641 00:26:41.340 --> 00:26:43.040 to see if there are any questions.

642 00:26:53.740 --> 00:26:55.550 And I can't see the chat for some reason

643 00:26:55.550 --> 00:26:56.900 so if anyone...

644 00:26:56.900 --> 00:26:58.230 So please unmute yourself

645 00:26:58.230 --> 00:27:00.130 if you have a question.

646 00:27:00.130 --> 00:27:03.976 - I don't think people are (indistinct)

647 00:27:03.976 --> 00:27:04.809 every question in the chat.

648 00:27:04.809 --> 00:27:07.330 At least, I didn't see any. - Good.

649 00:27:07.330 --> 00:27:08.163 Okay, thank you.

650 00:27:10.160 --> 00:27:10.993 It's all clear.

651 00:27:12.028 --> 00:27:13.775 I'm happy to go on but

652 00:27:13.775 --> 00:27:14.880 I just wanna...

653 00:27:24.570 --> 00:27:26.830 Okay, so we've been trying to...

654 00:27:26.830 --> 00:27:28.300 My group has been trying to think about

655 00:27:28.300 --> 00:27:29.830 how to use these ideas,

656 00:27:29.830 --> 00:27:31.900 make these ideas useful in practice

657 00:27:31.900 --> 00:27:34.601 for a range of practical applications.

658 00:27:34.601 --> 00:27:37.487 We've done work on multiple testing,

659 00:27:37.487 --> 00:27:39.930 on high dimensional linear aggression,

660 00:27:39.930 --> 00:27:42.510 and also some on matrix factorization.

661 00:27:42.510 --> 00:27:43.500 From previous experience,

662 00:27:43.500 --> 00:27:45.360 I'll probably get time to talk about the first two

663 00:27:45.360 --> 00:27:47.190 and maybe not the last one,

664 00:27:47.190 --> 00:27:49.140 but there's a pre-print on the archive.

665 00:27:49.140 --> 00:27:50.270 You can see if you're interested

666 00:27:50.270 --> 00:27:51.230 in matrix factorization.

667 00:27:51.230 --> 00:27:55.338 Maybe I'll get to get to talk about that briefly.

668 00:27:55.338 --> 00:27:58.620 But let me talk about multiple testing first.

669 00:27:58.620 --> 00:28:02.017 So the typical multiple testing setup,

670 00:28:02.017 --> 00:28:04.770 where you might typically, say,

671 00:28:04.770 --> 00:28:07.561 apply a Benjamini-Hochberg type procedure is

672 00:28:07.561 --> 00:28:09.080 you've got a large number of tests,

673 00:28:09.080 --> 00:28:11.100 So J equals one to N,

674 00:28:11.100 --> 00:28:14.660 and test J yields a P value, PJ,

675 00:28:14.660 --> 00:28:16.410 and then you reject all tests with

676 00:28:16.410 --> 00:28:19.410 some PJ less than a threshold gamma,

677 00:28:19.410 --> 00:28:20.810 where that threshold is chosen

678 00:28:20.810 --> 00:28:23.320 to control the FDR in a frequented sense.

679 00:28:23.320 --> 00:28:25.650 So that's the typical setup.

680 00:28:25.650 --> 00:28:27.163 So how are we going to apply

681 00:28:27.163 --> 00:28:30.293 the normal means model to this problem?

682 00:28:32.874 --> 00:28:36.520 Okay, well, in many applications,

683 00:28:36.520 --> 00:28:38.010 not all but in many,

684 00:28:38.010 --> 00:28:39.590 the P values are derived from

685 00:28:39.590 --> 00:28:41.450 some kind of effect size estimate,

686 00:28:41.450 --> 00:28:44.540 which I'm going to call "Beta hat J,"

687 00:28:44.540 --> 00:28:46.710 which have standard errors, SJ,

688 00:28:46.710 --> 00:28:48.960 that satisfy approximately, at least,

689 00:28:48.960 --> 00:28:52.800 that Beta J hat is normally distributed

690 00:28:52.800 --> 00:28:55.140 about the true value Beta J

691 00:28:55.140 --> 00:28:58.250 with some variance given it by SJ.

692 00:28:58.250 --> 00:29:01.260 So in a lot...

693 00:29:01.260 --> 00:29:03.270 I work a lot in genetic applications.

694 00:29:03.270 --> 00:29:04.910 So in genetic applications,

695 00:29:04.910 --> 00:29:06.820 we're looking at different genes here.

696 00:29:06.820 --> 00:29:10.520 So Beta J hat might be the estimate

697 00:29:10.520 --> 00:29:13.291 of the difference in expression, let's say,

698 00:29:13.291 --> 00:29:17.613 of a gene, J, between, say, males and females.

699 00:29:17.613 --> 00:29:21.090 And Beta J would be the true difference

700 00:29:21.090 --> 00:29:22.400 at that gene.

701 00:29:22.400 --> 00:29:25.180 And you're interested in identifying

702 00:29:25.180 --> 00:29:28.370 which genes are truly different...

703 00:29:28.370 --> 00:29:30.140 Have a different mean expression

704 00:29:30.140 --> 00:29:32.310 between males and females here.

705 00:29:32.310 --> 00:29:36.250 And the reason that SJ is approximately known is

706 00:29:36.250 --> 00:29:37.990 because you've got multiple males

707 00:29:37.990 --> 00:29:40.920 and multiple females that you're using

708 00:29:40.920 --> 00:29:43.310 to estimate this difference.

709 00:29:43.310 --> 00:29:44.810 And so, you get an estimated standard error

710 00:29:44.810 --> 00:29:46.623 of that Beta hat as well.

711 00:29:48.360 --> 00:29:50.080 And so, once you've set the problem up like this,

712 00:29:50.080 --> 00:29:54.820 of course, it looks suddenly like a normal means problem

713 00:29:54.820 --> 00:29:56.727 and we can kind of apply

714 00:29:56.727 --> 00:30:00.177 the empirical Bayes normal means idea.

715 00:30:00.177 --> 00:30:03.200 We're gonna put a prior on the Beta Js

716 00:30:03.200 --> 00:30:04.770 that is sparsity inducing.

717 00:30:04.770 --> 00:30:06.340 That is, it's kind of centered at zero,

718 00:30:06.340 --> 00:30:08.343 maybe it's got a point mass at zero.

719 00:30:08.343 --> 00:30:11.793 But we're gonna estimate that prior from the data.

720 00:30:14.450 --> 00:30:15.283 Okay.

721 00:30:16.520 --> 00:30:21.520 And so, not only can you get posterior means for Beta,

722 00:30:23.330 --> 00:30:25.460 as I said, you can get posterior interval estimates.

723 00:30:25.460 --> 00:30:27.410 So you can kind of do things

724 00:30:27.410 --> 00:30:31.276 like compute the posterior in 90% credible interval,

725 00:30:31.276 --> 00:30:33.020 given that prior and the likelihood

726 00:30:33.020 --> 00:30:34.120 for each Beta J

727 00:30:34.120 --> 00:30:35.363 and we could reject, for example,

728 00:30:35.363 --> 00:30:38.600 if the interval does not contain zero.

729 00:30:38.600 --> 00:30:42.140 And I'm not going to talk about this in detail

730 00:30:42.140 --> 00:30:43.720 because the details are in

731 00:30:43.720 --> 00:30:46.285 a biostatistics paper from 2017.

732 00:30:46.285 --> 00:30:50.377 I should say that the idea of using empirical Bayse for FDR

733 00:30:50.377 --> 00:30:53.663 actually dates back to before Benjamini and Hoffberg.

734 00:30:53.663 --> 00:30:57.050 Duncan Thomas has a really nice paper

735 00:30:57.050 --> 00:30:58.631 that was pointed out to me by John Witty

736 00:30:58.631 --> 00:31:01.740 that actually contains these basic ideas

737 00:31:02.850 --> 00:31:06.880 but not nice software implementation,

738 00:31:06.880 --> 00:31:08.990 which maybe explains why it hasn't caught on

739 00:31:08.990 --> 00:31:10.330 in practice yet.

740 00:31:10.330 --> 00:31:13.318 Efron's also been a pioneer in this area.

741 00:31:13.318 --> 00:31:15.527 So...

742 00:31:15.527 --> 00:31:18.430 Okay, so I don't want to dwell on that

743 00:31:18.430 --> 00:31:21.380 because, actually, I think I'll just summarize

744 00:31:21.380 --> 00:31:24.851 what I think is true compared with Benjamini-Hochberg.

745 00:31:24.851 --> 00:31:26.950 You get a bit of an increase in power

746 00:31:26.950 --> 00:31:28.731 by using an empirical Bayse approach.

747 00:31:28.731 --> 00:31:31.517 The Benjamini-Hochberg approach is

748 00:31:31.517 --> 00:31:34.050 more robust to correlated tests though,

749 00:31:34.050 --> 00:31:37.640 so the empirical Bayse normal means model does assume

750 00:31:37.640 --> 00:31:38.877 that the tests are independent

751 00:31:38.877 --> 00:31:42.900 and, in practice, we have seen

752 00:31:42.900 --> 00:31:44.540 that correlations can cause problems.

753 00:31:44.540 --> 00:31:45.550 If you're interested in that,

754 00:31:45.550 --> 00:31:48.812 I have a pre-print with Lei Sun on my website.

755 00:31:48.812 --> 00:31:51.945 But the empirical Bayse normal means

756 00:31:51.945 --> 00:31:54.590 also provides these interval estimates,

757 00:31:54.590 --> 00:31:55.990 which is kind of nice.

758 00:31:55.990 --> 00:31:57.524 Benjamini-Hochberg does not.

759 00:31:57.524 --> 00:31:59.354 So there are some advantages

760 00:31:59.354 --> 00:32:01.240 of the empirical Bayes approach

761 00:32:01.240 --> 00:32:03.820 and maybe some disadvantages compared

762 00:32:03.820 --> 00:32:04.653 with Benjamini-Hochberg.

763 00:32:04.653 --> 00:32:06.218 But I think that the real benefit

764 00:32:06.218 --> 00:32:07.990 of the empirical Bayse approach

765 00:32:07.990 --> 00:32:10.830 actually comes when we look at multi-variate extensions

766 00:32:10.830 --> 00:32:11.900 of this idea.

767 00:32:11.900 --> 00:32:14.455 So I just wanted to briefly highlight those

768 00:32:14.455 --> 00:32:16.466 and spend some time on those.

769 00:32:16.466 --> 00:32:19.400 So here's the multi-variate version

770 00:32:19.400 --> 00:32:23.630 of the empirical Bayse normal means models.

771 00:32:23.630 --> 00:32:28.630 And now, my Beta Js are a vector of observation.

772 00:32:28.770 --> 00:32:31.880 So I think of this as measuring, say,

773 00:32:31.880 --> 00:32:34.480 gene J in multiple different tissues.

774 00:32:34.480 --> 00:32:35.620 Think of different tissues.

775 00:32:35.620 --> 00:32:37.023 You look at at heart

776 00:32:37.023 --> 00:32:38.096 you look at lung,

777 00:32:38.096 --> 00:32:39.444 you look brain,

778 00:32:39.444 --> 00:32:42.382 you look at the spleen.

779 00:32:42.382 --> 00:32:45.730 In fact, we've got 50 different tissues in the example

780 00:32:45.730 --> 00:32:47.614 I'm gonna show in a minute.

781 00:32:47.614 --> 00:32:52.286 So we've measured some kind of effect

782 00:32:52.286 --> 00:32:56.180 in each gene, in each of these 50 different tissues

783 00:32:56.180 --> 00:33:01.050 and we want to know where the effects are...

784 00:33:01.050 --> 00:33:03.820 Which genes show effects in which tissues.

785 00:33:03.820 --> 00:33:07.620 So Beta J is now a vector of length R,

786 00:33:07.620 --> 00:33:08.780 the number of tissues.

787 00:33:08.780 --> 00:33:10.912 R is 50 in our example.

788 00:33:10.912 --> 00:33:12.134 And so you've got...

789 00:33:12.134 --> 00:33:15.800 We're gonna assume that the estimates are

790 00:33:15.800 --> 00:33:17.170 normally distributed with mean,

791 00:33:17.170 --> 00:33:19.128 the true values and some variance,

792 00:33:19.128 --> 00:33:20.890 covariance matrix now,

793 00:33:20.890 --> 00:33:22.910 which we're going to assume, for now, is known.

794 00:33:22.910 --> 00:33:25.720 That's actually a little trickier

795 00:33:25.720 --> 00:33:27.900 but I'm gonna gloss over that for...

796 00:33:27.900 --> 00:33:28.982 If you want to see details,

797 00:33:28.982 --> 00:33:30.680 take a look at the paper.

798 00:33:30.680 --> 00:33:33.008 I just wanna get the essence of the idea across.

799 00:33:33.008 --> 00:33:35.410 We're still going to assume that Beta J comes

800 00:33:35.410 --> 00:33:36.410 from some prior, G,

801 00:33:36.410 --> 00:33:38.380 and we're still gonna use a mixture of normals,

802 00:33:38.380 --> 00:33:39.410 but now we're using a mixture

803 00:33:39.410 --> 00:33:40.880 of multi-variate normals.

804 00:33:40.880 --> 00:33:42.658 And unlike the univariate case,

805 00:33:42.658 --> 00:33:44.680 we can't use a grid of...

806 00:33:44.680 --> 00:33:47.160 We can't use a grid of values

807 00:33:47.160 --> 00:33:50.419 that span all possible covariance matrices here.

808 00:33:50.419 --> 00:33:51.730 It's just too much.

809 00:33:51.730 --> 00:33:53.249 So we have to do something to estimate

810 00:33:53.249 --> 00:33:54.850 these covariance matrices,

811 00:33:54.850 --> 00:33:57.010 as well as estimate the pis.

812 00:33:57.010 --> 00:33:59.520 And again, if you want to see the details,

813 00:33:59.520 --> 00:34:01.453 take a look at a Urbut et al.

814 00:34:02.940 --> 00:34:04.341 But let me just illustrate

815 00:34:04.341 --> 00:34:06.039 the idea of what's going on here,

816 00:34:06.039 --> 00:34:08.490 or what happens when you apply this method

817 00:34:08.490 --> 00:34:09.630 to some data.

818 00:34:09.630 --> 00:34:10.463 So this is...

819 00:34:10.463 --> 00:34:11.410 I said 50,

820 00:34:11.410 --> 00:34:14.420 we have 44 tissues in this particular example.

821 00:34:14.420 --> 00:34:17.114 So each row here is a tissue.

822 00:34:17.114 --> 00:34:20.102 These yellow ones here are brain tissues,

823 00:34:20.102 --> 00:34:21.620 different brain tissues,

824 00:34:21.620 --> 00:34:24.583 and I think we'll see one later that's blood.

825 00:34:24.583 --> 00:34:26.410 I think this one might be blood.

826 00:34:26.410 --> 00:34:27.610 Anyway, each one is a tissue;

827 00:34:27.610 --> 00:34:29.220 lung, blood, etc.

828 00:34:29.220 --> 00:34:31.554 You don't need to know which ones are which, for now.

829 00:34:31.554 --> 00:34:34.210 And so, what we've done here is plot

830 00:34:34.210 --> 00:34:37.794 the Beta hat and plus or minus two standard deviations

831 00:34:37.794 --> 00:34:40.730 for each tissue at a particular...

832 00:34:40.730 --> 00:34:43.835 In this case, a particular snip, actually (indistinct).

833 00:34:43.835 --> 00:34:45.820 So this is an eQTL analysis,

834 00:34:45.820 --> 00:34:47.843 for those of you who know what that means.

835 00:34:47.843 --> 00:34:49.850 If you don't, don't worry about it.

836 00:34:49.850 --> 00:34:52.940 Just think of it as having an estimated effect

837 00:34:52.940 --> 00:34:54.659 plus or minus two standard deviations

838 00:34:54.659 --> 00:34:57.774 in 44 different tissues,

839 00:34:57.774 --> 00:35:01.247 and we want to know which ones are quote,

840 00:35:01.247 --> 00:35:03.307 ”significantly different from zero.”

841 00:35:05.350 --> 00:35:07.680 And so what happens...

842 00:35:09.936 --> 00:35:11.120 Sorry.

843 00:35:11.120 --> 00:35:12.570 Didn’t expect that to happen.

844 00:35:16.660 --> 00:35:17.493 Sorry, okay.

845 00:35:17.493 --> 00:35:18.341 Yeah, these are just...

846 00:35:18.341 --> 00:35:20.010 These are just two examples.

847 00:35:20.010 --> 00:35:21.620 So this is one example,

848 00:35:21.620 --> 00:35:22.710 here’s another example

849 00:35:22.710 --> 00:35:24.310 where we’ve done the same thing.

850 00:35:25.330 --> 00:35:28.010 Estimated effects, plus or minus two standard deviations.

851 00:35:28.010 --> 00:35:29.683 So what you can see in this first one is

852 00:35:29.683 --> 00:35:30.900 that it looks like, at least,

853 00:35:30.900 --> 00:35:33.430 that the brain tissues have some kind of effect.

854 00:35:33.430 --> 00:35:35.970 That’s what you’re supposed to see here.

855 00:35:35.970 --> 00:35:37.733 And maybe there are some effects in other tissues.

856 00:35:37.733 --> 00:35:40.240 There’s a tendency for effects to be positive,

857 00:35:40.240 --> 00:35:43.640 which might suggest that maybe everything has

858 00:35:43.640 --> 00:35:44.950 a small effect to everywhere,

859 00:35:44.950 --> 00:35:47.180 but particularly strong in the brain.

860 00:35:47.180 --> 00:35:50.110 And whereas in this example,

861 00:35:50.110 --> 00:35:52.020 this one appears to have an effect

862 00:35:52.020 --> 00:35:53.350 in just one tissue.

863 00:35:53.350 --> 00:35:54.320 This is the blood actually.

864 00:35:54.320 --> 00:35:56.380 So this is an effect in blood

865 00:35:56.380 --> 00:35:58.130 but mostly, it doesn’t look like

866 00:35:58.130 --> 00:36:00.280 there’s an effect in other tissues.

867 00:36:00.280 --> 00:36:01.480 But these, just to emphasize,

868 00:36:01.480 --> 00:36:02.610 these are the raw data,

869 00:36:02.610 --> 00:36:04.250 in the sense that they're the Beta hats

870 00:36:04.250 --> 00:36:05.083 and the standard errors.

871 00:36:05.083 --> 00:36:07.499 There's no shrinkage occurred yet.

872 00:36:07.499 --> 00:36:10.650 But the idea is that the empirical Bayse approach takes

873 00:36:10.650 --> 00:36:11.630 all these examples,

874 00:36:11.630 --> 00:36:13.553 examples like this and examples like this,

875 00:36:13.553 --> 00:36:17.040 to learn about what kinds of patterns are present

876 00:36:17.040 --> 00:36:17.873 in the data.

877 00:36:17.873 --> 00:36:19.036 That is, "What does G look like?"

878 00:36:19.036 --> 00:36:21.161 So it learns from these examples

879 00:36:21.161 --> 00:36:24.440 that there are some effects that look like

880 00:36:24.440 --> 00:36:26.500 they're shared among the brain tissues,

881 00:36:26.500 --> 00:36:28.371 and there are some effects that are...

882 00:36:28.371 --> 00:36:31.020 These are actually somewhat rare

883 00:36:31.020 --> 00:36:33.340 but rarely, there's an effect that's specific

884 00:36:33.340 --> 00:36:35.314 to one tissue like, in this case, blood.

885 00:36:35.314 --> 00:36:38.866 And it also learns, in this case actually,

886 00:36:38.866 --> 00:36:40.773 that there's a lot of null things,

887 00:36:40.773 --> 00:36:44.020 because there are a lot of null things as well.

888 00:36:44.020 --> 00:36:46.220 So it puts lots of mass on the null as well

889 00:36:46.220 --> 00:36:47.772 and that causes the shrinkage.

890 00:36:47.772 --> 00:36:51.142 And then, having estimated those patterns from the data,

891 00:36:51.142 --> 00:36:52.689 it computes posteriors.

892 00:36:52.689 --> 00:36:57.689 And so, here's the data and then the posterior intervals

893 00:36:57.820 --> 00:36:58.690 for the same...

894 00:36:58.690 --> 00:37:00.330 For that first example.

895 00:37:00.330 --> 00:37:02.012 And what you can see is that because of

896 00:37:02.012 --> 00:37:05.825 the combining information across tissues,

897 00:37:05.825 --> 00:37:08.879 you get standard errors that are getting smaller,

898 00:37:08.879 --> 00:37:12.214 the brain estimates all get shrunk towards one another,

899 00:37:12.214 --> 00:37:13.565 and all these...

900 00:37:13.565 --> 00:37:16.450 There's some borrowing strength of information,

901 00:37:16.450 --> 00:37:18.070 borrowing information across these tissues,

902 00:37:18.070 --> 00:37:19.903 to make these these look like

903 00:37:19.903 --> 00:37:21.850 some of them are kind of borderline significant.

904 00:37:21.850 --> 00:37:22.930 Now, it looks like there's probably

905 00:37:22.930 --> 00:37:24.530 an effect in every tissue

906 00:37:24.530 --> 00:37:26.483 but a much stronger effect in brain.

907 00:37:26.483 --> 00:37:28.696 Whereas this example here,

908 00:37:28.696 --> 00:37:31.700 it recognizes that this looks like an effect

909 00:37:31.700 --> 00:37:33.430 that's specific to blood.

910 00:37:33.430 --> 00:37:35.910 And so, it shrinks everything else strongly towards zero

911 00:37:35.910 --> 00:37:37.571 because it knows that most things are null,

912 00:37:37.571 --> 00:37:39.520 it's learned that from the data,

913 00:37:39.520 --> 00:37:42.980 but the blood estimate gets hardly shrunk at all.

914 00:37:42.980 --> 00:37:44.313 We saw that kind of behavior where things

915 00:37:44.313 --> 00:37:46.500 that are near zero can get shrunk towards zero,

916 00:37:46.500 --> 00:37:48.030 whereas other things that are far

917 00:37:48.030 --> 00:37:49.649 away don't get shrunk as much.

918 00:37:49.649 --> 00:37:52.835 And it's really hard to do that kind of thing

919 00:37:52.835 --> 00:37:56.750 without doing some kind of model-based analysis,

920 00:37:56.750 --> 00:38:01.124 doing Benjamini-Hochberg type art non-model based

921 00:38:01.124 --> 00:38:03.100 without making any assumptions

922 00:38:03.100 --> 00:38:05.550 or making minimal assumptions,

923 00:38:05.550 --> 00:38:09.020 very hard to capture this kind of thing, I think.

924 00:38:09.020 --> 00:38:11.020 So I think the empirical Bayse approach

925 00:38:11.020 --> 00:38:13.703 has big advantages in this setting.

926 00:38:16.860 --> 00:38:19.500 I'll pause before I talk about regression.

927 00:38:19.500 --> 00:38:20.983 Any questions there?

928 00:38:23.650 --> 00:38:26.220 - So Matthew, I have some basic questions.

929 00:38:26.220 --> 00:38:30.060 So in your means multivariate multiple testing case,

930 00:38:30.060 --> 00:38:31.970 I guess for each of the plot,

931 00:38:31.970 --> 00:38:34.550 you are looking at maybe a particular genes influence

932 00:38:34.550 --> 00:38:36.790 on some... - Good, yeah.

933 00:38:36.790 --> 00:38:38.638 Sorry, I did skip over it a bit.

934 00:38:38.638 --> 00:38:40.080 So these are eQTLs.

935 00:38:40.080 --> 00:38:42.983 So actually, what I'm plotting here is each...

936 00:38:42.983 --> 00:38:46.420 This is a single snip associated

937 00:38:46.420 --> 00:38:47.520 with a single gene.

938 00:38:47.520 --> 00:38:49.437 And this is it's,

939 00:38:49.437 --> 00:38:51.515 "How associated is this snip

940 00:38:51.515 --> 00:38:53.975 "with this genes expression level

941 00:38:53.975 --> 00:38:56.395 "in the different brain tissues,

942 00:38:56.395 --> 00:39:00.347 "in the blood tissue in lung and spleen, etc?"

943 00:39:02.260 --> 00:39:03.910 The idea is that...

944 00:39:03.910 --> 00:39:06.750 What the scientific goal is to understand

945 00:39:06.750 --> 00:39:10.170 which genetic variants are impacting gene expression

946 00:39:10.170 --> 00:39:11.810 in different tissues,

947 00:39:11.810 --> 00:39:13.340 which might tell us something

948 00:39:13.340 --> 00:39:14.817 about the biology of the tissues

949 00:39:14.817 --> 00:39:18.120 and the regulation going on in the different tissues.

950 00:39:18.120 --> 00:39:18.953 - Got it.

951 00:39:18.953 --> 00:39:20.378 So in this case,

952 00:39:20.378 --> 00:39:23.330 I don't think I fully understand

953 00:39:23.330 --> 00:39:26.660 why it's multivariate multiple tests, not univariate

954 00:39:26.660 --> 00:39:28.400 because you are looking at each gene

955 00:39:28.400 --> 00:39:30.033 versus each snip.

956 00:39:31.890 --> 00:39:32.782 - Right, so sorry.

957 00:39:32.782 --> 00:39:35.650 Think of J indexing eQTL.

958 00:39:35.650 --> 00:39:39.070 So we've got 2 million potential eQTLs,

959 00:39:39.070 --> 00:39:41.424 so that's the multiple part of it.

960 00:39:41.424 --> 00:39:44.700 For 2 million potential eQTLs, that's...

961 00:39:44.700 --> 00:39:49.700 And then each eQTL has data on 44 tissues,

962 00:39:50.021 --> 00:39:52.499 so that's the multi-variate part of it.

963 00:39:52.499 --> 00:39:53.480 (speaking over each other)

964 00:39:53.480 --> 00:39:54.450 If you thought about it

965 00:39:54.450 --> 00:39:57.150 in terms of say P values or maybe Z scores,

966 00:39:57.150 --> 00:39:58.880 you have a matrix of Z scores.

967 00:39:58.880 --> 00:40:00.629 There are two million rows

968 00:40:00.629 --> 00:40:02.970 and there are 44 columns

969 00:40:02.970 --> 00:40:04.736 and you have a Z score or a P value

970 00:40:04.736 --> 00:40:08.510 for each element in that matrix,

971 00:40:08.510 --> 00:40:11.242 and what we're assuming is that

972 00:40:11.242 --> 00:40:12.840 the rows are independent,

973 00:40:12.840 --> 00:40:14.970 which is not quite true but still,

974 00:40:14.970 --> 00:40:16.515 we're assuming that the rows are independent

975 00:40:16.515 --> 00:40:17.792 and the columns,

976 00:40:17.792 --> 00:40:20.110 we're assuming that they can be correlated.

977 00:40:20.110 --> 00:40:20.943 And in particular,

978 00:40:20.943 --> 00:40:22.259 we're assuming that the...

979 00:40:22.259 --> 00:40:24.330 Well, we're assuming that both

980 00:40:24.330 --> 00:40:25.910 the measurements can be correlated,

981 00:40:25.910 --> 00:40:27.300 so it's V,

982 00:40:27.300 --> 00:40:29.840 but also that the effects can be correlated.

983 00:40:29.840 --> 00:40:31.360 So that's to capture the idea

984 00:40:31.360 --> 00:40:32.550 that there might be some effects

985 00:40:32.550 --> 00:40:35.770 that are shared between say brain tissues--

986 00:40:35.770 --> 00:40:36.603 - I see.

987 00:40:36.603 --> 00:40:37.930 I see.

988 00:40:37.930 --> 00:40:39.740 So this multi-variate is different

989 00:40:39.740 --> 00:40:42.990 from our usual notion where the multivariate

990 00:40:42.990 --> 00:40:43.950 and multivariate snip.

991 00:40:43.950 --> 00:40:46.030 So there's multivariate tissue.

992 00:40:46.030 --> 00:40:49.464 I guess, are the samples from the same cohort?

993 00:40:49.464 --> 00:40:51.860 - Yeah, so in this particular case,

994 00:40:51.860 --> 00:40:54.830 the samples are from the same individuals.

995 00:40:54.830 --> 00:40:56.180 So these different brain tissues...

996 00:40:56.180 --> 00:40:57.361 There's overlap anyway, let's say.

997 00:40:57.361 --> 00:40:59.920 And so, that's what causes this...

998 00:40:59.920 --> 00:41:02.080 That causes headaches, actually, for this--

999 00:41:02.080 --> 00:41:04.063 - Okay, got it, thanks. - Yeah.

1000 00:41:04.063 --> 00:41:05.300 Just to emphasize,

1001 00:41:05.300 --> 00:41:06.928 it doesn't have to be different tissues.

1002 00:41:06.928 --> 00:41:08.234 The whole method works

1003 00:41:08.234 --> 00:41:10.880 on any matrix of Z scores, basically.

1004 00:41:10.880 --> 00:41:13.750 As long as you think that the rows correspond

1005 00:41:13.750 --> 00:41:14.600 to different tests

1006 00:41:14.600 --> 00:41:15.860 and the columns correspond

1007 00:41:15.860 --> 00:41:20.860 to different, say, conditions for the same test.

1008 00:41:21.046 --> 00:41:24.090 So examples might be

1009 00:41:24.090 --> 00:41:25.170 you're looking at the same snip

1010 00:41:25.170 --> 00:41:27.070 across lots of different phenotypes,

1011 00:41:27.070 --> 00:41:28.670 so looking at schizophrenia,

1012 00:41:28.670 --> 00:41:31.470 looking at bipolar,

1013 00:41:31.470 --> 00:41:33.145 looking at different diseases

1014 00:41:33.145 --> 00:41:34.500 or different traits,

1015 00:41:34.500 --> 00:41:36.942 and you can have a Beta hat for that snip

1016 00:41:36.942 --> 00:41:38.750 and a standard error for that snip

1017 00:41:38.750 --> 00:41:40.060 in every trait.

1018 00:41:40.060 --> 00:41:41.610 And you could try to learn,

1019 00:41:41.610 --> 00:41:43.287 "Oh look, there are some traits

1020 00:41:43.287 --> 00:41:44.697 "that tend to share effects

1021 00:41:44.697 --> 00:41:46.326 "and other traits that don't,"

1022 00:41:46.326 --> 00:41:48.570 or, often in experiments,

1023 00:41:48.570 --> 00:41:50.600 people treat their samples

1024 00:41:50.600 --> 00:41:51.556 with different treatments.

1025 00:41:51.556 --> 00:41:54.060 They challenge them with different viruses.

1026 00:41:54.060 --> 00:41:57.970 They look to see which things are being changed

1027 00:41:57.970 --> 00:42:00.240 when you challenge a cell with different viruses

1028 00:42:00.240 --> 00:42:02.070 or different heat shock treatments

1029 00:42:02.070 --> 00:42:03.338 or any kind of different treatment.

1030 00:42:03.338 --> 00:42:06.325 So yeah, basically, the idea is very generic.

1031 00:42:06.325 --> 00:42:08.587 The idea is if you've got a matrix of Z scores

1032 00:42:08.587 --> 00:42:12.270 where the effect, say, look likely to be shared

1033 00:42:12.270 --> 00:42:13.453 among column sometimes

1034 00:42:13.453 --> 00:42:16.700 and the rows are gonna be approximately independent,

1035 00:42:16.700 --> 00:42:19.780 or at least you're willing to assume that,

1036 00:42:19.780 --> 00:42:21.240 then you can apply the method.

1037 00:42:21.240 --> 00:42:22.460 - Okay, got it, thanks.

1038 00:42:22.460 --> 00:42:24.750 - So, actually, that's an important kind of...

1039 00:42:24.750 --> 00:42:28.030 Also, something that I've been thinking about a lot is

1040 00:42:28.030 --> 00:42:33.030 the benefits of modular or generic methods.

1041 00:42:34.410 --> 00:42:36.100 So if you think about what methods are applied

1042 00:42:36.100 --> 00:42:37.200 in statistics a lot,

1043 00:42:37.200 --> 00:42:39.390 you think T-test, linear regression.

1044 00:42:39.390 --> 00:42:42.190 These are all kind of very generic ideas.

1045 00:42:42.190 --> 00:42:43.690 They don't...

1046 00:42:43.690 --> 00:42:44.880 And Benjamini-Hochberg.

1047 00:42:44.880 --> 00:42:46.590 The nice thing about Benjamini-Hochberg is

1048 00:42:46.590 --> 00:42:48.028 you just need a set of P values

1049 00:42:48.028 --> 00:42:50.040 and you can apply Benjamini-Hochberg.

1050 00:42:50.040 --> 00:42:51.920 You don't have to worry too much

1051 00:42:51.920 --> 00:42:54.420 about where those P values came from.

1052 00:42:54.420 --> 00:42:56.040 So I think, for applications,

1053 00:42:56.040 --> 00:42:57.754 it's really useful to try to think about

1054 00:42:57.754 --> 00:43:01.410 what's the simplest type of data

1055 00:43:01.410 --> 00:43:04.200 you could imagine inputting into the procedure

1056 00:43:04.200 --> 00:43:06.202 in order to output something useful?

1057 00:43:06.202 --> 00:43:08.609 And sometimes, that involves making compromises

1058 00:43:08.609 --> 00:43:11.630 because to make a procedure generic enough,

1059 00:43:11.630 --> 00:43:13.724 you have to compromise on what...

1060 00:43:13.724 --> 00:43:16.740 On maybe what the details of what are going in.

1061 00:43:16.740 --> 00:43:18.270 So here, what we've compromised on is

1062 00:43:18.270 --> 00:43:19.912 that we take a matrix of Z scores,

1063 00:43:19.912 --> 00:43:22.653 or potentially Beta hats and their standard errors,

1064 00:43:22.653 --> 00:43:23.758 we can do either, 1065 00:43:23.758 --> 00:43:25.537 and that's the input.

1066 00:43:25.537 --> 00:43:28.090 So that makes it relatively generic.

1067 00:43:28.090 --> 00:43:29.660 You don't have to worry too much

1068 00:43:29.660 --> 00:43:31.200 about whether those Beta hats

1069 00:43:31.200 --> 00:43:33.150 and the standard errors, or the Z scores,

1070 00:43:33.150 --> 00:43:34.720 are coming from logistic regression

1071 00:43:34.720 --> 00:43:35.810 or linear regression,

1072 00:43:35.810 --> 00:43:37.930 or whether that controlling for some covariance

1073 00:43:37.930 --> 00:43:39.160 or all sorts of...

1074 00:43:39.160 --> 00:43:41.040 From a mixed model, etc.

1075 00:43:41.040 --> 00:43:43.879 As long as they have the basic property that

1076 00:43:43.879 --> 00:43:47.450 the Beta hat is normally distributed

1077 00:43:47.450 --> 00:43:48.330 about the true Beta

1078 00:43:48.330 --> 00:43:50.760 with some variance that you are willing to estimate,

1079 00:43:50.760 --> 00:43:53.033 then you can go.

1080 00:43:58.420 --> 00:43:59.520 - Sorry, (indistinct).

1081 00:44:00.700 --> 00:44:01.533 A short question.

1082 00:44:01.533 --> 00:44:04.650 So in practice, how you choose...

1083 00:44:04.650 --> 00:44:05.930 How many mix...

1084 00:44:05.930 --> 00:44:08.240 How many distribution you want to mixture

1085 00:44:08.240 --> 00:44:09.700 like the (indistinct)? - (indistinct)

1086 00:44:09.700 --> 00:44:11.923 Yeah, so great question.

1087 00:44:11.923 --> 00:44:14.410 And my answer, generally,

1088 00:44:14.410 --> 00:44:15.953 is just use as many as you want.

1089 00:44:15.953 --> 00:44:19.140 So as many as you can stomach.

1090 00:44:19.140 --> 00:44:22.323 The more you use, the slower it is.

1091 00:44:24.687 --> 00:44:26.840 And so, you might worry about over-fitting,

1092 00:44:26.840 --> 00:44:29.010 but it turns out that these procedures are

1093 00:44:29.010 --> 00:44:30.470 very robust to over-fitting

1094 00:44:30.470 --> 00:44:34.350 because of this fact that the mean is fixed at zero.

1095 00:44:34.350 --> 00:44:37.214 So all the components have a mean zero

1096 00:44:37.214 --> 00:44:38.767 and have some covariance

1097 00:44:38.767 --> 00:44:40.046 and because of that,

1098 00:44:40.046 --> 00:44:43.670 they have limited flexibility to overfit.

1099 00:44:45.373 --> 00:44:47.673 They're just not that flexible.

1100 00:44:47.673 --> 00:44:49.670 And in the univariate case,

1101 00:44:49.670 --> 00:44:51.169 that's even more obvious, I think.

1102 00:44:51.169 --> 00:44:52.620 That in the univariate case,

1103 00:44:52.620 --> 00:44:54.884 every one of those distributions,

1104 00:44:54.884 --> 00:44:57.160 any mixture of normals that are

1105 00:44:57.160 --> 00:45:01.100 all centered at zero is unimodal at zero

1106 00:45:01.100 --> 00:45:02.140 and has limited...

1107 00:45:02.140 --> 00:45:03.810 Can't have wiggly distributions

1108 00:45:03.810 --> 00:45:06.188 that are very spiky and overfitting.

1109 00:45:06.188 --> 00:45:08.960 So these methods are relatively immune

1110 00:45:08.960 --> 00:45:11.798 to overfitting in practice.

1111 00:45:11.798 --> 00:45:13.110 If you're worried about that,

1112 00:45:13.110 --> 00:45:15.120 you can do a test-train type thing

1113 00:45:15.120 --> 00:45:17.883 where you use half your tests to train,

1114 00:45:17.883 --> 00:45:20.400 and then you look at the log likelihood

1115 00:45:20.400 --> 00:45:22.160 out of sample on others,

1116 00:45:22.160 --> 00:45:26.820 and then tweak the number to avoid overfitting.

1117 00:45:26.820 --> 00:45:30.160 And we did do that early on in the methods

1118 00:45:30.160 --> 00:45:32.290 but we don't do it very often now,

1119 00:45:32.290 --> 00:45:33.980 or we only do it now when we're worried

1120 00:45:33.980 --> 00:45:37.010 'cause generally it seems like overfitting doesn't seem

1121 00:45:37.010 --> 00:45:37.843 to be a problem,

1122 00:45:37.843 --> 00:45:39.480 but if we see results are a little bit weird

1123 00:45:39.480 --> 00:45:40.568 or a bit concerning,

1124 00:45:40.568 --> 00:45:43.933 we try it to make sure we're not overfitting.

1125 00:45:45.530 --> 00:45:46.719 - Okay, thank you.

1126 00:45:46.719 --> 00:45:48.830 - I should say that, in the paper,

1127 00:45:48.830 --> 00:45:51.190 we kind of outlined some procedures we use

1128 00:45:51.190 --> 00:45:53.670 for estimating these variance, co-variance matrices

1129 00:45:53.670 --> 00:45:54.705 but they're not like...

1130 00:45:54.705 --> 00:45:55.538 They're kind of like...

1131 00:45:57.090 --> 00:46:01.446 The whole philosophy is that we could probably do better

1132 00:46:01.446 --> 00:46:03.250 and we're continuing to try and work

1133 00:46:03.250 --> 00:46:04.894 on better methods for estimating this

1134 00:46:04.894 --> 00:46:06.910 as we go forward.

1135 00:46:06.910 --> 00:46:08.490 So we're continually improving

1136 00:46:08.490 --> 00:46:10.103 the ways we can estimate this.

1137 00:46:15.870 --> 00:46:18.580 Okay, so briefly, I'll talk about linear regression.

1138 00:46:18.580 --> 00:46:21.470 So here's your standard linear regression where,

1139 00:46:21.470 --> 00:46:22.979 so we've N observations,

1140 00:46:22.979 --> 00:46:25.100 X is the matrix of covariates here,

1141 00:46:25.100 --> 00:46:27.317 B are the regression coefficients.

1142 00:46:27.317 --> 00:46:32.210 I'm kind of thinking of P as being big, potentially here.

1143 00:46:32.210 --> 00:46:33.387 And the errors normal.

1144 00:46:33.387 --> 00:46:36.360 And so, the empirical Bayes idea would be

1145 00:46:36.360 --> 00:46:38.330 to assume that the Bs come from

1146 00:46:38.330 --> 00:46:39.890 some prior distribution, G,

1147 00:46:39.890 --> 00:46:42.101 which comes from some family, curly G.

1148 00:46:42.101 --> 00:46:45.300 And what we'd like to do is estimate G

1149 00:46:45.300 --> 00:46:49.247 and then shrink the estimates of B, 1150 00:46:49.247 --> 00:46:52.480 using empirical Bayse type ideas

1151 00:46:52.480 --> 00:46:55.070 and posterior count computations.

1152 00:46:55.070 --> 00:46:58.370 But it's not a simple normal means model here,

1153 00:46:58.370 --> 00:46:59.737 so how do we end up applying

1154 00:46:59.737 --> 00:47:04.110 the empirical Bayse methods to this problem?

1155 00:47:04.110 --> 00:47:05.260 Well, let's just...

1156 00:47:05.260 --> 00:47:07.278 I'm gonna explain our algorithm

1157 00:47:07.278 --> 00:47:11.010 by analogy with penalized regression algorithms

1158 00:47:11.010 --> 00:47:12.810 because the algorithm ends up looking very similar,

1159 00:47:12.810 --> 00:47:13.643 and then I'll tell you

1160 00:47:13.643 --> 00:47:15.501 what the algorithm is actually kind of doing.

1161 00:47:15.501 --> 00:47:19.267 So a penalized regression would solve this problem.

1162 00:47:19.267 --> 00:47:21.530 So if you've seen the Lasso before...

1163 00:47:21.530 --> 00:47:23.120 I hope many of you might have.

1164 00:47:23.120 --> 00:47:24.150 If you've seen the Lasso before,

1165 00:47:24.150 --> 00:47:25.660 this would be solving this problem

1166 00:47:25.660 --> 00:47:29.090 with H being the L1 penalty,

1167 00:47:29.090 --> 00:47:31.120 absolute value of B, right?

1168 00:47:31.120 --> 00:47:32.420 So this...

1169 00:47:32.420 --> 00:47:33.910 So what algorithm...

1170 00:47:33.910 --> 00:47:36.510 There are many, many algorithms to solve this problem

1171 00:47:36.510 --> 00:47:38.773 but a very simple one is coordinate ascent.

1172 00:47:39.900 --> 00:47:41.790 So essentially, for each coordi...

1173 00:47:41.790 --> 00:47:43.048 it just iterates the following.

1174 00:47:43.048 --> 00:47:44.140 For each coordinate,

1175 00:47:44.140 --> 00:47:46.757 you have some kind of current estimate for Bs.

1176 00:47:46.757 --> 00:47:47.850 (indistinct)

1177 00:47:47.850 --> 00:47:49.929 So what you do here is you form the residuals

1178 00:47:49.929 --> 00:47:54.929 by taking away the effects of all the Bs

1179 00:47:55.270 --> 00:47:56.980 except the one you're trying to update,

1180 00:47:56.980 --> 00:47:58.520 the one you're trying to estimate.

1181 00:47:58.520 --> 00:48:00.760 So X minus J here is all the covariates

1182 00:48:00.760 --> 00:48:02.140 except covariate J.

1183 00:48:02.140 --> 00:48:06.412 B minus J is all the corresponding coefficients.

1184 00:48:06.412 --> 00:48:08.100 So this is the residual.

1185 00:48:08.100 --> 00:48:09.117 RJ is the residual,

1186 00:48:09.117 --> 00:48:12.410 after removing all the current estimated effects

1187 00:48:12.410 --> 00:48:14.495 apart from the Jth one.

1188 00:48:14.495 --> 00:48:18.340 And then you basically compute a estimate

1189 00:48:18.340 --> 00:48:23.340 of the Jth effect by regressing those residuals on XJ.

1190 00:48:23.607 --> 00:48:28.607 And then you shrink that using a shrinkage operator

1191 00:48:29.296 --> 00:48:31.130 that we saw earlier.

1192 00:48:31.130 --> 00:48:32.543 Just to remind you

1193 00:48:32.543 --> 00:48:34.130 that a shrinkage operator is the one

1194 00:48:34.130 --> 00:48:37.854 that minimizes this penalized least squares problem.

1195 00:48:37.854 --> 00:48:39.227 And it turns out,

1196 00:48:39.227 --> 00:48:44.227 it's not hard to show that this is coordinate ascent

1197 00:48:44.571 --> 00:48:48.981 for minimizing this, penalized objective function.

1198 00:48:48.981 --> 00:48:53.250 And so every iteration of this increases

1199 00:48:53.250 --> 00:48:54.530 that objective function

1200 00:48:54.530 --> 00:48:55.453 or decreases it.

1201 00:48:57.810 --> 00:48:58.643 Okay.

1202 00:48:59.605 --> 00:49:01.450 Okay, so it turns...

1203 00:49:01.450 --> 00:49:03.297 So our algorithm looks very similar.

1204 00:49:03.297 --> 00:49:05.642 You still compute the residuals,

1205 00:49:05.642 --> 00:49:07.550 you compute a Beta hat

1206 00:49:07.550 --> 00:49:09.750 by regressing the residuals on XJ.

1207 00:49:09.750 --> 00:49:10.980 You also, at the same time,

1208 00:49:10.980 --> 00:49:12.371 compute a standard error,

1209 00:49:12.371 --> 00:49:15.623 which is familiar form.

1210 00:49:17.489 --> 00:49:20.670 And then you, instead of shrinking using

1211 00:49:20.670 --> 00:49:22.990 that penalized regression operator,

1212 00:49:22.990 --> 00:49:24.527 you use a...

1213 00:49:25.400 --> 00:49:26.610 Sorry, I should say,

1214 00:49:26.610 --> 00:49:28.210 this is assuming G is known.

1215 00:49:28.210 --> 00:49:29.280 I'm starting with G.

1216 00:49:29.280 --> 00:49:30.580 G is known.

1217 00:49:30.580 --> 00:49:32.130 So you can shrink...

1218 00:49:32.130 --> 00:49:35.010 Instead of using the penalty-based method,

1219 00:49:35.010 --> 00:49:37.720 you use the posterior mean shrinkage operator here

1220 00:49:37.720 --> 00:49:39.600 that I introduced earlier.

1221 00:49:39.600 --> 00:49:41.470 So it's basically exactly the same algorithm,

1222 00:49:41.470 --> 00:49:46.470 except replacing this penalty-based shrinkage operator

1223 00:49:46.834 --> 00:49:48.383 with an empirical Bayse

1224 00:49:48.383 --> 00:49:50.653 or a Bayesean shrinkage operator.

1225 00:49:53.640 --> 00:49:55.540 And so, you could ask what that's doing

1226 00:49:55.540 --> 00:49:57.288 and it turns out that what it's doing is

1227 00:49:57.288 --> 00:50:01.248 it's minimizing the Kullback-Leibler Divergence

1228 00:50:01.248 --> 00:50:04.864 between some approximate posterior, Q,

1229 00:50:04.864 --> 00:50:08.380 and the true posterior, P, here

1230 00:50:08.380 --> 00:50:13.380 under the constraint that this Q is factorized.

1231 00:50:13.490 --> 00:50:14.560 So this is what's called

1232 00:50:14.560 --> 00:50:16.610 a variational approximation

1233 00:50:16.610 --> 00:50:17.443 or a mean-field,

1234 00:50:17.443 --> 00:50:20.752 or fully factorized variational approximation.

1235 00:50:20.752 --> 00:50:22.150 If you've seen that before,

1236 00:50:22.150 --> 00:50:23.507 you'll know what's going on here.

1237 00:50:23.507 --> 00:50:25.030 If you haven't seen it before,

1238 00:50:25.030 --> 00:50:27.190 it's trying to find an approximation

1239 00:50:27.190 --> 00:50:28.670 to the posterior.

1240 00:50:28.670 --> 00:50:29.790 This is the true posterior,

1241 00:50:29.790 --> 00:50:31.280 it's trying to find an approximation

1242 00:50:31.280 --> 00:50:32.836 to that posterior that minimizes

1243 00:50:32.836 --> 00:50:34.540 the Kullbert-Leibler Divergence

1244 00:50:34.540 --> 00:50:36.365 between the approximation

1245 00:50:36.365 --> 00:50:39.530 and the true value under in a simplifying assumption

1246 00:50:39.530 --> 00:50:40.810 that the posterior factorizes,

1247 00:50:40.810 --> 00:50:41.830 which, of course, it doesn't,

1248 00:50:41.830 --> 00:50:43.910 so that's why it's an approximation.

1249 00:50:43.910 --> 00:50:46.335 So that algorithm I just said is

1250 00:50:46.335 --> 00:50:48.200 a coordinate ascent algorithm

1251 00:50:48.200 --> 00:50:52.880 for maximizing F or minimizing the KL divergence.

1252 00:50:52.880 --> 00:50:55.258 So every iteration of that algorithm gets

1253 00:50:55.258 --> 00:50:58.180 a better estimate estimate

1254 00:50:58.180 --> 00:51:00.107 of the posterior, essentially.

1255 00:51:02.380 --> 00:51:03.370 Just to outline

1256 00:51:03.370 --> 00:51:05.750 and just to give you the intuition

1257 00:51:05.750 --> 00:51:07.720 for how you could maybe estimate G,

1258 00:51:07.720 --> 00:51:09.826 this isn't actually quite what we do

1259 00:51:09.826 --> 00:51:12.620 so the details get a bit more complicated,

1260 00:51:12.620 --> 00:51:14.200 but just to give you an intuition

1261 00:51:14.200 --> 00:51:17.728 for how you might think that you can estimate G;

1262 00:51:17.728 --> 00:51:21.157 Every iteration of this algorithm computes a B hat

1263 00:51:21.157 --> 00:51:23.320 and a corresponding standard error,

1264 00:51:23.320 --> 00:51:24.658 so you could imagine...

1265 00:51:24.658 --> 00:51:28.360 These two steps here, you could imagine storing these

1266 00:51:28.360 --> 00:51:29.362 through the iterations

1267 00:51:29.362 --> 00:51:30.515 and, at the end,

1268 00:51:30.515 --> 00:51:35.090 you could apply the empirical Bayes normal means procedure

1269 00:51:35.090 --> 00:51:37.370 to estimate G from these B hats

1270 00:51:37.370 --> 00:51:38.404 and standard errors,

1271 00:51:38.404 --> 00:51:43.210 and something close to that kind of works.

1272 00:51:43.210 --> 00:51:45.660 The details are a bit more complicated than that.

1273 00:51:46.830 --> 00:51:51.460 So let me give you some kind of intuition

1274 00:51:51.460 --> 00:51:53.770 for what we're trying to achieve here based

1275 00:51:53.770 --> 00:51:54.770 on simulation results.

1276 00:51:54.770 --> 00:51:57.140 So these are some simulations we've done.

1277 00:51:57.140 --> 00:51:59.740 The covariates are all independent here.

1278 00:51:59.740 --> 00:52:01.990 The true prior is a point normal,

1279 00:52:01.990 --> 00:52:06.867 that means that most of the effects are zero.

1280 00:52:06.867 --> 00:52:09.116 Well, actually maybe here,

1281 00:52:09.116 --> 00:52:10.915 one of the effects is nonzero,

1282 00:52:10.915 --> 00:52:12.821 five of the effects is nonzero,

1283 00:52:12.821 --> 00:52:14.720 50 of the effects are nonzero

1284 00:52:14.720 --> 00:52:16.970 and 500 of the effects of nonzero.

1285 00:52:16.970 --> 00:52:18.404 And actually, there are 500 effects,

1286 00:52:18.404 --> 00:52:20.963 500 variables in this example.

1287 00:52:22.687 --> 00:52:25.260 So the X-axis here just shows the number

1288 00:52:25.260 --> 00:52:26.966 of non-zero coordinates

1289 00:52:26.966 --> 00:52:30.180 and the results I've shown here are the prediction error,

1290 00:52:30.180 --> 00:52:31.749 so we're focusing on prediction error,

1291 00:52:31.749 --> 00:52:33.430 the out of sample prediction error,

1292 00:52:33.430 --> 00:52:36.945 using three different penalty-based approaches.

1293 00:52:36.945 --> 00:52:39.994 The Lasso, which is this line,

1294 00:52:39.994 --> 00:52:42.980 the L0Learn, which is this line,

1295 00:52:42.980 --> 00:52:44.970 which is L0 zero penalty,

1296 00:52:44.970 --> 00:52:46.860 and Ridge, which is this penalty,

1297 00:52:46.860 --> 00:52:48.178 the L2 penalty.

1298 00:52:48.178 --> 00:52:51.150 So the important thing to know is that

1299 00:52:51.150 --> 00:52:54.844 the L0 penalty is really designed, if you like,

1300 00:52:54.844 --> 00:52:57.753 to do well under very sparse models.

1301 00:52:57.753 --> 00:53:01.520 So that's why it's got the lowest prediction error

1302 00:53:02.750 --> 00:53:04.770 when the model is very sparse,

1303 00:53:04.770 --> 00:53:07.130 but when the model is completely densed,

1304 00:53:07.130 --> 00:53:08.299 it does very poorly.

1305 00:53:08.299 --> 00:53:13.299 Whereas Ridge is designed much more to...

1306 00:53:13.640 --> 00:53:14.910 It's actually based on a prior

1307 00:53:14.910 --> 00:53:17.120 that the effects are normally distributed.

1308 00:53:17.120 --> 00:53:18.810 So it's much better at dense models

1309 00:53:18.810 --> 00:53:20.110 than sparse models.

1310 00:53:20.110 --> 00:53:22.970 And you can see that at least relative to L0Learn,

1311 00:53:22.970 --> 00:53:25.992 Ridge is much better for the dense case

1312 00:53:25.992 --> 00:53:29.556 but also much worse for the sparse case.

1313 00:53:29.556 --> 00:53:32.205 And then Lasso has some kind of ability

1314 00:53:32.205 --> 00:53:34.700 to deal with both scenarios,

1315 00:53:34.700 --> 00:53:37.390 but it's not quite as good as the L0 penalty

1316 00:53:37.390 --> 00:53:38.691 when things are very sparse,

1317 00:53:38.691 --> 00:53:41.060 and it's not quite as good as the Ridge penalty

1318 00:53:41.060 --> 00:53:43.253 when things are very dense.

1319 00:53:44.650 --> 00:53:48.790 So our goal is that by learning the prior G

1320 00:53:48.790 --> 00:53:49.940 from the data,

1321 00:53:49.940 --> 00:53:52.824 we can adapt to each of these scenarios

1322 00:53:52.824 --> 00:53:55.454 and get performance close to L0Learn

1323 00:53:55.454 --> 00:53:57.843 when the truth is sparse

1324 00:53:57.843 --> 00:54:00.610 and get performance close to Ridge regression

1325 00:54:00.610 --> 00:54:02.890 when the truth is dense.

1326 00:54:02.890 --> 00:54:06.450 And so, the red line here actually shows

1327 00:54:06.450 --> 00:54:08.997 the performance of our method.

1328 00:54:08.997 --> 00:54:10.130 And you can see, indeed,

1329 00:54:10.130 --> 00:54:12.880 we do even slightly better than L0Learn

1330 00:54:12.880 --> 00:54:14.243 in this part here

1331 00:54:14.243 --> 00:54:17.657 and slightly better than cross-validated Ridge regression

1332 00:54:17.657 --> 00:54:18.737 in this here.

1333 00:54:18.737 --> 00:54:21.037 The difference between these two is just that

1334 00:54:21.037 --> 00:54:23.360 the Ridge regression is doing cross-validation

1335 00:54:23.360 --> 00:54:24.690 to estimate the tuning parameter

1336 00:54:24.690 --> 00:54:27.650 and we're using empirical Bayse maximum likelihood

1337 00:54:27.650 --> 00:54:28.483 to estimate it.

1338 00:54:28.483 --> 00:54:29.960 So that's just that difference there.

1339 00:54:29.960 --> 00:54:31.927 And the Oracle here is using the true...

1340 00:54:31.927 --> 00:54:34.700 You can do the Oracle computation

1341 00:54:34.700 --> 00:54:35.610 for the Ridge regression

1342 00:54:35.610 --> 00:54:37.603 with the true tuning parameter here.

1343 00:54:37.603 --> 00:54:41.598 I should say that may be that this...

1344 00:54:41.598 --> 00:54:45.040 Maybe I should just show you the results.

1345 00:54:45.040 --> 00:54:47.010 So here is a bunch of other penalties,

1346 00:54:47.010 --> 00:54:49.109 including elastic net, for example, you might wonder,

1347 00:54:49.109 --> 00:54:50.980 which is kind of a compromise

1348 00:54:50.980 --> 00:54:52.290 between L1 and L2.

1349 00:54:52.290 --> 00:54:55.790 And you can see, it does do the compromising

1350 00:54:55.790 --> 00:54:56.900 but it doesn't do as well

1351 00:54:56.900 --> 00:54:59.170 as the empirical Bayse approach.

1352 00:54:59.170 --> 00:55:01.380 And here are some other non-convex methods

1353 00:55:01.380 --> 00:55:03.700 that are more, again...

1354 00:55:03.700 --> 00:55:06.110 They're kind of more tuned to the sparse case

1355 00:55:06.110 --> 00:55:07.313 than to the dense case.

1356 00:55:09.212 --> 00:55:11.980 As promised, I'm gonna skip over

1357 00:55:11.980 --> 00:55:14.030 the matrix factorization

1358 00:55:14.030 --> 00:55:18.320 and just summarize to give time for questions.

1359 00:55:18.320 --> 00:55:20.900 So the summary is that

1360 00:55:20.900 --> 00:55:23.130 the empirical Bayse normal means model provides

1361 00:55:23.130 --> 00:55:24.740 a flexible and convenient way

1362 00:55:24.740 --> 00:55:26.390 to induce shrinkage and sparsity

1363 00:55:26.390 --> 00:55:27.978 in a range of applications.

1364 00:55:27.978 --> 00:55:31.780 And we've been spending a lot of time trying

1365 00:55:31.780 --> 00:55:32.875 to apply these methods

1366 00:55:32.875 --> 00:55:34.610 and provide software to do

1367 00:55:34.610 --> 00:55:36.210 some of these different things.

1368 00:55:36.210 --> 00:55:38.580 And there's a bunch of things on my publications page

1369 00:55:38.580 --> 00:55:39.550 and if you're interested in...

1370 00:55:39.550 --> 00:55:40.991 If you can't find what you're looking for,

1371 00:55:40.991 --> 00:55:43.686 just let me know, I'd be happy to point you to it.

1372 00:55:43.686 --> 00:55:44.803 Thanks very much.

1373 00:55:47.149 --> 00:55:49.490 - Thanks Matthew, that's a great talk.

1374 00:55:49.490 --> 00:55:51.330 I wonder whether the audience have

1375 00:55:51.330 --> 00:55:52.630 any questions for Matthew.

1376 00:55:57.110 --> 00:55:59.730 So I do have some questions for you.

1377 00:55:59.730 --> 00:56:01.890 So I think I really like the idea

1378 00:56:01.890 --> 00:56:05.750 of applying empirical Bayes to a lot of applications

1379 00:56:05.750 --> 00:56:10.033 and it's really seems empirical Bayes has great success.

1380 00:56:10.033 --> 00:56:13.400 But I do have a question or some doubt

1381 00:56:13.400 --> 00:56:14.952 about the inference part,

1382 00:56:14.952 --> 00:56:17.672 especially in that linear regression model.

1383 00:56:17.672 --> 00:56:22.111 So currently, for the current work you have been doing,

1384 00:56:22.111 --> 00:56:24.380 you essentially shrink each

1385 00:56:24.380 --> 00:56:26.440 of the co-efficient that based on, essentially,

1386 00:56:26.440 --> 00:56:28.232 their estimated value,

1387 00:56:28.232 --> 00:56:30.810 but in some applications,

1388 00:56:30.810 --> 00:56:33.186 such as a GWAS study or fine mapping,

1389 00:56:33.186 --> 00:56:34.953 different snips can have

1390 00:56:34.953 --> 00:56:37.640 very different LD score structure.

1391 00:56:37.640 --> 00:56:38.800 So in this case,

1392 00:56:38.800 --> 00:56:43.447 how much we can trust the inference,

1393 00:56:43.447 --> 00:56:46.977 the P value, from this (indistinct)?

1394 00:56:48.414 --> 00:56:51.420 - So, great question.

1395 00:56:51.420 --> 00:56:53.510 So let me just first...

1396 00:56:55.213 --> 00:56:57.260 First emphasize that the shrink...

1397 00:56:57.260 --> 00:57:01.450 The estimate here is being done removing the effects,

1398 00:57:01.450 --> 00:57:02.510 or the estimated effects,

1399 00:57:02.510 --> 00:57:03.950 of all the other variables.

1400 00:57:03.950 --> 00:57:05.527 So each iteration of this,

1401 00:57:05.527 --> 00:57:06.970 when you're estimating the effect

1402 00:57:06.970 --> 00:57:09.590 of snip J, in your example,

1403 00:57:09.590 --> 00:57:11.265 you're taking the estimated effects

1404 00:57:11.265 --> 00:57:14.125 of the other variables into account.

1405 00:57:14.125 --> 00:57:18.300 So the LD structure, as you mentioned,

1406 00:57:18.300 --> 00:57:19.520 that's the correlation structure

1407 00:57:19.520 --> 00:57:20.580 for those who don't know,

1408 00:57:20.580 --> 00:57:23.640 between the Xs is formerly taken into account.

1409 00:57:23.640 --> 00:57:26.287 However, there is a problem with this approach

1410 00:57:26.287 --> 00:57:29.714 for very highly correlated variables.

1411 00:57:29.714 --> 00:57:33.800 So let's just suppose there are two variables

1412 00:57:33.800 --> 00:57:35.016 that are completely correlated,

1413 00:57:35.016 --> 00:57:37.674 what does this algorithm end up doing?

1414 00:57:37.674 --> 00:57:40.130 It ends up basically choosing one of them

1415 00:57:40.130 --> 00:57:41.280 and ignoring the other.

1416 00:57:43.851 --> 00:57:46.372 The Lasso does the same in fact.

1417 00:57:46.372 --> 00:57:50.481 So it ends up choosing one of them

1418 00:57:50.481 --> 00:57:52.048 and ignoring the other.

1419 00:57:52.048 --> 00:57:55.450 And if you look to the posterior distribution

1420 00:57:55.450 --> 00:57:56.510 on its effect,

1421 00:57:56.510 --> 00:57:57.990 it would be far too confident

1422 00:57:57.990 --> 00:57:59.530 in the size of the effect

1423 00:57:59.530 --> 00:58:03.601 because it would assume that the other one had zero effect.

1424 00:58:03.601 --> 00:58:06.260 And so it would have a small credible

1425 00:58:06.260 --> 00:58:07.820 and for, let's say, around the effect size

1426 00:58:07.820 --> 00:58:09.030 when, really, it should be saying

1427 00:58:09.030 --> 00:58:11.324 you don't know which one to include.

1428 00:58:11.324 --> 00:58:14.079 And so, we've worked recently

1429 00:58:14.079 --> 00:58:16.964 on a method for doing that.

1430 00:58:16.964 --> 00:58:18.565 A different method,

1431 00:58:18.565 --> 00:58:21.890 different work than what I've just described here

1432 00:58:21.890 --> 00:58:25.130 for doing fine mapping using variational approximations

1433 00:58:25.130 --> 00:58:27.144 that don't have this problem,

1434 00:58:27.144 --> 00:58:29.380 and it's on my webpage.

1435 00:58:29.380 --> 00:58:34.210 It's Wang et al in JRSS-B,

1436 00:58:34.210 --> 00:58:35.723 just recently, this year.

1437 00:58:35.723 --> 00:58:39.387 2021, I guess. - That's awesome.

1438 00:58:39.387 --> 00:58:44.387 Thanks, so any more question for Matthew from the audience?

1439 00:58:47.260 --> 00:58:50.559 Okay, I think we're of running out of time also.

1440 00:58:50.559 --> 00:58:52.780 So if you have any question

1441 00:58:52.780 --> 00:58:54.290 about the stuff to (indistinct)

1442 00:58:54.290 --> 00:58:55.123 you want to use,

1443 00:58:55.123 --> 00:58:56.329 I think you can contact either

1444 00:58:56.329 --> 00:59:00.109 the authors of the paper or Matthew off the line.

1445 00:59:00.109 --> 00:59:03.640 And thank you again for agreeing to present your work here.

1446 00:59:03.640 --> 00:59:07.026 It's looks really useful and interesting.

1447 00:59:07.026 --> 00:59:08.423 - Thank you for having me.