**Deep Learning Neural Network H2O result**

> summary(dl.model)

Model Details:

==============

H2OBinomialModel: deeplearning

Model Key: DeepLearning\_model\_R\_1542593485281\_1

Status of Neuron Layers: predicting Label, 2-class classification, bernoulli distribution, CrossEntropy loss, 45,202 weights/biases, 540.8 KB, 270,583 training samples, mini-batch size 1

layer units type dropout l1 l2 mean\_rate rate\_rms momentum

1 1 22 Input 0.00 % NA NA NA NA NA

2 2 200 Tanh 0.00 % 0.000000 0.000000 0.106349 0.151768 0.000000

3 3 200 Tanh 0.00 % 0.000000 0.000000 0.356311 0.214840 0.000000

4 4 2 Softmax NA 0.000000 0.000000 0.002564 0.000728 0.000000

mean\_weight weight\_rms mean\_bias bias\_rms

1 NA NA NA NA

2 -0.007482 0.214525 0.008684 0.172861

3 -0.000131 0.094033 0.040507 0.401267

4 0.032631 0.246762 -0.000099 0.718862

H2OBinomialMetrics: deeplearning

\*\* Reported on training data. \*\*

\*\* Metrics reported on temporary training frame with 9847 samples \*\*

MSE: 0.1044441

RMSE: 0.3231782

LogLoss: 0.3570206

Mean Per-Class Error: 0.4884238

AUC: 0.6715266

Gini: 0.3430533

Confusion Matrix (vertical: actual; across: predicted) for F1-optimal threshold:

0 1 Error Rate

0 29 1205 0.976499 =1205/1234

1 3 8610 0.000348 =3/8613

Totals 32 9815 0.122677 =1208/9847

Maximum Metrics: Maximum metrics at their respective thresholds

metric threshold value idx

1 max f1 0.668857 0.934448 385

2 max f2 0.668857 0.972508 385

3 max f0point5 0.804926 0.902776 312

4 max accuracy 0.668857 0.877323 385

5 max precision 0.973270 1.000000 0

6 max recall 0.339013 1.000000 398

7 max specificity 0.973270 1.000000 0

8 max absolute\_mcc 0.844055 0.203963 257

9 max min\_per\_class\_accuracy 0.898052 0.629978 163

10 max mean\_per\_class\_accuracy 0.879691 0.632843 197

Gains/Lift Table: Extract with `h2o.gainsLift(<model>, <data>)` or `h2o.gainsLift(<model>, valid=<T/F>, xval=<T/F>)`

H2OBinomialMetrics: deeplearning

\*\* Reported on validation data. \*\*

\*\* Metrics reported on full validation frame \*\*

MSE: 0.1140994

RMSE: 0.3377861

LogLoss: 0.3851992

Mean Per-Class Error: 0.4885811

AUC: 0.6487866

Gini: 0.2975733

Confusion Matrix (vertical: actual; across: predicted) for F1-optimal threshold:

0 1 Error Rate

0 21 864 0.976271 =864/885

1 5 5606 0.000891 =5/5611

Totals 26 6470 0.133775 =869/6496

Maximum Metrics: Maximum metrics at their respective thresholds

metric threshold value idx

1 max f1 0.689542 0.928069 385

2 max f2 0.689542 0.969427 385

3 max f0point5 0.775536 0.891076 345

4 max accuracy 0.695926 0.866225 383

5 max precision 0.973201 1.000000 0

6 max recall 0.339013 1.000000 399

7 max specificity 0.973201 1.000000 0

8 max absolute\_mcc 0.851091 0.163335 250

9 max min\_per\_class\_accuracy 0.900900 0.607910 162

10 max mean\_per\_class\_accuracy 0.901665 0.610619 161

Gains/Lift Table: Extract with `h2o.gainsLift(<model>, <data>)` or `h2o.gainsLift(<model>, valid=<T/F>, xval=<T/F>)`

Scoring History:

timestamp duration training\_speed epochs iterations

1 2018-11-18 18:11:35 0.000 sec NA 0.00000 0

2 2018-11-18 18:11:46 13.347 sec 1999 obs/sec 0.73007 1

3 2018-11-18 18:12:15 42.105 sec 2099 obs/sec 2.92970 4

4 2018-11-18 18:12:36 1 min 2.283 sec 2080 obs/sec 4.39424 6

5 2018-11-18 18:12:55 1 min 21.295 sec 2098 obs/sec 5.85295 8

6 2018-11-18 18:13:14 1 min 40.214 sec 2105 obs/sec 7.30951 10

7 2018-11-18 18:13:24 1 min 49.900 sec 2110 obs/sec 8.04038 11

8 2018-11-18 18:13:33 1 min 59.562 sec 2117 obs/sec 8.76679 12

9 2018-11-18 18:13:44 2 min 9.637 sec 2113 obs/sec 9.49777 13

10 2018-11-18 18:13:53 2 min 19.103 sec 2120 obs/sec 10.22071 14

samples training\_rmse training\_logloss training\_r2 training\_auc

1 0.000000 NA NA NA NA

2 19328.000000 0.34599 0.41234 -0.09209 0.57166

3 77561.000000 0.32594 0.36640 0.03082 0.63959

4 116333.000000 0.32719 0.36688 0.02338 0.63244

5 154951.000000 0.32726 0.36873 0.02293 0.64126

6 193512.000000 0.32631 0.36600 0.02861 0.64971

7 212861.000000 0.32491 0.36164 0.03691 0.65703

8 232092.000000 0.32628 0.36624 0.02878 0.65929

9 251444.000000 0.32522 0.36131 0.03508 0.65713

10 270583.000000 0.32318 0.35702 0.04715 0.67153

training\_lift training\_classification\_error validation\_rmse

1 NA NA NA

2 1.07468 0.12522 0.36241

3 1.10863 0.12420 0.33934

4 1.05181 0.12450 0.34193

5 1.10863 0.12511 0.34154

6 1.04038 0.12471 0.33943

7 1.10965 0.12471 0.33940

8 1.12085 0.12461 0.34130

9 1.05181 0.12450 0.33986

10 1.12018 0.12268 0.33779

validation\_logloss validation\_r2 validation\_auc validation\_lift

1 NA NA NA NA

2 0.45035 -0.11614 0.54665 1.08756

3 0.39297 0.02146 0.62451 1.13991

4 0.39682 0.00648 0.59774 1.08648

5 0.39815 0.00872 0.61970 1.15773

6 0.39124 0.02094 0.64056 1.10429

7 0.38911 0.02111 0.63732 1.13991

8 0.39754 0.01014 0.63382 1.14045

9 0.38825 0.01848 0.62872 1.12210

10 0.38520 0.03040 0.64879 1.13991

validation\_classification\_error

1 NA

2 0.13624

3 0.13608

4 0.13578

5 0.13624

6 0.13547

7 0.13624

8 0.13593

9 0.13562

10 0.13377

Variable Importances: (Extract with `h2o.varimp`)

=================================================

Variable Importances:

variable relative\_importance scaled\_importance percentage

1 B3 1.000000 1.000000 0.091875

2 B1 0.737113 0.737113 0.067722

3 B4 0.674429 0.674429 0.061963

4 B2 0.627872 0.627872 0.057686

5 A4 0.571214 0.571214 0.052480

---

variable relative\_importance scaled\_importance percentage

17 Param5 0.382370 0.382370 0.035130

18 BN1 0.367622 0.367622 0.033775

19 Param3 0.338942 0.338942 0.031140

20 B5 0.324115 0.324115 0.029778

21 A1 0.314279 0.314279 0.028874

22 Mix50.50 0.287239 0.287239 0.026390

> plot(dl.model)

>

> dl.model.predict <- h2o.predict(dl.model, test.hex)

|======================================================================| 100%

> dl.result <- as.data.frame(dl.model.predict)

> dl.result

predict p0 p1

1 1 0.04266477 0.9573352

2 1 0.04068659 0.9593134

3 1 0.07507370 0.9249263

4 1 0.04263091 0.9573691

5 1 0.07106908 0.9289309

6 1 0.05908374 0.9409163

7 1 0.04972925 0.9502708

8 1 0.05178540 0.9482146

9 1 0.10020639 0.8997936

10 1 0.11611099 0.8838890

11 1 0.11446021 0.8855398

12 1 0.10251066 0.8974893

13 1 0.06199187 0.9380081

14 1 0.04740796 0.9525920

15 1 0.05779640 0.9422036

16 1 0.05783790 0.9421621

17 1 0.05732448 0.9426755

18 1 0.06456525 0.9354347

19 1 0.06169807 0.9383019

20 1 0.06094172 0.9390583

21 1 0.17088427 0.8291157

22 1 0.13310288 0.8668971

23 1 0.07228295 0.9277170

24 1 0.06041919 0.9395808

25 1 0.05151501 0.9484850

26 1 0.07266926 0.9273307

27 1 0.04118097 0.9588190

28 1 0.04263091 0.9573691

29 1 0.07106908 0.9289309

30 1 0.06772018 0.9322798

31 1 0.07168757 0.9283124

32 1 0.10123590 0.8987641

33 1 0.04972761 0.9502724

34 1 0.06852063 0.9314794

35 1 0.19111903 0.8088810

36 1 0.13167140 0.8683286

37 1 0.14769007 0.8523099

38 1 0.06260763 0.9373924

39 1 0.23237819 0.7676218

40 1 0.21719875 0.7828012

41 1 0.05912854 0.9408715

42 1 0.06024013 0.9397599

43 1 0.06019304 0.9398070

44 1 0.06702665 0.9329733

45 1 0.08252497 0.9174750

46 1 0.09106550 0.9089345

47 1 0.09402361 0.9059764

48 1 0.04489241 0.9551076

49 1 0.02994767 0.9700523

50 1 0.04507604 0.9549240

51 1 0.15532576 0.8446742

52 1 0.10746703 0.8925330

53 1 0.07248995 0.9275100

54 1 0.10070621 0.8992938

55 1 0.10206940 0.8979306

56 1 0.07363483 0.9263652

57 1 0.09648996 0.9035100

58 1 0.11351675 0.8864832

59 1 0.18181310 0.8181869

60 1 0.08911787 0.9108821

61 1 0.05482998 0.9451700

62 1 0.05467630 0.9453237

63 1 0.05632284 0.9436772

64 1 0.06473555 0.9352645

65 1 0.04549175 0.9545082

66 1 0.04616573 0.9538343

67 1 0.05581328 0.9441867

68 1 0.15353439 0.8464656

69 1 0.14459208 0.8554079

70 1 0.14946368 0.8505363

71 1 0.08809567 0.9119043

72 1 0.12231223 0.8776878

73 1 0.10899099 0.8910090

74 1 0.07345483 0.9265452

75 1 0.05706444 0.9429356

76 1 0.06283587 0.9371641

77 1 0.06260763 0.9373924

78 1 0.18357254 0.8164275

79 1 0.18329713 0.8167029

80 1 0.20016483 0.7998352

81 1 0.06821143 0.9317886

82 1 0.08870463 0.9112954

83 1 0.07231941 0.9276806

84 1 0.08021383 0.9197862

85 1 0.05342001 0.9465800

86 1 0.09571198 0.9042880

87 1 0.15660332 0.8433967

88 1 0.11643517 0.8835648

89 1 0.11139295 0.8886070

90 1 0.10931193 0.8906881

91 1 0.10582339 0.8941766

92 1 0.14312997 0.8568700

93 1 0.09721288 0.9027871

94 1 0.07051901 0.9294810

95 1 0.10357847 0.8964215

96 1 0.09209623 0.9079038

97 1 0.06773463 0.9322654

98 1 0.09167468 0.9083253

99 1 0.08490253 0.9150975

100 1 0.08672696 0.9132730

101 1 0.05946879 0.9405312

102 1 0.08781684 0.9121832

103 1 0.08567279 0.9143272

104 1 0.05792681 0.9420732

105 1 0.11117030 0.8888297

106 1 0.10958813 0.8904119

107 1 0.06843875 0.9315613

108 1 0.08339978 0.9166002

109 1 0.08334194 0.9166581

110 1 0.06276385 0.9372362

111 1 0.06723256 0.9327674

112 1 0.06019304 0.9398070

113 1 0.07486906 0.9251309

114 1 0.05720096 0.9427990

115 1 0.05224961 0.9477504

116 1 0.05255198 0.9474480

117 1 0.05734651 0.9426535

118 1 0.04118097 0.9588190

119 1 0.06645099 0.9335490

120 1 0.02850769 0.9714923

121 1 0.05609261 0.9439074

122 1 0.10896363 0.8910364

123 1 0.31029770 0.6897023

124 1 0.11801493 0.8819851

125 1 0.10603604 0.8939640

126 1 0.08495298 0.9150470

127 1 0.16350026 0.8364997

128 1 0.22793016 0.7720698

129 1 0.15600317 0.8439968

130 1 0.15434795 0.8456520

131 1 0.16208434 0.8379157

132 1 0.15497229 0.8450277

133 1 0.12737674 0.8726233

134 1 0.12456241 0.8754376

135 1 0.12208248 0.8779175

136 1 0.13445040 0.8655496

137 1 0.05247608 0.9475239

138 1 0.08122799 0.9187720

139 1 0.06847845 0.9315215

140 1 0.05238027 0.9476197

141 1 0.06733691 0.9326631

142 1 0.05232776 0.9476722

143 1 0.05336915 0.9466309

144 1 0.06277156 0.9372284

145 1 0.07572761 0.9242724

146 1 0.06268701 0.9373130

147 1 0.06516066 0.9348393

148 1 0.11446021 0.8855398

149 1 0.10466842 0.8953316

150 1 0.06788717 0.9321128

151 1 0.06771711 0.9322829

152 1 0.05811848 0.9418815

153 1 0.05887231 0.9411277

154 1 0.05986236 0.9401376

155 1 0.05018725 0.9498128

156 1 0.06529744 0.9347026

157 1 0.06301270 0.9369873

158 1 0.09205506 0.9079449

159 1 0.21713656 0.7828634

160 1 0.18156985 0.8184301

161 1 0.16903796 0.8309620

162 1 0.15248969 0.8475103

163 1 0.12830957 0.8716904

164 1 0.12232411 0.8776759

165 1 0.07553241 0.9244676

166 1 0.07593595 0.9240640

167 1 0.07563993 0.9243601

168 1 0.06934669 0.9306533

169 1 0.19761045 0.8023896

170 1 0.06670401 0.9332960

171 1 0.06645995 0.9335401

172 1 0.11719550 0.8828045

173 1 0.11950327 0.8804967

174 1 0.06975806 0.9302419

175 1 0.07168757 0.9283124

176 1 0.05086893 0.9491311

177 1 0.11538084 0.8846192

178 1 0.05432869 0.9456713

179 1 0.06041919 0.9395808

180 1 0.07103519 0.9289648

181 1 0.05151501 0.9484850

182 1 0.04583900 0.9541610

183 1 0.07686626 0.9231337

184 1 0.17605864 0.8239414

185 1 0.17447262 0.8255274

186 1 0.12642888 0.8735711

187 1 0.13414056 0.8658594

188 1 0.11641958 0.8835804

189 1 0.14984062 0.8501594

190 1 0.11186244 0.8881376

191 1 0.02672620 0.9732738

192 1 0.05697871 0.9430213

193 1 0.05693132 0.9430687

194 1 0.05613226 0.9438677

195 1 0.05586483 0.9441352

196 1 0.02777131 0.9722287

197 1 0.09762318 0.9023768

198 1 0.13278768 0.8672123

199 1 0.11301524 0.8869848

200 1 0.15176022 0.8482398

201 1 0.08437395 0.9156261

202 1 0.12020735 0.8797926

203 1 0.14921613 0.8507839

204 1 0.08365676 0.9163432

205 1 0.06057810 0.9394219

206 1 0.15657706 0.8434229

207 1 0.05951306 0.9404869

208 1 0.24116749 0.7588325

209 1 0.23958877 0.7604112

210 1 0.05683547 0.9431645

211 1 0.18406078 0.8159392

212 1 0.05565163 0.9443484

213 1 0.05910400 0.9408960

214 1 0.05632284 0.9436772

215 1 0.05647859 0.9435214

216 1 0.04928962 0.9507104

217 1 0.06216323 0.9378368

218 1 0.06466461 0.9353354

219 1 0.07476319 0.9252368

220 1 0.12886360 0.8711364

221 1 0.12541646 0.8745835

222 1 0.10187660 0.8981234

223 1 0.13278768 0.8672123

224 1 0.08089148 0.9191085

225 1 0.07754131 0.9224587

226 1 0.07618816 0.9238118

227 1 0.05815741 0.9418426

228 1 0.05725064 0.9427494

229 1 0.06816124 0.9318388

230 1 0.07228295 0.9277170

231 1 0.09368170 0.9063183

232 1 0.12886360 0.8711364

233 1 0.07675962 0.9232404

234 1 0.07475167 0.9252483

235 1 0.07394292 0.9260571

236 1 0.07249473 0.9275053

237 1 0.06654199 0.9334580

238 1 0.08101472 0.9189853

239 1 0.08431053 0.9156895

240 1 0.10478254 0.8952175

241 1 0.05192581 0.9480742

242 1 0.05322546 0.9467745

243 1 0.05226833 0.9477317

244 1 0.18229316 0.8177068

245 1 0.13177706 0.8682229

246 1 0.15960777 0.8403922

247 1 0.19315940 0.8068406

248 1 0.15780720 0.8421928

249 1 0.14998008 0.8500199

250 1 0.15376613 0.8462339

251 1 0.09721288 0.9027871

252 1 0.09458059 0.9054194

253 1 0.09251487 0.9074851

254 1 0.07480191 0.9251981

255 1 0.09369990 0.9063001

256 1 0.07214550 0.9278545

257 1 0.23324028 0.7667597

258 1 0.08224735 0.9177527

259 1 0.05648400 0.9435160

260 1 0.03689704 0.9631030

261 1 0.06159017 0.9384098

262 1 0.03049408 0.9695059

263 1 0.05897019 0.9410298

264 1 0.12036045 0.8796395

265 1 0.11139295 0.8886070

266 1 0.09986204 0.9001380

267 1 0.07870284 0.9212972

268 1 0.11046406 0.8895359

269 1 0.07306734 0.9269327

270 1 0.10070621 0.8992938

271 1 0.10356878 0.8964312

272 1 0.10680396 0.8931960

273 1 0.12408614 0.8759139

274 1 0.12460331 0.8753967

275 1 0.13042415 0.8695758

276 1 0.07093982 0.9290602

277 1 0.08047658 0.9195234

278 1 0.05556236 0.9444376

279 1 0.07969164 0.9203084

280 1 0.19375970 0.8062403

281 1 0.05507198 0.9449280

282 1 0.05947472 0.9405253

283 1 0.05935397 0.9406460

284 1 0.05933115 0.9406688

285 1 0.06490518 0.9350948

286 1 0.09544987 0.9045501

287 1 0.06091695 0.9390831

288 1 0.10187660 0.8981234

289 1 0.13278768 0.8672123

290 1 0.12886360 0.8711364

291 1 0.07675962 0.9232404

292 1 0.07493872 0.9250613

293 1 0.11459922 0.8854008

294 1 0.15541176 0.8445882

295 1 0.22266550 0.7773345

296 1 0.09277392 0.9072261

297 1 0.26338875 0.7366113

298 1 0.08281750 0.9171825

299 1 0.06325014 0.9367499

300 1 0.08809567 0.9119043

301 1 0.14468047 0.8553195

302 1 0.05877389 0.9412261

303 1 0.05774930 0.9422507

304 1 0.05304866 0.9469513

305 1 0.11415635 0.8858436

306 1 0.08146345 0.9185366

307 1 0.10282719 0.8971728

308 1 0.08412346 0.9158765

309 1 0.09278233 0.9072177

310 1 0.08405255 0.9159475

311 1 0.05302352 0.9469765

312 1 0.05487983 0.9451202

313 1 0.02685904 0.9731410

314 1 0.08884771 0.9111523

315 1 0.07402768 0.9259723

316 1 0.11164658 0.8883534

317 1 0.07530521 0.9246948

318 1 0.04972761 0.9502724

319 1 0.04928385 0.9507162

320 1 0.10390658 0.8960934

321 1 0.06208905 0.9379110

322 1 0.05261016 0.9473898

323 1 0.06771711 0.9322829

324 1 0.05808082 0.9419192

325 1 0.07266926 0.9273307

326 1 0.05737438 0.9426256

327 1 0.07372091 0.9262791

328 1 0.04200979 0.9579902

329 1 0.07120008 0.9287999

330 1 0.06187444 0.9381256

331 1 0.06333612 0.9366639

332 1 0.05621819 0.9437818

333 1 0.06688440 0.9331156

[ reached getOption("max.print") -- omitted 10657 rows ]

> h2o.shutdown()

Are you sure you want to shutdown the H2O instance running at http://localhost:54321/ (Y/N)?

[1] TRUE

> # examine the dl.result

> summary(dl.result)

predict p0 p1

0: 70 Min. :0.02643 Min. :0.1985

1:10920 1st Qu.:0.06515 1st Qu.:0.8628

Median :0.08709 Median :0.9129

Mean :0.10576 Mean :0.8942

3rd Qu.:0.13725 3rd Qu.:0.9349

Max. :0.80149 Max. :0.9736

> #### Explanation

> # p0 is the probability that 0 is chosen.

> # p1 is the probability that 1 is chosen.

> # predict: is made by applying a threshold to p1

>

> # List the important variables

> head(as.data.frame(h2o.varimp(dl.model)))

variable relative\_importance scaled\_importance percentage

1 B3 1.0000000 1.0000000 0.09187479

2 B1 0.7371128 0.7371128 0.06772208

3 B4 0.6744285 0.6744285 0.06196298

4 B2 0.6278723 0.6278723 0.05768563

5 A4 0.5712137 0.5712137 0.05248014

6 B6 0.5320271 0.5320271 0.04887988

>

> # Confusion Matrix

> # install.packages("gmodels")

> library(gmodels)

> CrossTable(splited.train2$Label, dl.result$predict,

+ prop.chisq = FALSE, prop.c = FALSE, prop.r = FALSE,

+ dnn = c('actual Labels', 'predicted Labels'))

Cell Contents

|-------------------------|

| N |

| N / Table Total |

|-------------------------|

Total Observations in Table: 10990

| predicted Labels

actual Labels | 0 | 1 | Row Total |

--------------|-----------|-----------|-----------|

0 | 38 | 1370 | 1408 |

| 0.003 | 0.125 | |

--------------|-----------|-----------|-----------|

1 | 32 | 9550 | 9582 |

| 0.003 | 0.869 | |

--------------|-----------|-----------|-----------|

Column Total | 70 | 10920 | 10990 |

--------------|-----------|-----------|-----------|

> # accuracy

> table.NN <- table(splited.train2$Label, dl.result$predict)

> nn.accuracy = round(sum(diag(table.NN)/sum(table.NN)),digits=5)

> nn.accuracy

[1] 0.87243

**Naïve Bay Result**

> nb.classifier <- naiveBayes(splited.train1, splited.train1$Label)

>

> nb.predict <- predict(nb.classifier, splited.train2)

> head(nb.predict)

[1] 1 1 1 1 1 1

Levels: 0 1

>

> # Confusion Matrix

> library(gmodels)

> CrossTable(nb.predict, splited.train2$Label,

+ prop.chisq = FALSE, prop.t = FALSE, prop.r = FALSE,

+ dnn = c('predicted', 'actual'))

Cell Contents

|-------------------------|

| N |

| N / Col Total |

|-------------------------|

Total Observations in Table: 10990

| actual

predicted | 0 | 1 | Row Total |

-------------|-----------|-----------|-----------|

0 | 1367 | 277 | 1644 |

| 0.971 | 0.029 | |

-------------|-----------|-----------|-----------|

1 | 41 | 9305 | 9346 |

| 0.029 | 0.971 | |

-------------|-----------|-----------|-----------|

Column Total | 1408 | 9582 | 10990 |

| 0.128 | 0.872 | |

-------------|-----------|-----------|-----------|

>

> # accuracy

> table.NB <- table(splited.train2$Label, nb.predict)

> nb.accuracy = round(sum(diag(table.NB)/sum(table.NB)),digits=5)

> nb.accuracy

[1] 0.97106

**Decision Tree Result**

> dt.classifier <- C5.0(splited.train1[-1], splited.train1$Label)

>

> # generate predictions for the testing dataset

> dt.predict <- predict(dt.classifier, splited.train2)

>

> # cross tabulation of predicted versus actual classes

> library(gmodels)

> CrossTable(splited.train2$Label, dt.predict,

+ prop.chisq = FALSE, prop.c = FALSE, prop.r = FALSE,

+ dnn = c('actual', 'predicted'))

Cell Contents

|-------------------------|

| N |

| N / Table Total |

|-------------------------|

Total Observations in Table: 10990

| predicted

actual | 0 | 1 | Row Total |

-------------|-----------|-----------|-----------|

0 | 466 | 942 | 1408 |

| 0.042 | 0.086 | |

-------------|-----------|-----------|-----------|

1 | 111 | 9471 | 9582 |

| 0.010 | 0.862 | |

-------------|-----------|-----------|-----------|

Column Total | 577 | 10413 | 10990 |

-------------|-----------|-----------|-----------|

>

> # accuracy

> table.DT <- table(splited.train2$Label, dt.predict)

> dt.accuracy = round(sum(diag(table.DT)/sum(table.DT)),digits=5)

> dt.accuracy

[1] 0.90419

**SVM Result**

> svm.classifier <- svm(Label~.,data=splited.train1, scale=FALSE)

>

> # generate predictions for the testing dataset

> svm.predict <- predict(svm.classifier, splited.train2)

> summary(svm.predict)

0 1

0 10990

>

> # cross tabulation of predicted versus actual classes

> library(gmodels)

> CrossTable(splited.train2$Label, svm.predict,

+ prop.chisq = FALSE, prop.c = FALSE, prop.r = FALSE,

+ dnn = c('actual', 'predicted'))

Cell Contents

|-------------------------|

| N |

| N / Table Total |

|-------------------------|

Total Observations in Table: 10990

| svm.predict

splited.train2$Label | 1 | Row Total |

---------------------|-----------|-----------|

0 | 1408 | 1408 |

| 0.128 | |

---------------------|-----------|-----------|

1 | 9582 | 9582 |

| 0.872 | |

---------------------|-----------|-----------|

Column Total | 10990 | 10990 |

---------------------|-----------|-----------|

>

> # accuracy

> table.svm <- table(splited.train2$Label, svm.predict)

> svm.accuracy = round(sum(diag(table.svm)/sum(table.svm)),digits=5)

> svm.accuracy

[1] 0.87188

**Accuracy Comparison Table**

> com.table <- matrix(c('Neural Network', 'Naive Bayes', 'Decision Tree', 'SVM',

+ nn.accuracy, nb.accuracy, dt.accuracy, svm.accuracy), ncol=4, byrow=TRUE)

> com.table

[,1] [,2] [,3] [,4]

[1,] "Neural Network" "Naive Bayes" "Decision Tree" "SVM"

[2,] "0.87243" "0.97106" "0.90419" "0.87188"

**Apply on Testing Set**

**Naïve Bayes**

|  |
| --- |
| > library(e1071)  > nb.classifier.data <- naiveBayes(Label~., data=scaled.train)  > nb.predict.data <- predict(nb.classifier.data, scaled.test, type="class")  >  > # Percentage of good quality prediction  > length(which(nb.predict.data=="1"))\*100/length(nb.predict.data)  [1] 8.123043 |
|  |
| |  | | --- | | > | |

**Decision Tree**

> library(C50)

> dt.classifier.data <- C5.0(scaled.train[-1], scaled.train$Label)

>

> # generate predictions for the testing dataset

> dt.predict.data <- predict(dt.classifier.data, scaled.test)

> summary(dt.predict.data)

0 1

6800 36632

> # Percentage of good quality prediction

> length(which(dt.predict.data=="1"))\*100/length(dt.predict.data)

[1] 84.34334

**SVM**

> library(e1071)

> svm.classifier.data <- svm(Label~.,data=scaled.train, scale=FALSE)

>

> # generate predictions for the testing dataset

> svm.predict.data <- predict(svm.classifier.data, scaled.test)

> summary(svm.predict.data)

0 1

0 43432

>

> # Percentage of good quality prediction

> length(which(svm.predict.data=="1"))\*100/length(svm.predict.data)

[1] 100

**Deep Learning H2O**

h2o.init(nthreads=8, max\_mem\_size="2G")

Connection successful!

R is connected to the H2O cluster:

H2O cluster uptime: 2 minutes 26 seconds

H2O cluster timezone: America/Los\_Angeles

H2O data parsing timezone: UTC

H2O cluster version: 3.20.0.8

H2O cluster version age: 1 month and 28 days

H2O cluster name: H2O\_started\_from\_R\_SEAN\_\_PHAN\_ktg449

H2O cluster total nodes: 1

H2O cluster total memory: 1.61 GB

H2O cluster total cores: 4

H2O cluster allowed cores: 4

H2O cluster healthy: TRUE

H2O Connection ip: localhost

H2O Connection port: 54321

H2O Connection proxy: NA

H2O Internal Security: FALSE

H2O API Extensions: Algos, AutoML, Core V3, Core V4

R Version: R version 3.4.4 (2018-03-15)

> h2o.removeAll() ## clean slate - just in case the cluster was already running

[1] 0

> h2o.init()

Connection successful!

R is connected to the H2O cluster:

H2O cluster uptime: 2 minutes 26 seconds

H2O cluster timezone: America/Los\_Angeles

H2O data parsing timezone: UTC

H2O cluster version: 3.20.0.8

H2O cluster version age: 1 month and 28 days

H2O cluster name: H2O\_started\_from\_R\_SEAN\_\_PHAN\_ktg449

H2O cluster total nodes: 1

H2O cluster total memory: 1.61 GB

H2O cluster total cores: 4

H2O cluster allowed cores: 4

H2O cluster healthy: TRUE

H2O Connection ip: localhost

H2O Connection port: 54321

H2O Connection proxy: NA

H2O Internal Security: FALSE

H2O API Extensions: Algos, AutoML, Core V3, Core V4

R Version: R version 3.4.4 (2018-03-15)

>

> # split train data for validation

> train.hex <- as.h2o(scaled.train)

|======================================================================| 100%

> test.hex <- as.h2o(scaled.test)

|======================================================================| 100%

>

> splits <- h2o.splitFrame(train.hex, 0.8, seed=777)

> split.train <- h2o.assign(splits[[1]], "train.hex") # 80%

> split.valid <- h2o.assign(splits[[2]], "valid.hex") # 20%

>

> dl.model <- h2o.deeplearning(x=2:26,

+ y="Label",

+ training\_frame=split.train,

+ validation\_frame=split.valid,

+ activation = "Tanh",

+ hidden = c(200,200),

+ variable\_importances=T)

|======================================================================| 100%

> summary(dl.model)

Model Details:

==============

H2OBinomialModel: deeplearning

Model Key: DeepLearning\_model\_R\_1542604542620\_1

Status of Neuron Layers: predicting Label, 2-class classification, bernoulli distribution, CrossEntropy loss, 45,802 weights/biases, 548.3 KB, 362,294 training samples, mini-batch size 1

layer units type dropout l1 l2 mean\_rate rate\_rms momentum

1 1 25 Input 0.00 % NA NA NA NA NA

2 2 200 Tanh 0.00 % 0.000000 0.000000 0.112851 0.164965 0.000000

3 3 200 Tanh 0.00 % 0.000000 0.000000 0.318020 0.195484 0.000000

4 4 2 Softmax NA 0.000000 0.000000 0.002114 0.000509 0.000000

mean\_weight weight\_rms mean\_bias bias\_rms

1 NA NA NA NA

2 0.011221 0.262688 0.018926 0.205040

3 0.000100 0.100394 -0.001711 0.384016

4 0.006518 0.253105 0.001438 0.929790

H2OBinomialMetrics: deeplearning

\*\* Reported on training data. \*\*

\*\* Metrics reported on temporary training frame with 9826 samples \*\*

MSE: 0.1045308

RMSE: 0.3233122

LogLoss: 0.3572419

Mean Per-Class Error: 0.4824882

AUC: 0.6703961

Gini: 0.3407922

Confusion Matrix (vertical: actual; across: predicted) for F1-optimal threshold:

0 1 Error Rate

0 46 1201 0.963111 =1201/1247

1 16 8563 0.001865 =16/8579

Totals 62 9764 0.123855 =1217/9826

Maximum Metrics: Maximum metrics at their respective thresholds

metric threshold value idx

1 max f1 0.646436 0.933653 374

2 max f2 0.277943 0.971772 398

3 max f0point5 0.780677 0.902506 302

4 max accuracy 0.646436 0.876145 374

5 max precision 0.983467 1.000000 0

6 max recall 0.277943 1.000000 398

7 max specificity 0.983467 1.000000 0

8 max absolute\_mcc 0.819503 0.217490 259

9 max min\_per\_class\_accuracy 0.883619 0.623095 162

10 max mean\_per\_class\_accuracy 0.857028 0.637805 206

Gains/Lift Table: Extract with `h2o.gainsLift(<model>, <data>)` or `h2o.gainsLift(<model>, valid=<T/F>, xval=<T/F>)`

H2OBinomialMetrics: deeplearning

\*\* Reported on validation data. \*\*

\*\* Metrics reported on full validation frame \*\*

MSE: 0.1102033

RMSE: 0.3319689

LogLoss: 0.3725337

Mean Per-Class Error: 0.4844986

AUC: 0.6623125

Gini: 0.324625

Confusion Matrix (vertical: actual; across: predicted) for F1-optimal threshold:

0 1 Error Rate

0 39 1129 0.966610 =1129/1168

1 18 7521 0.002388 =18/7539

Totals 57 8650 0.131733 =1147/8707

Maximum Metrics: Maximum metrics at their respective thresholds

metric threshold value idx

1 max f1 0.648520 0.929149 379

2 max f2 0.277943 0.969996 398

3 max f0point5 0.780767 0.896696 311

4 max accuracy 0.648520 0.868267 379

5 max precision 0.982736 1.000000 0

6 max recall 0.277943 1.000000 398

7 max specificity 0.982736 1.000000 0

8 max absolute\_mcc 0.820330 0.205895 266

9 max min\_per\_class\_accuracy 0.885173 0.616438 161

10 max mean\_per\_class\_accuracy 0.871369 0.626111 191

Gains/Lift Table: Extract with `h2o.gainsLift(<model>, <data>)` or `h2o.gainsLift(<model>, valid=<T/F>, xval=<T/F>)`

Scoring History:

timestamp duration training\_speed epochs iterations

1 2018-11-18 21:18:15 0.000 sec NA 0.00000 0

2 2018-11-18 21:18:25 11.300 sec 2292 obs/sec 0.64434 1

3 2018-11-18 21:18:36 22.395 sec 2282 obs/sec 1.28922 2

4 2018-11-18 21:18:47 33.769 sec 2239 obs/sec 1.93155 3

5 2018-11-18 21:18:59 45.215 sec 2214 obs/sec 2.57510 4

6 2018-11-18 21:19:10 56.765 sec 2197 obs/sec 3.22021 5

7 2018-11-18 21:19:22 1 min 8.144 sec 2186 obs/sec 3.85902 6

8 2018-11-18 21:19:33 1 min 19.517 sec 2185 obs/sec 4.50262 7

9 2018-11-18 21:19:45 1 min 30.829 sec 2183 obs/sec 5.14416 8

10 2018-11-18 21:19:56 1 min 42.112 sec 2182 obs/sec 5.78739 9

11 2018-11-18 21:20:07 1 min 53.212 sec 2183 obs/sec 6.42578 10

12 2018-11-18 21:20:18 2 min 4.419 sec 2183 obs/sec 7.06938 11

13 2018-11-18 21:20:30 2 min 17.195 sec 2171 obs/sec 7.70936 12

14 2018-11-18 21:20:53 2 min 38.876 sec 2173 obs/sec 8.99362 14

15 2018-11-18 21:21:04 2 min 50.195 sec 2173 obs/sec 9.63546 15

16 2018-11-18 21:21:15 3 min 1.365 sec 2174 obs/sec 10.27697 16

17 2018-11-18 21:21:16 3 min 2.202 sec 2174 obs/sec 10.27697 16

samples training\_rmse training\_logloss training\_r2 training\_auc

1 0.000000 NA NA NA NA

2 22715.000000 0.33474 0.38511 -0.01129 0.59837

3 45449.000000 0.33157 0.38219 0.00777 0.60987

4 68093.000000 0.32856 0.37234 0.02571 0.63294

5 90780.000000 0.33415 0.38367 -0.00768 0.62613

6 113522.000000 0.33031 0.37069 0.01530 0.64509

7 136042.000000 0.32895 0.37001 0.02339 0.63362

8 158731.000000 0.32671 0.36365 0.03665 0.65565

9 181347.000000 0.32772 0.36628 0.03072 0.65450

10 204023.000000 0.33205 0.38411 0.00491 0.61891

11 226528.000000 0.32750 0.36779 0.03199 0.65097

12 249217.000000 0.32548 0.36225 0.04390 0.65709

13 271778.000000 0.33782 0.38686 -0.02995 0.65086

14 317052.000000 0.32667 0.36478 0.03690 0.64735

15 339679.000000 0.32331 0.35724 0.05660 0.67040

16 362294.000000 0.32981 0.38218 0.01830 0.65741

17 362294.000000 0.32331 0.35724 0.05660 0.67040

training\_lift training\_classification\_error validation\_rmse

1 NA NA NA

2 1.07663 0.12528 0.34047

3 1.08751 0.12660 0.33987

4 1.07663 0.12599 0.33610

5 1.11099 0.12620 0.34207

6 1.11232 0.12630 0.33763

7 1.09908 0.12660 0.33492

8 1.07594 0.12528 0.33527

9 1.05891 0.12660 0.33507

10 1.12245 0.12620 0.34064

11 1.05280 0.12609 0.33598

12 1.09908 0.12559 0.33344

13 1.12267 0.12528 0.34362

14 1.14535 0.12487 0.33464

15 1.10088 0.12386 0.33197

16 1.11099 0.12508 0.33941

17 1.10088 0.12386 0.33197

validation\_logloss validation\_r2 validation\_auc validation\_lift

1 NA NA NA NA

2 0.39551 0.00198 0.61379 0.96244

3 0.39815 0.00547 0.61036 1.10243

4 0.38685 0.02745 0.63608 0.93182

5 0.39838 -0.00744 0.63904 1.04993

6 0.38391 0.01857 0.63791 1.10243

7 0.37918 0.03425 0.64749 1.11556

8 0.38077 0.03225 0.63290 1.05227

9 0.37947 0.03340 0.64628 1.06306

10 0.40093 0.00100 0.61687 1.07618

11 0.38380 0.02813 0.64899 1.09004

12 0.37667 0.04279 0.64946 1.11643

13 0.39663 -0.01657 0.64097 1.12868

14 0.37831 0.03586 0.65056 1.09147

15 0.37253 0.05120 0.66231 1.11556

16 0.40233 0.00817 0.65741 1.09004

17 0.37253 0.05120 0.66231 1.11556

validation\_classification\_error

1 NA

2 0.13346

3 0.13414

4 0.13357

5 0.13357

6 0.13392

7 0.13323

8 0.13265

9 0.13254

10 0.13357

11 0.13357

12 0.13300

13 0.13277

14 0.13185

15 0.13173

16 0.13357

17 0.13173

Variable Importances: (Extract with `h2o.varimp`)

=================================================

Variable Importances:

variable relative\_importance scaled\_importance percentage

1 B3 1.000000 1.000000 0.099652

2 B2 0.641025 0.641025 0.063879

3 B1 0.563344 0.563344 0.056138

4 B4 0.548830 0.548830 0.054692

5 Param1 0.512461 0.512461 0.051068

---

variable relative\_importance scaled\_importance percentage

20 B5 0.293561 0.293561 0.029254

21 s620 0.289340 0.289340 0.028833

22 BN1 0.279933 0.279933 0.027896

23 A1 0.259773 0.259773 0.025887

24 Param3 0.226018 0.226018 0.022523

25 Mix50.50 0.220440 0.220440 0.021967

> plot(dl.model)

>

> dl.model.predict.data <- h2o.predict(dl.model, test.hex)

|======================================================================| 100%

> dl.result.data <- as.data.frame(dl.model.predict.data)

> h2o.shutdown()

Are you sure you want to shutdown the H2O instance running at http://localhost:54321/ (Y/N)? y

> # examine the dl.result

> summary(dl.result.data)

predict p0 p1

0: 1400 Min. :0.02261 Min. :0.06193

1:42032 1st Qu.:0.04602 1st Qu.:0.89629

Median :0.07366 Median :0.92634

Mean :0.10304 Mean :0.89696

3rd Qu.:0.10371 3rd Qu.:0.95398

Max. :0.93807 Max. :0.97739

> # Percentage of good quality prediction

> length(which(dl.result.data=="1"))\*100/length(dt.predict.data)

[1] 96.77657