Boosting under high noise.

Adaboost is sensitive to label noise

- Letter / Irvine Database
- Focus on a binary problem: {F,I,J} vs. other letters.

Label Noise	Adaboost	Logitboost
0%	0.8% ±0.2%	0.8% ±0.1%
20%	33.3% ±0.7%	31.6% ±0.6%

- Boosting puts too much weight on outliers.
- Need to give up on outliers.

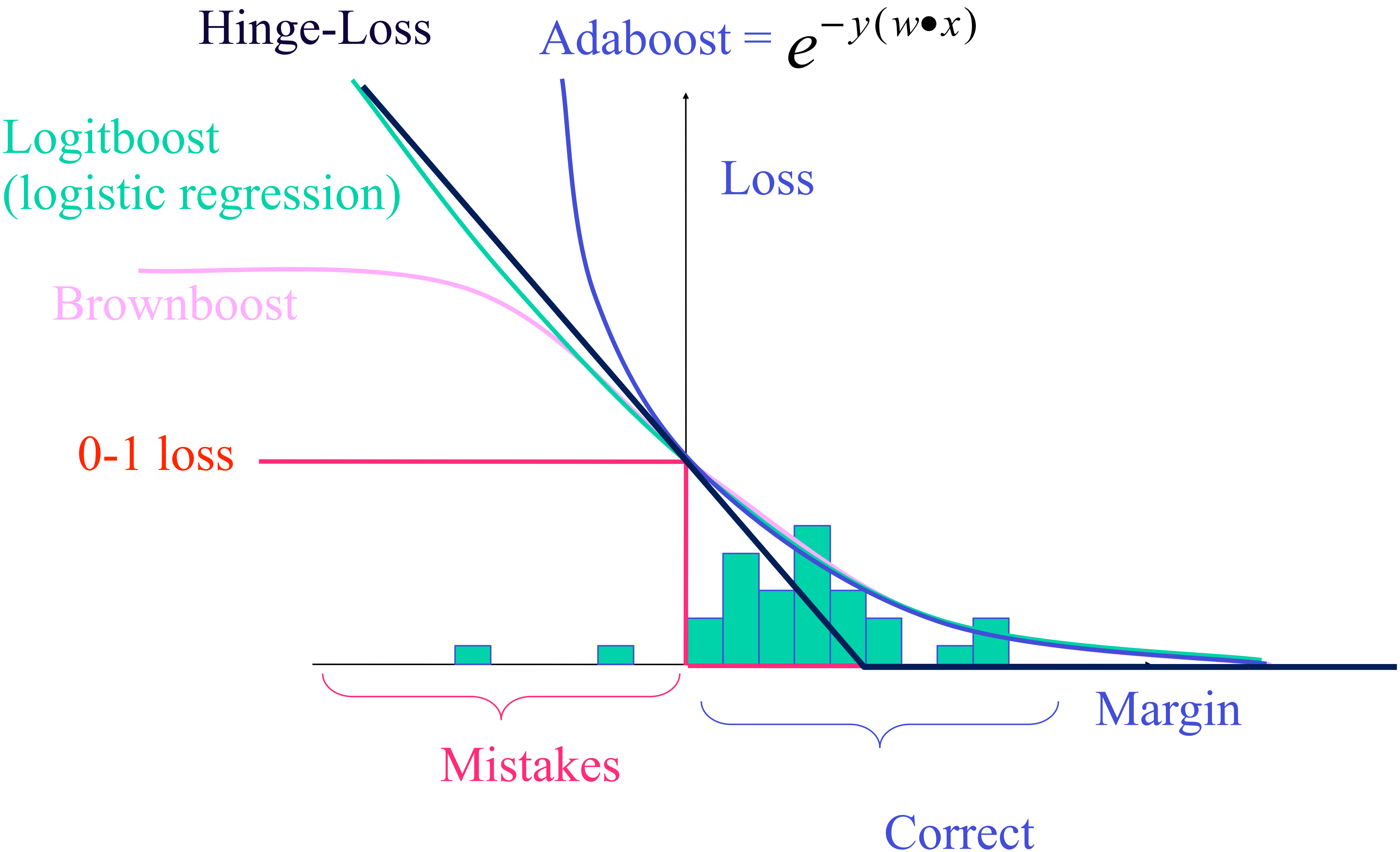
Robustboost - A new boosting algorithm

Label Noise	Adaboost	Logitboost	Robustboost
0%	0.8% ±0.2%	0.8% ±0.1%	2.9% ±0.2%
20%	33.3% ±0.7%	31.6% ±0.6%	22.2 ±0.8%

error with respect to original (noiseless) labels

20%	22.1% ±1.2%	19.4% ±1.3%	**3.7% ±0.4%**

Approximating mistake loss with convex functions

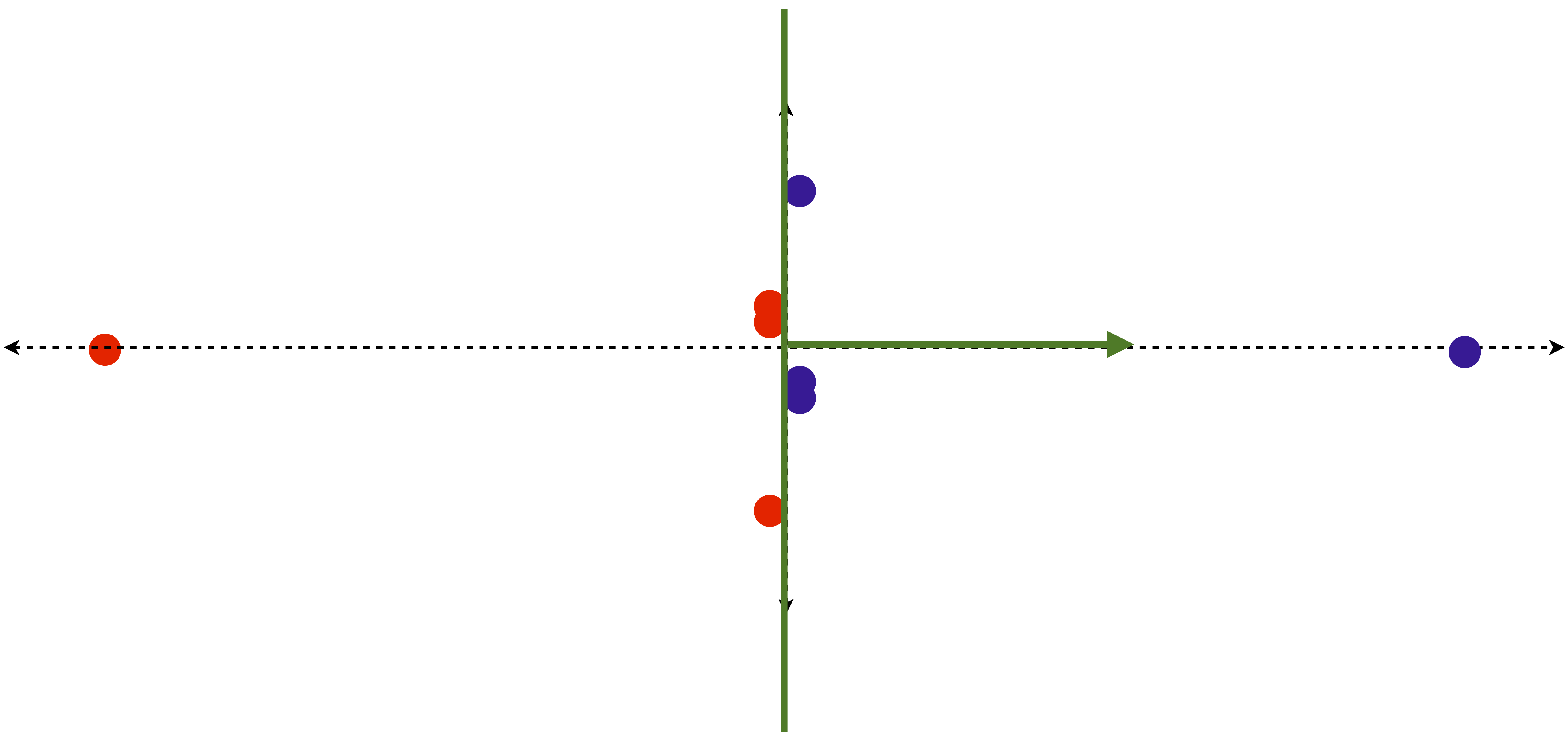


Label noise and convex loss functions

- Algorithms for learning a classifier based on minimizing a convex loss function: perceptron, Adaboost, Logitboost, Logistic regression, soft margins SVM.
- Work well when data is linearly separable.
- Can get into trouble when not linearly separable.
- Problem: Convex loss functions are a poor approximation for classification error.
- But: No known efficient algorithms for minimizing a non-convex loss function.

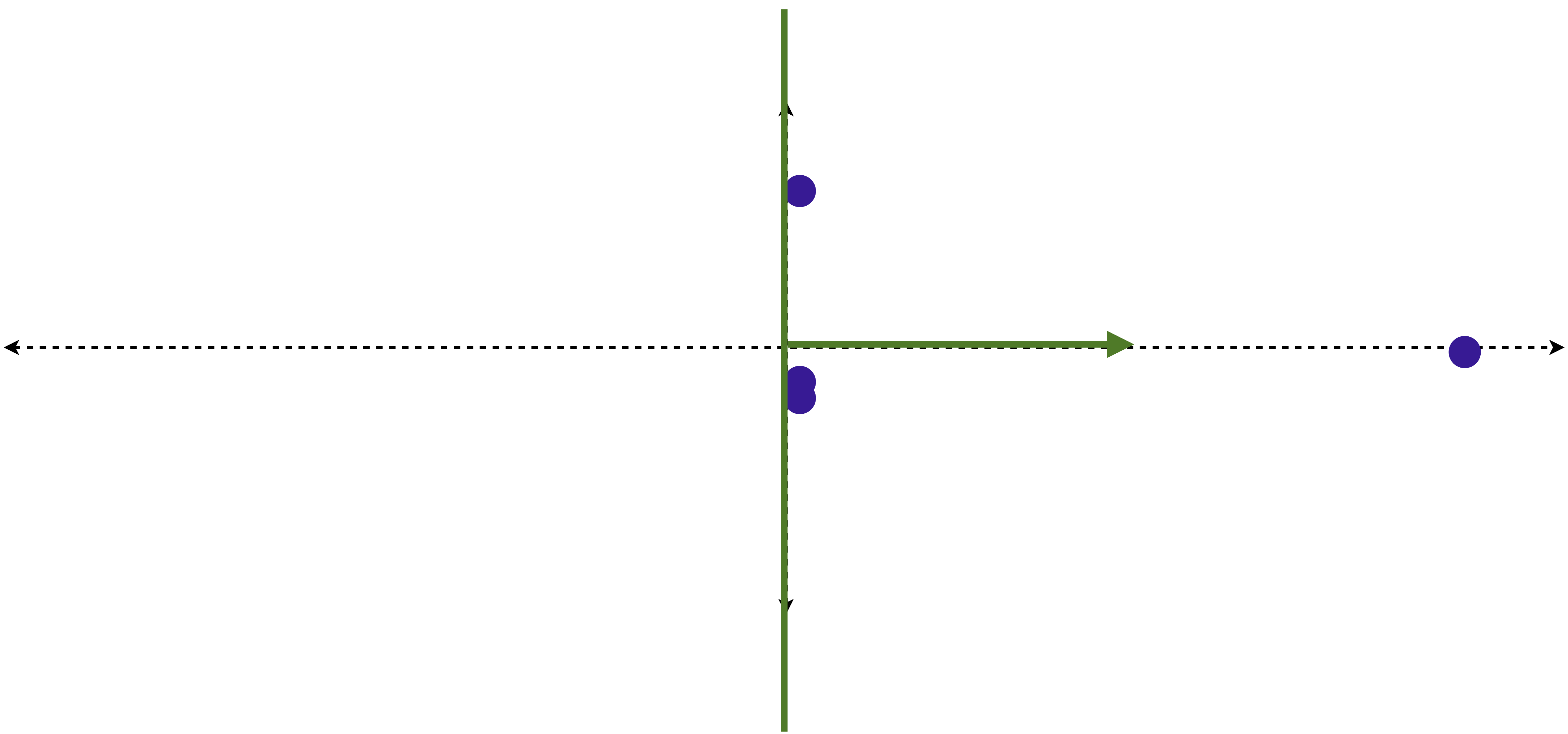
Random label noise defeats any convex loss function

[Servedio, Long 2010]



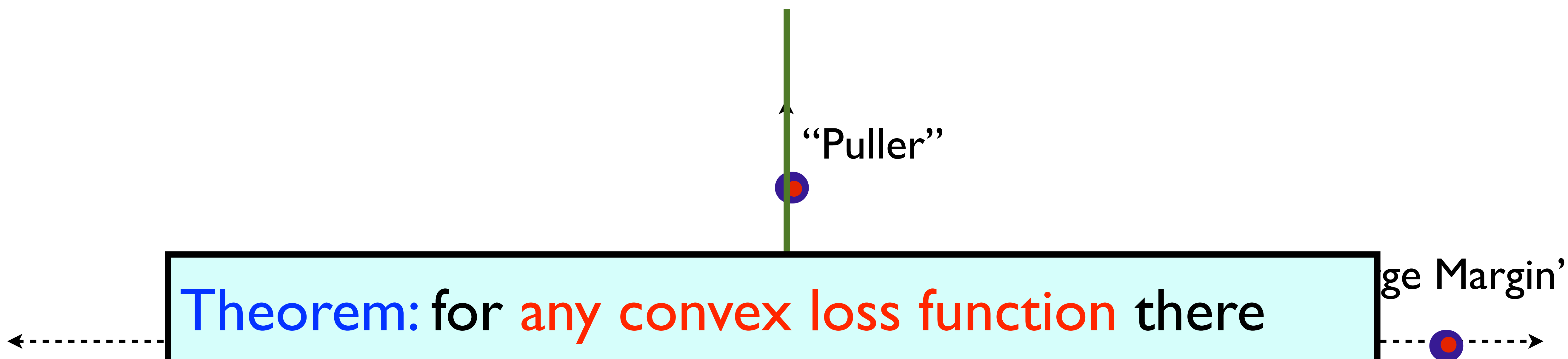
Considering one symmetric half

[Servedio, Long 2010]



Adding random label noise

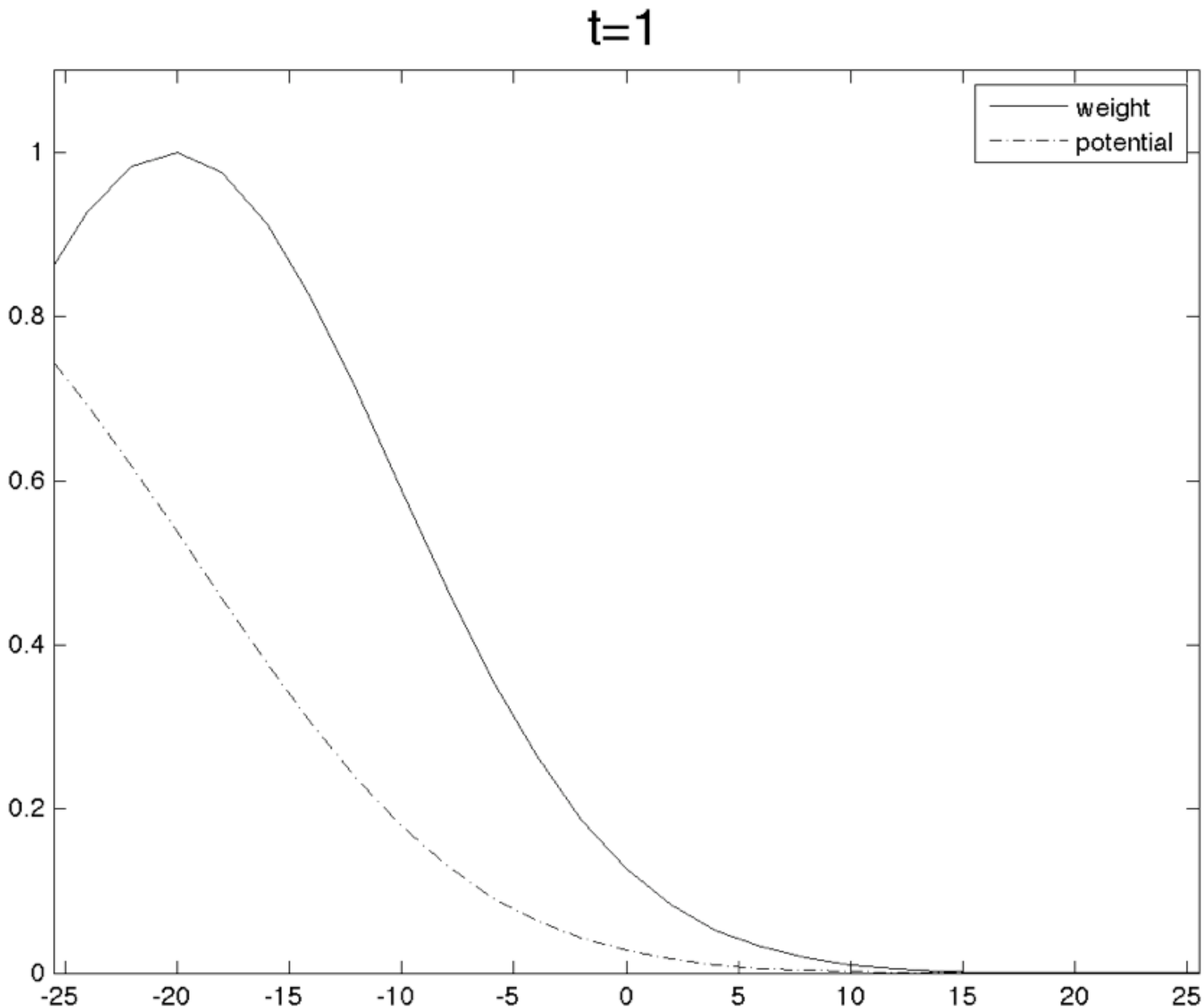
[Servedio, Long 2010]

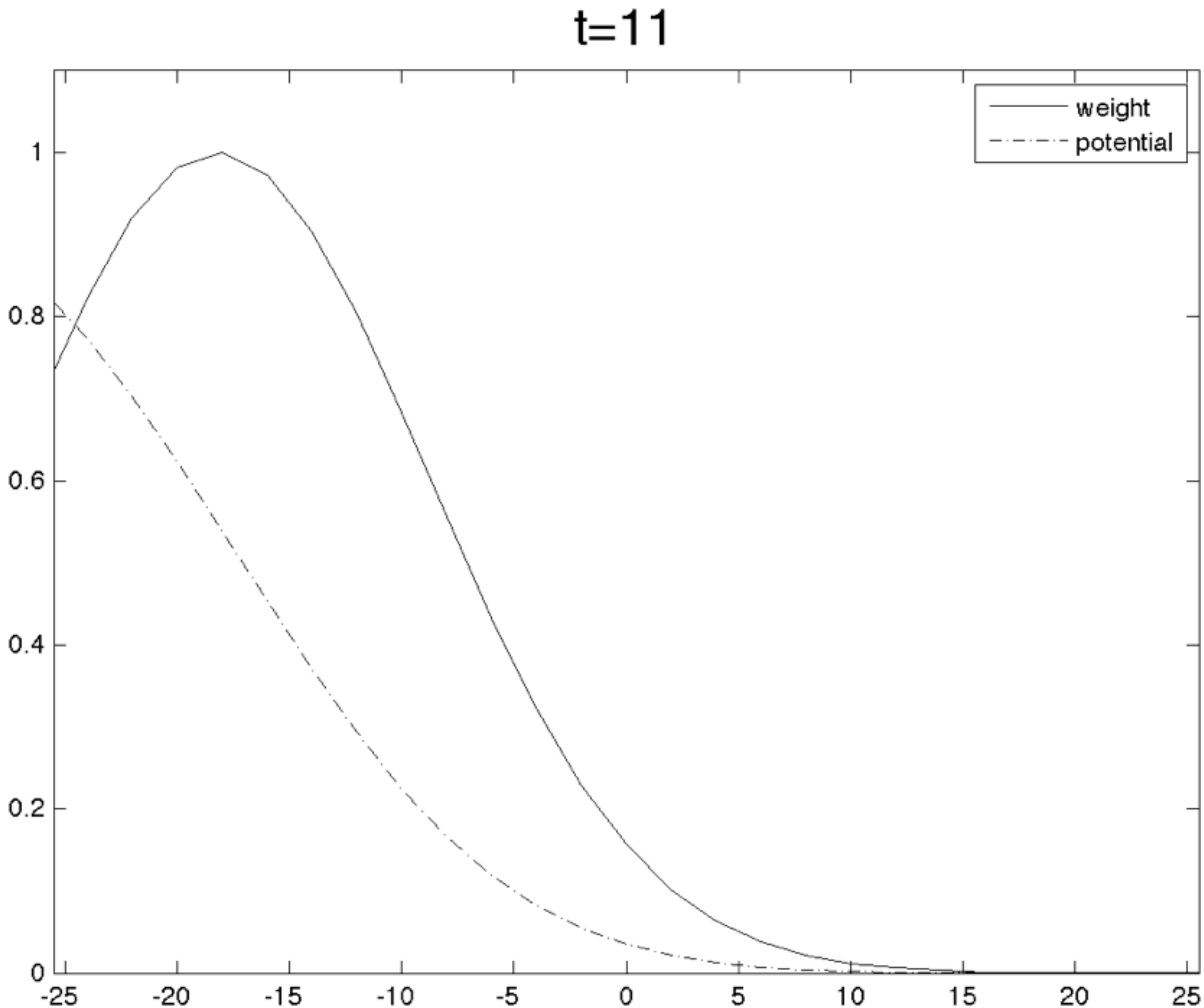


Theorem: for any convex loss function there exists a linearly separable distribution such that when independent label noise is added, the linear classifier that minimizes the loss function has very high classification error.

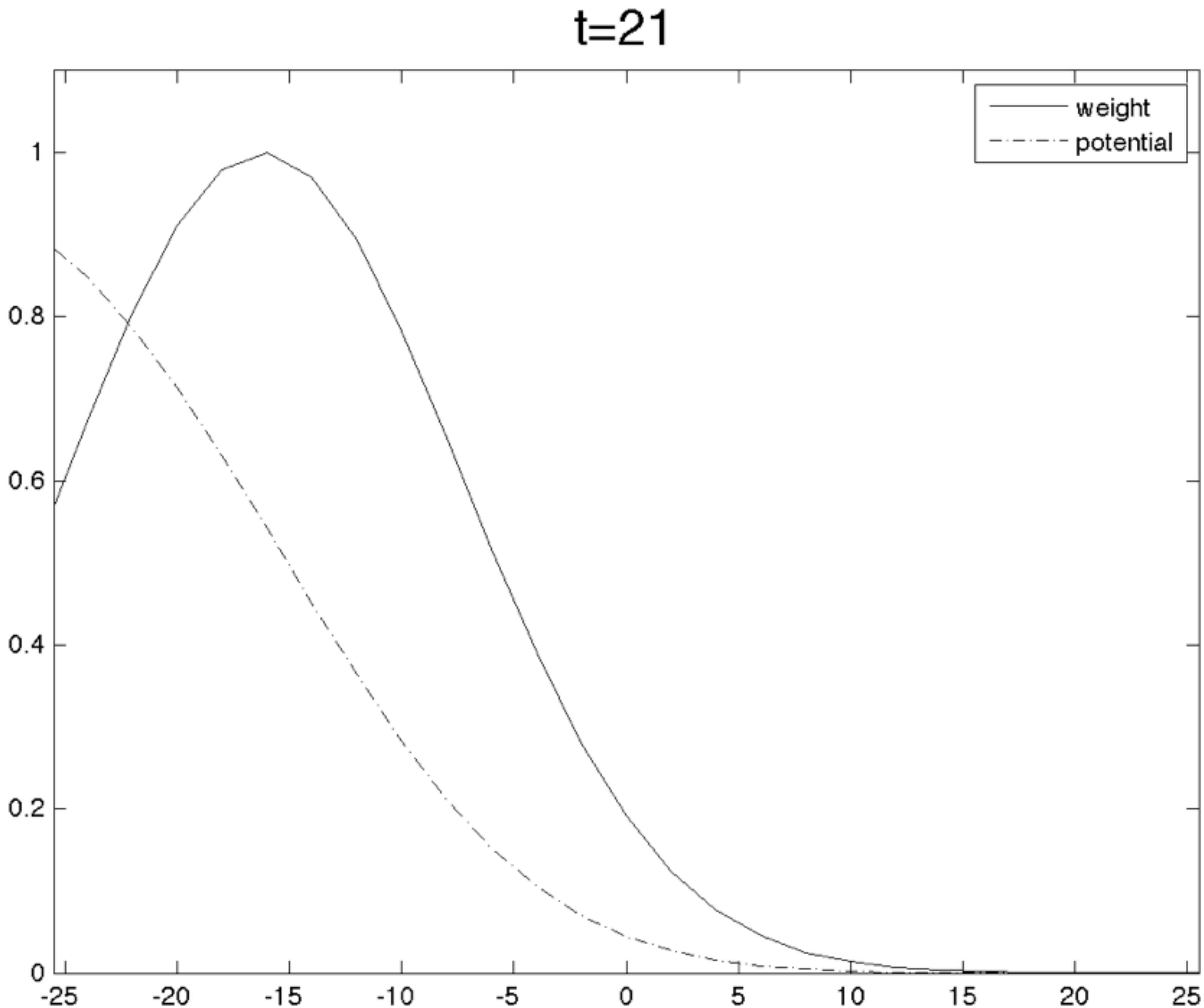
Boost by majority, Brownboost,

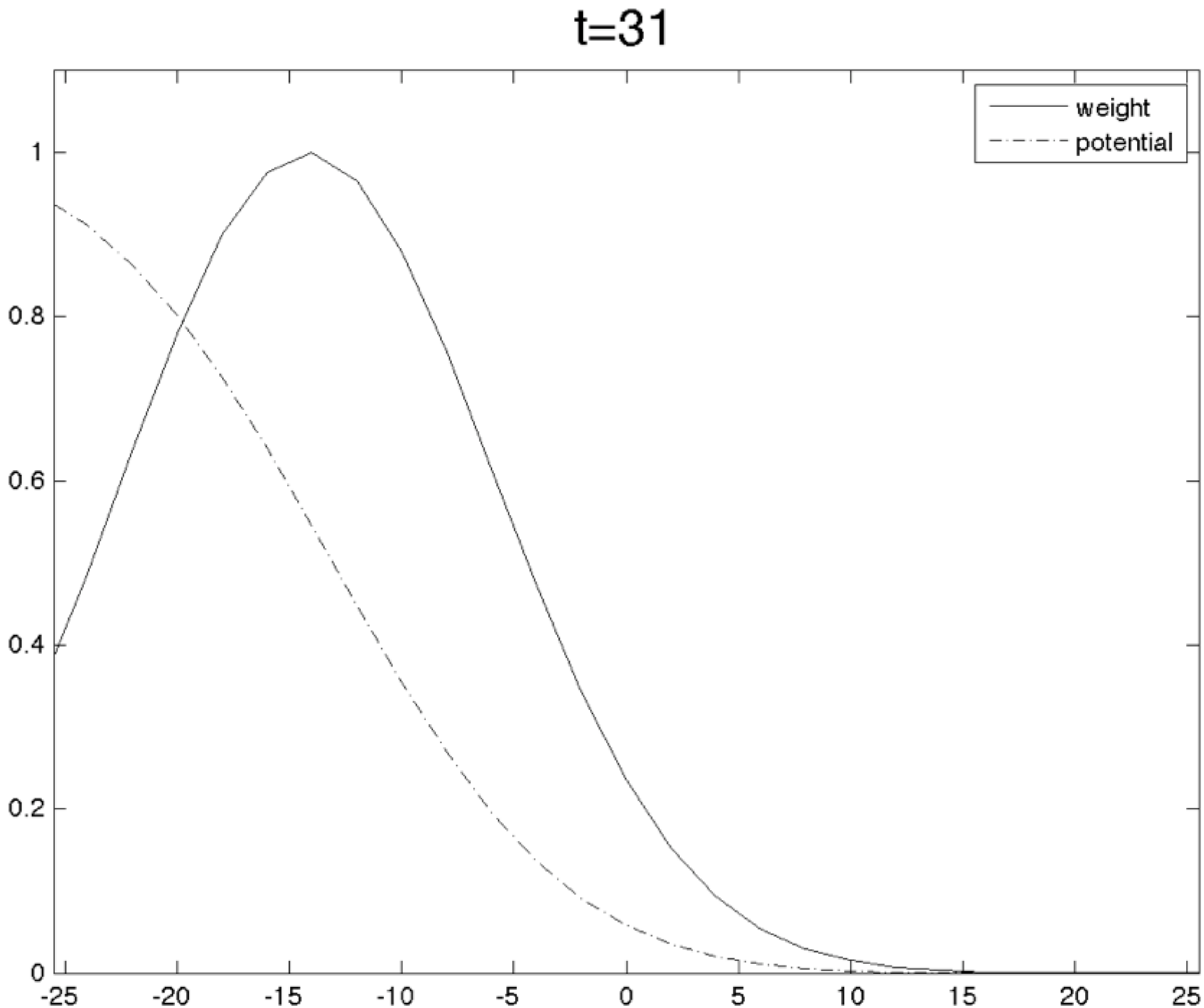
- Target error set at start.
- Defines how many boosting iterations are needed
- The loss function depends on the time-to-finish.
- Close to end - give up on examples with large negative margins.

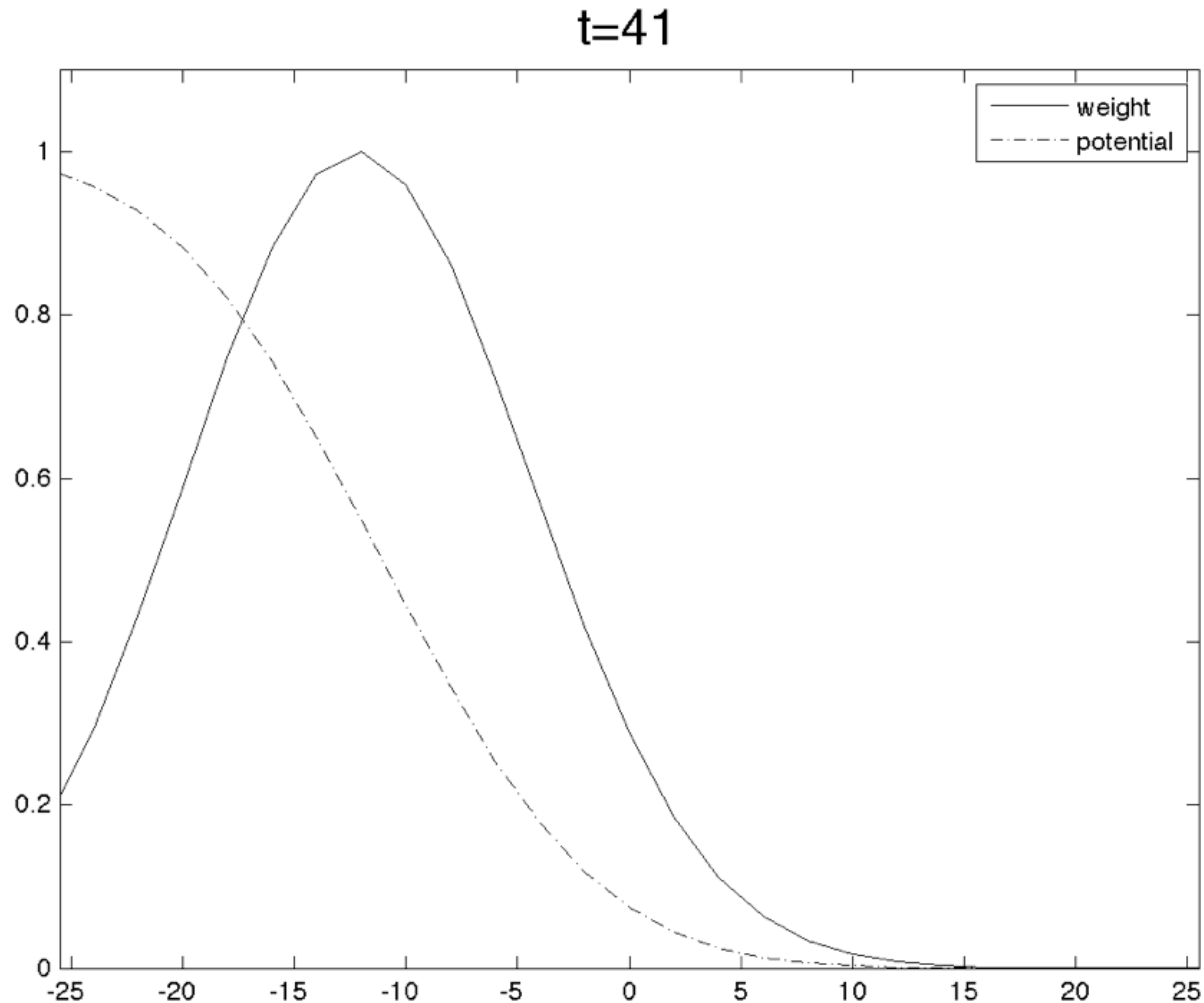

t=1
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25



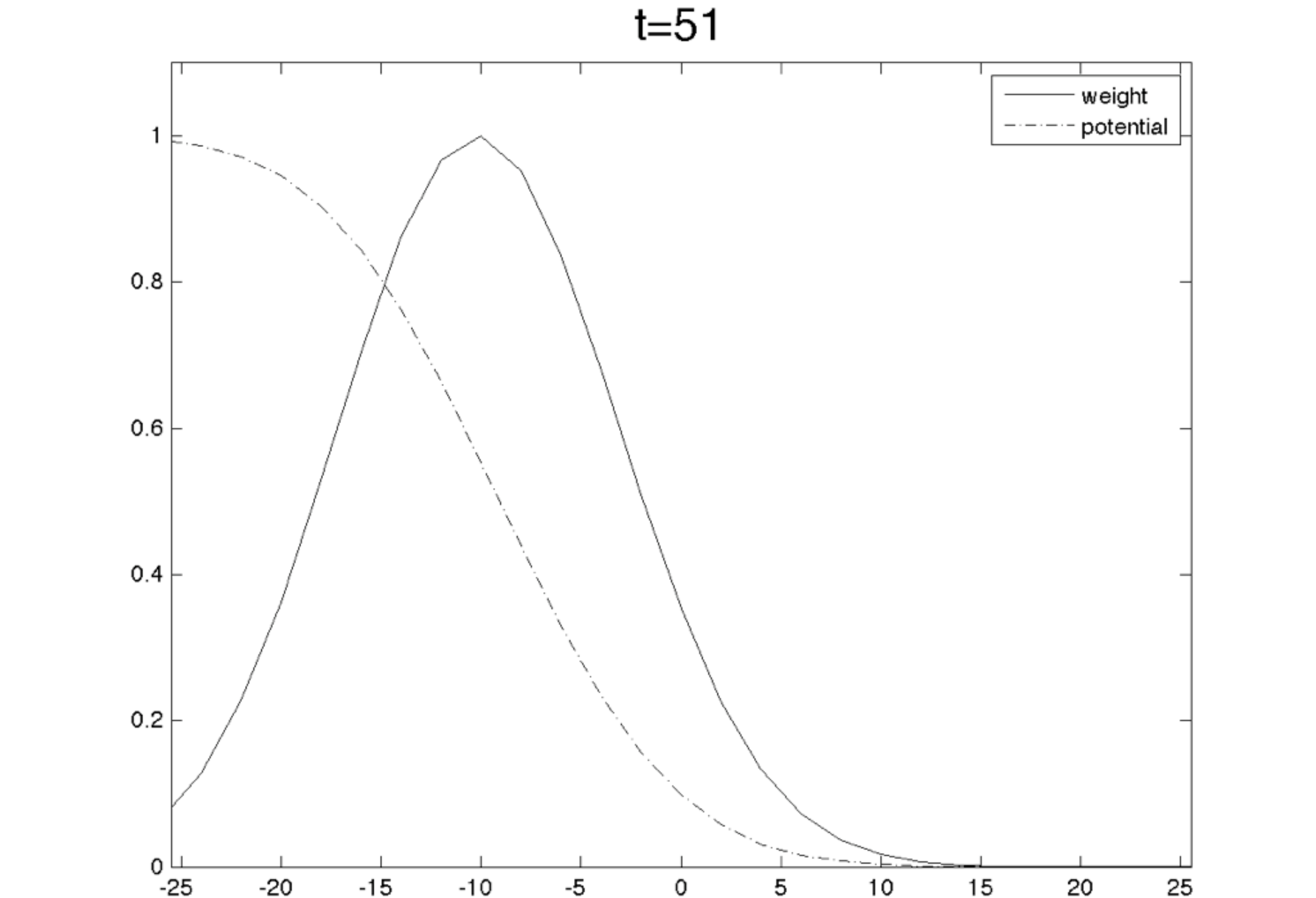
t=11
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25


t=21
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25

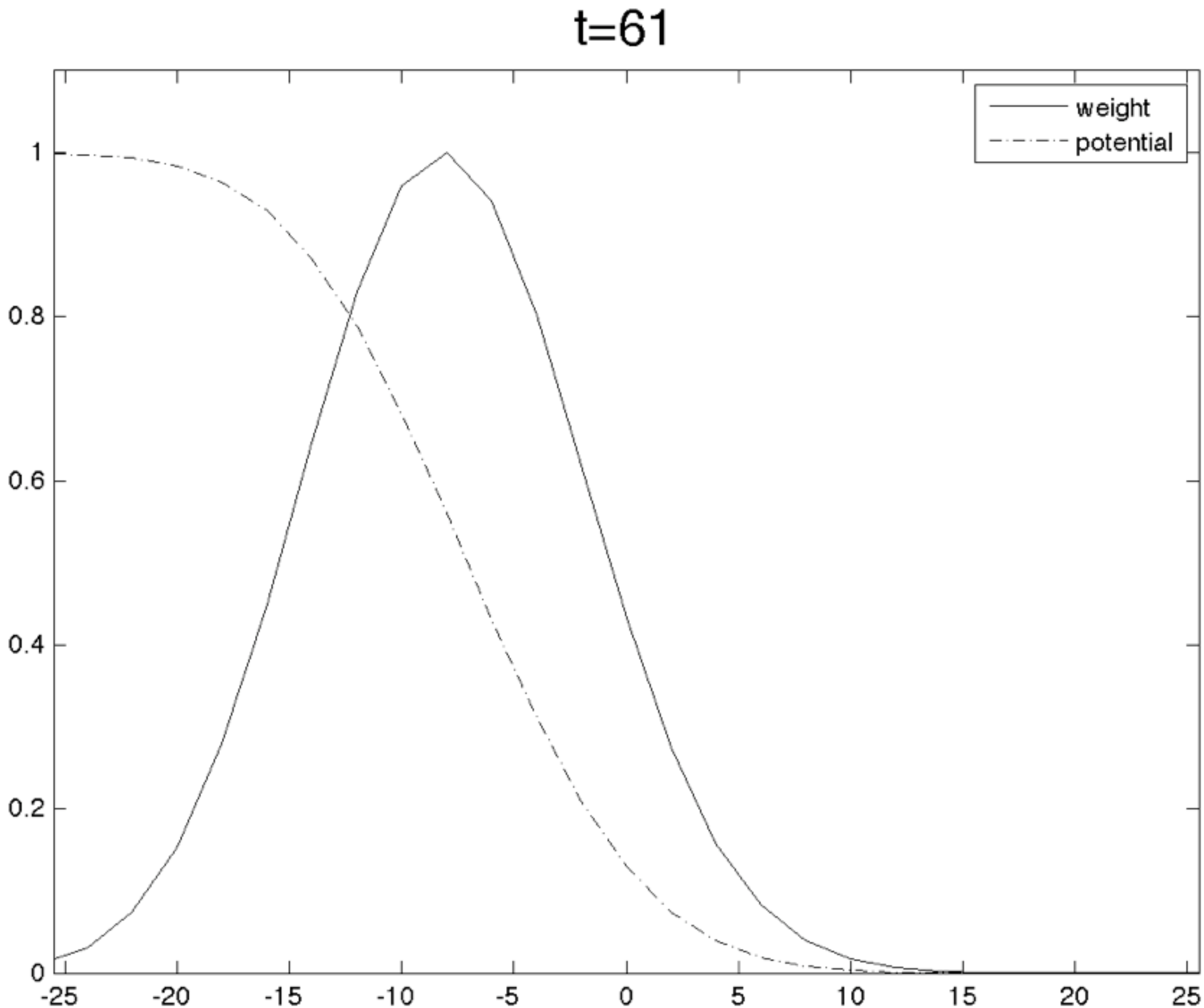

t=31
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25

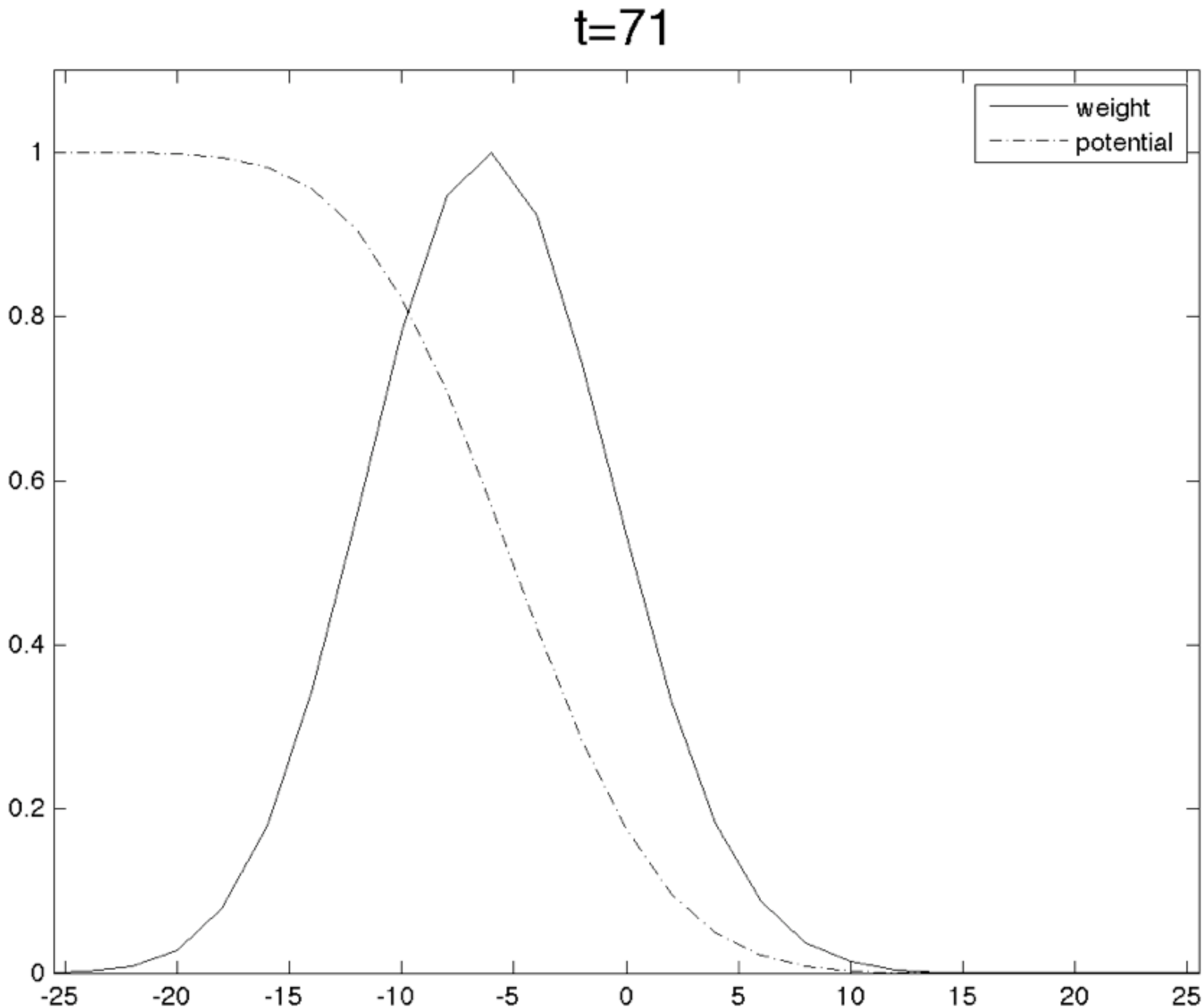


t=41
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25

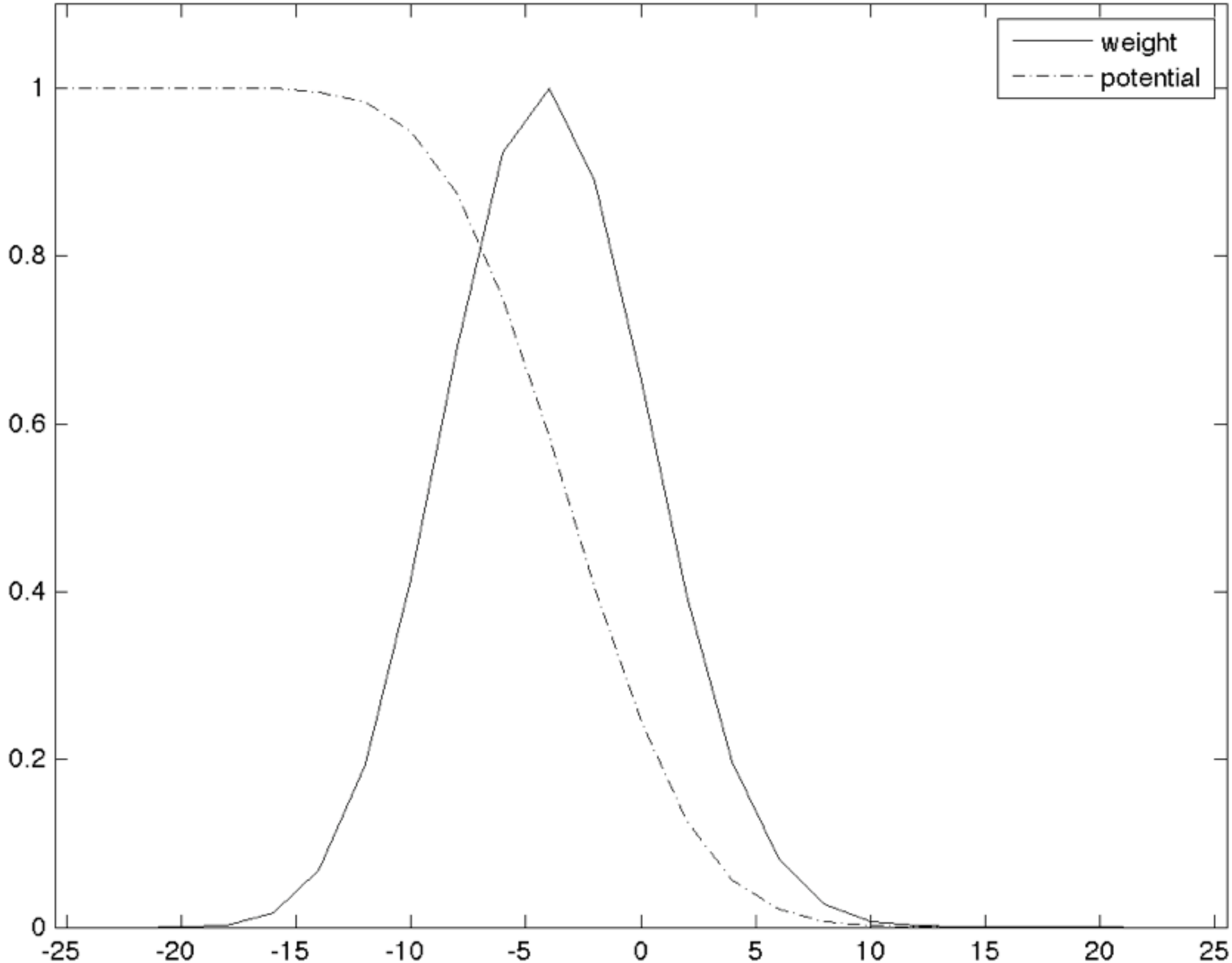


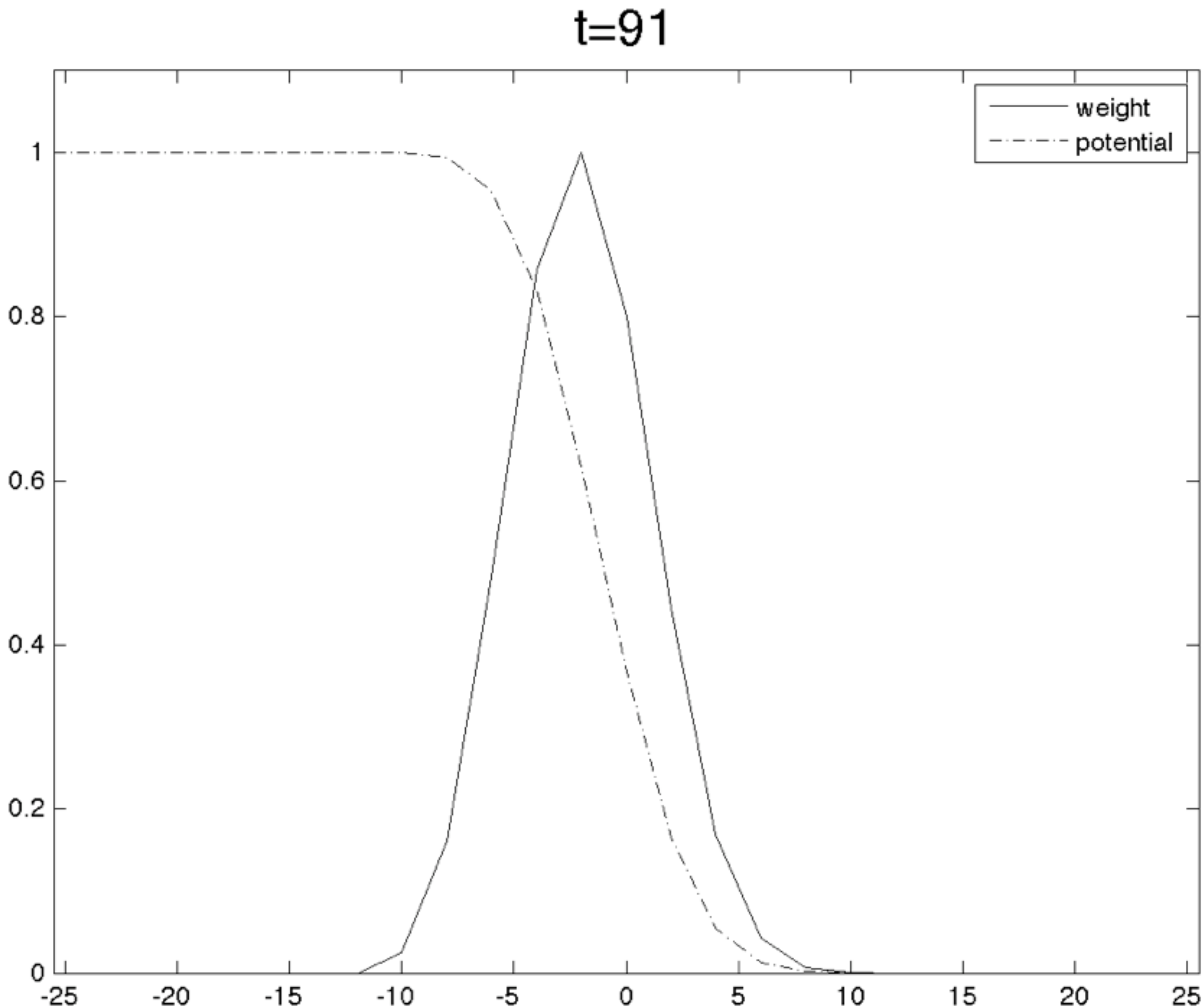
t=51
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25


t=61
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25

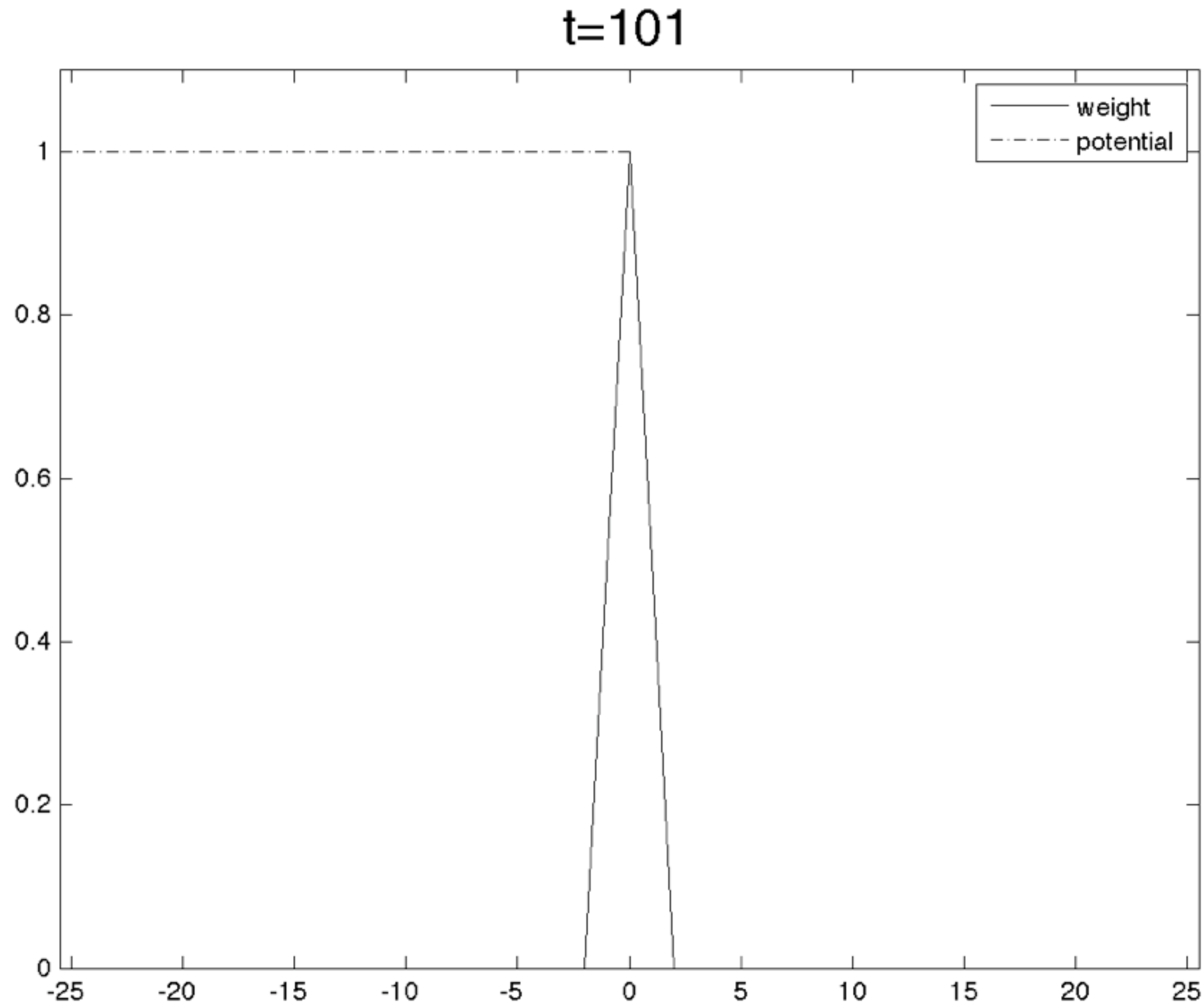

t=71
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25

t=81

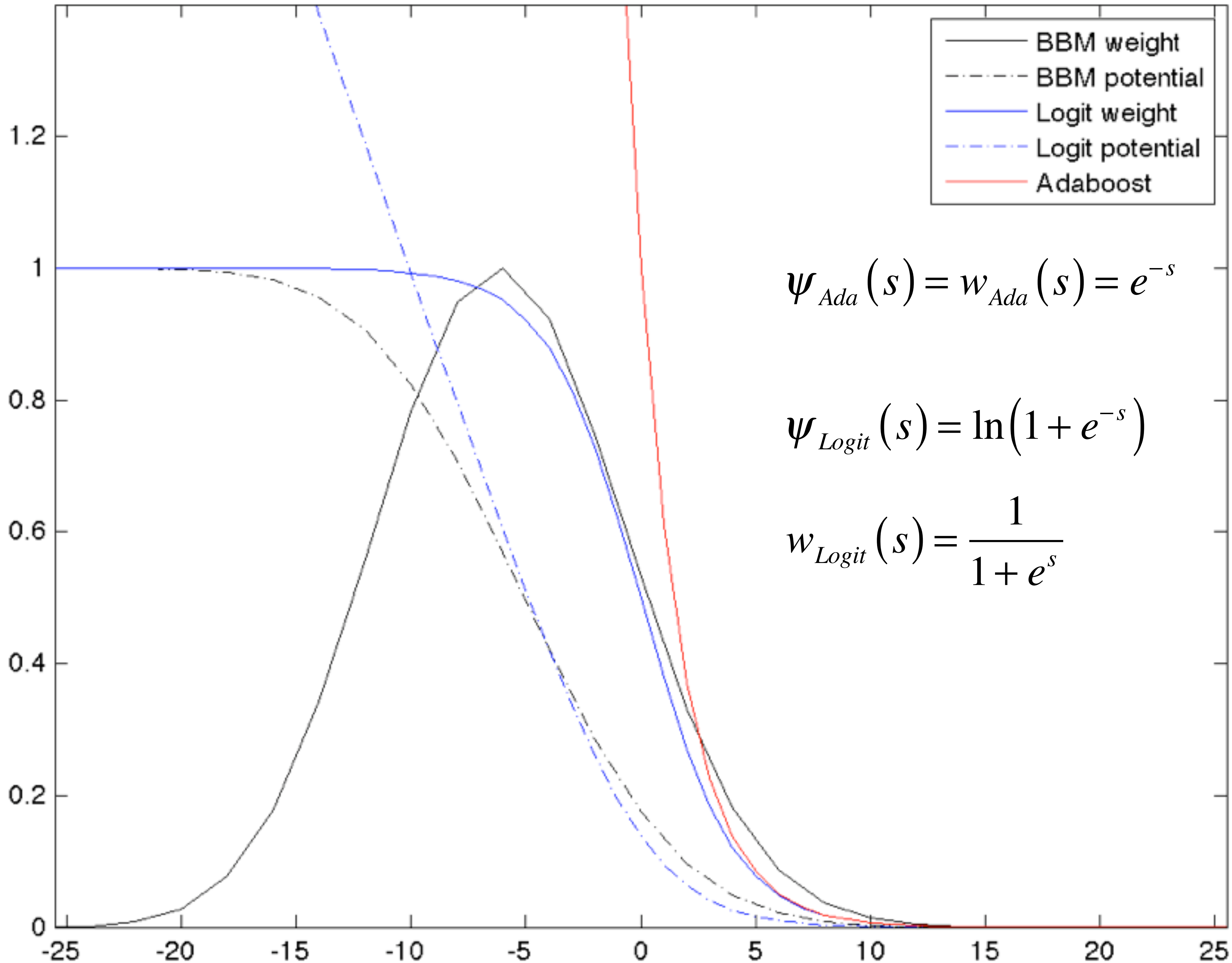




t=91
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25

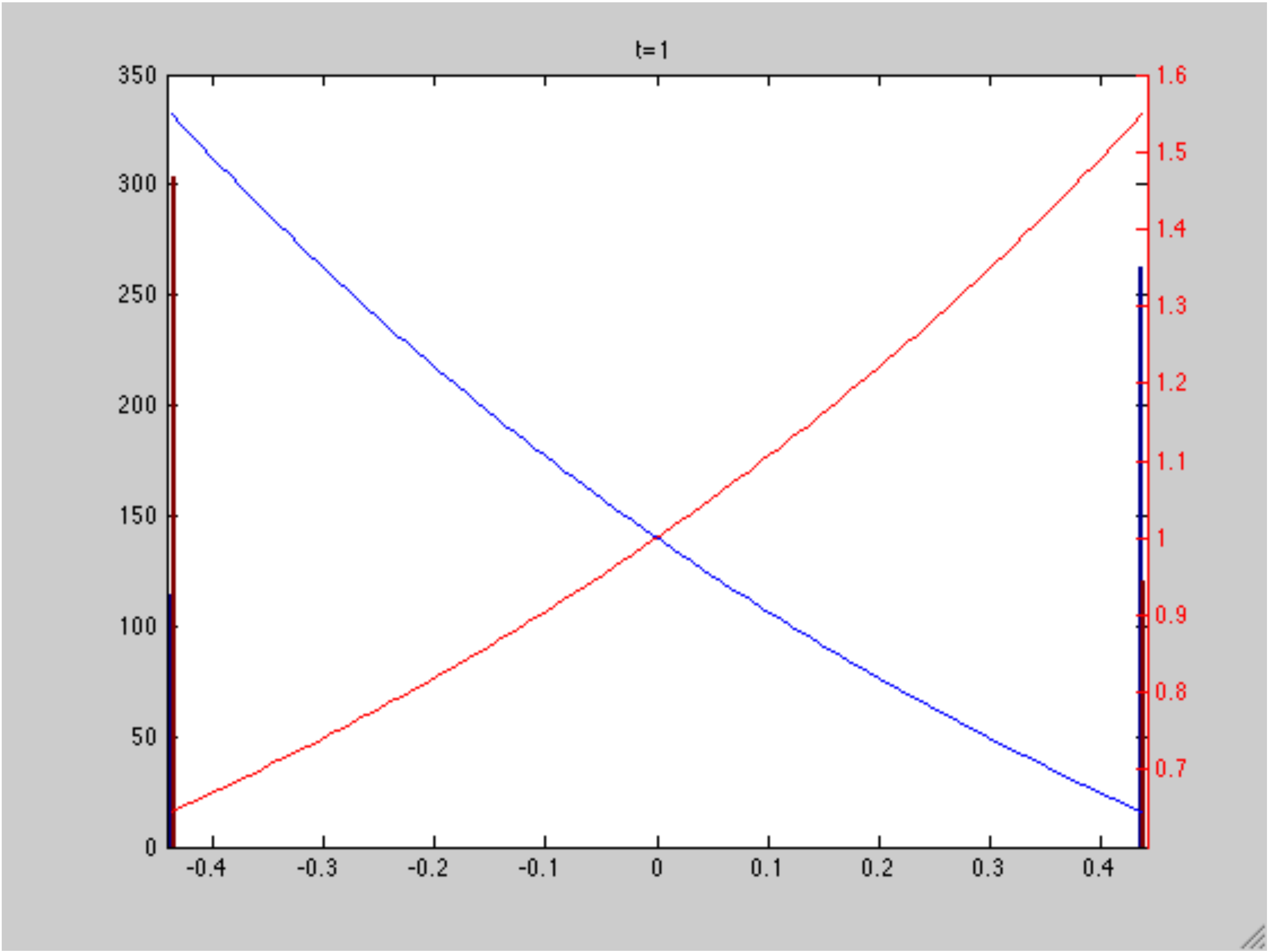

t=101
weight
potential
1
0.8
0.6
0.4
0.2
0
-25
-20
-15
-10
-5
0
5
10
15
20
25

BBM/Logitboost/Adaboost

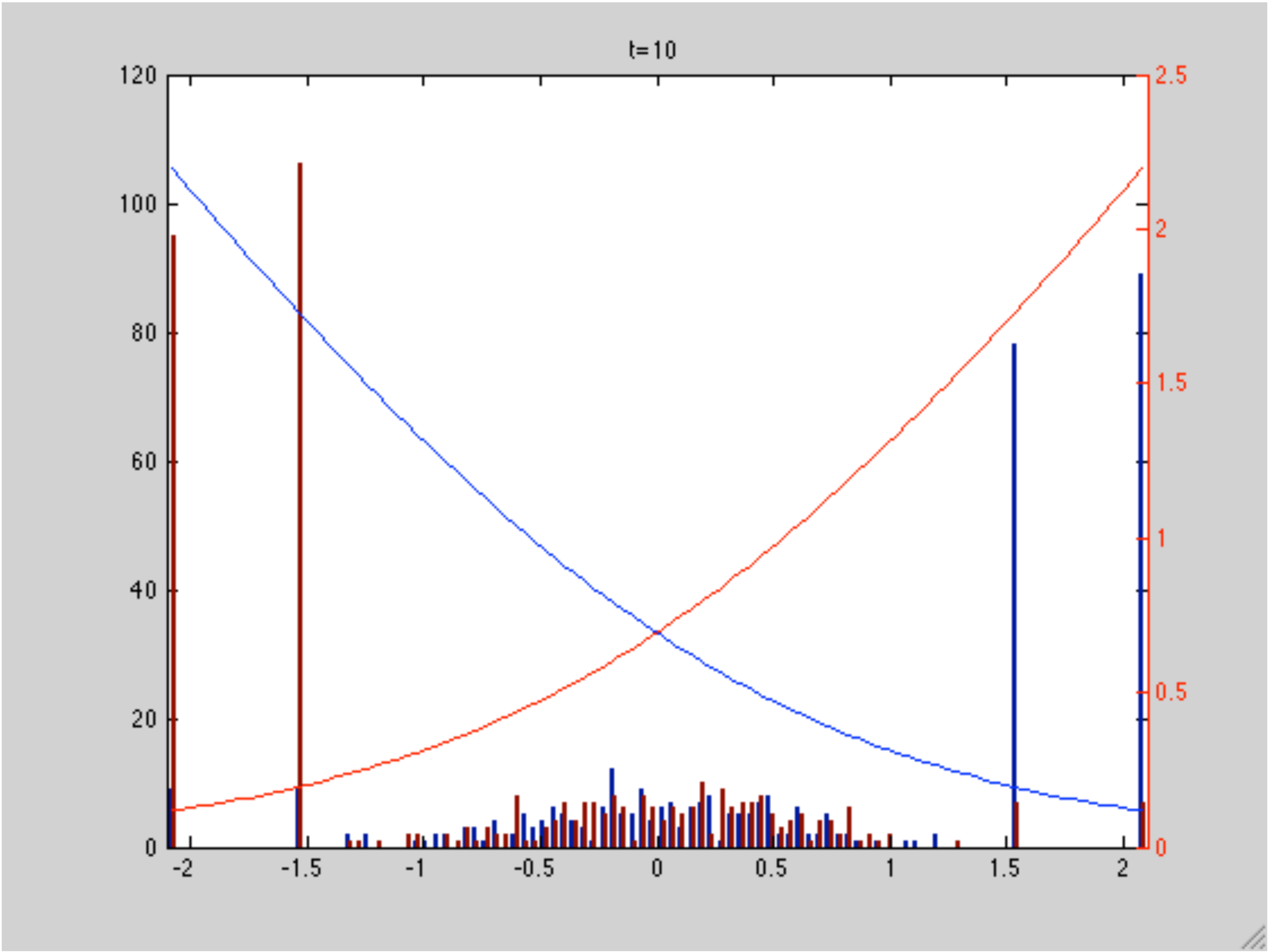


Experimental Results on Long/Servedio synthetic example

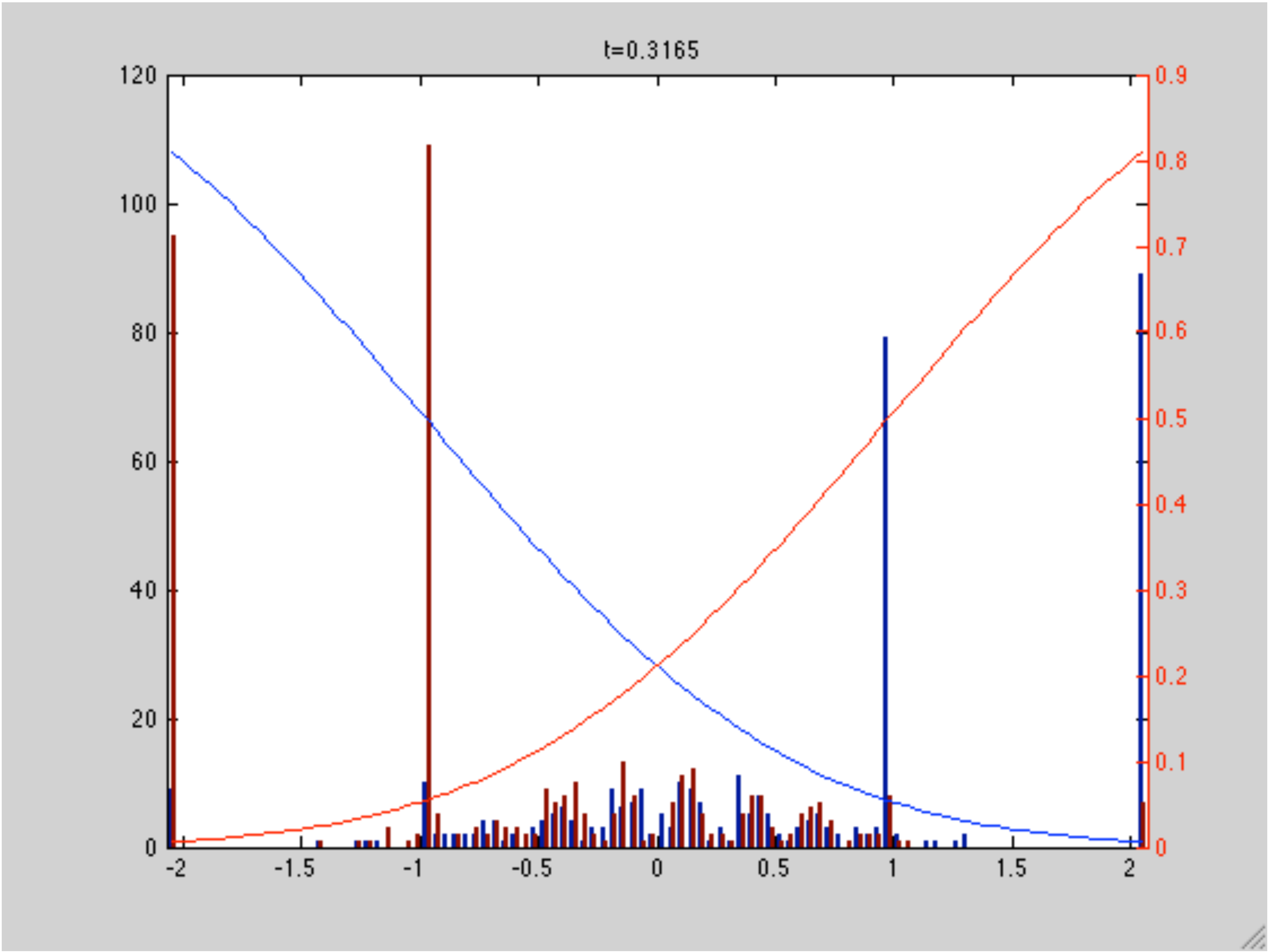
Adaboost on Long/Servedio



LogitBoost on Long/Servedio



Robustboost on Long/Servedio



Experimental Results on real-world data

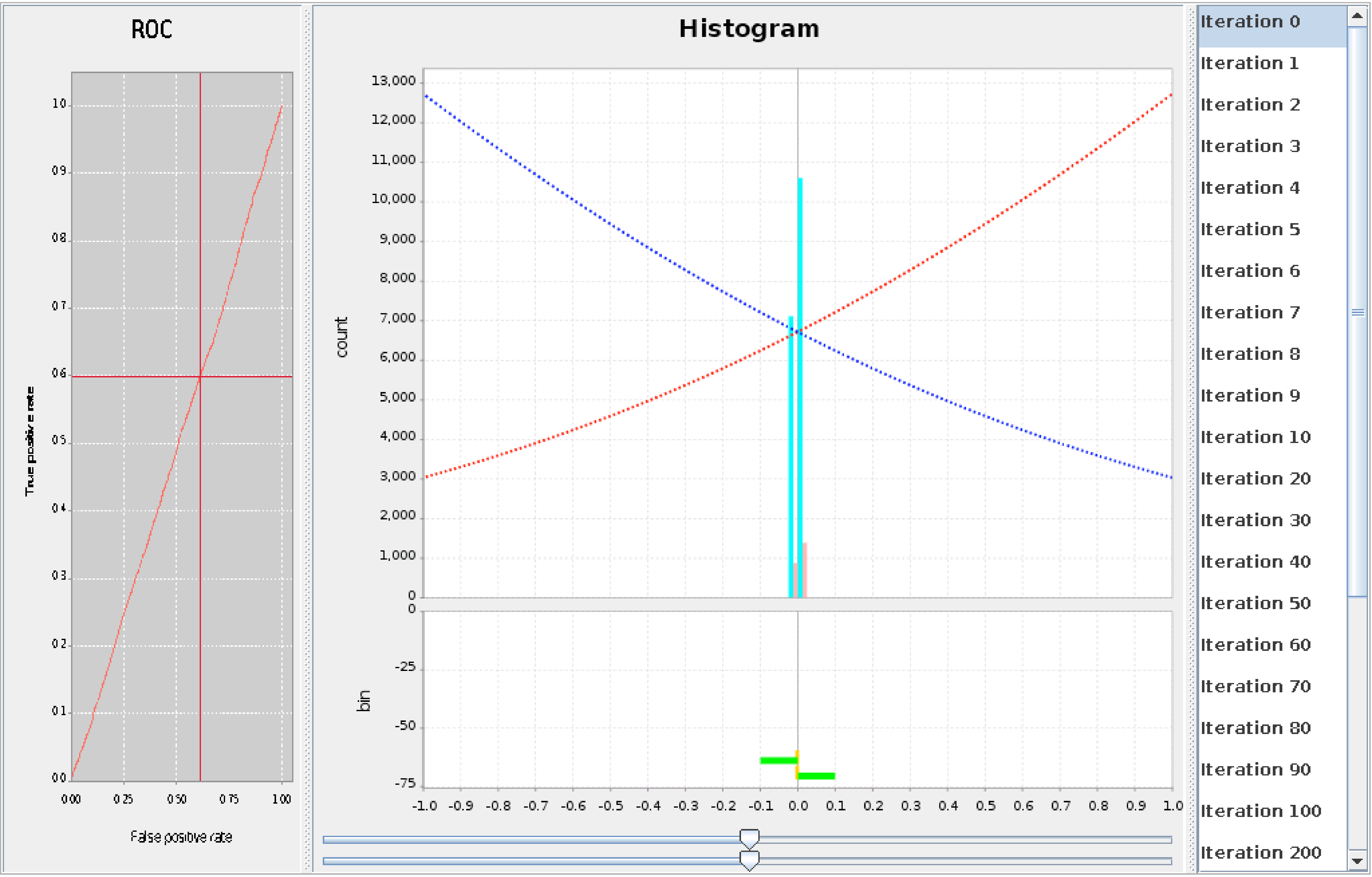
Robustboost - A new boosting algorithm

Label Noise	Adaboost	Logitboost	Robustboost
0%	0.8% ±0.2%	0.8% ±0.1%	2.9% ±0.2%
20%	33.3% ±0.7%	31.6% ±0.6%	22.2 ±0.8%

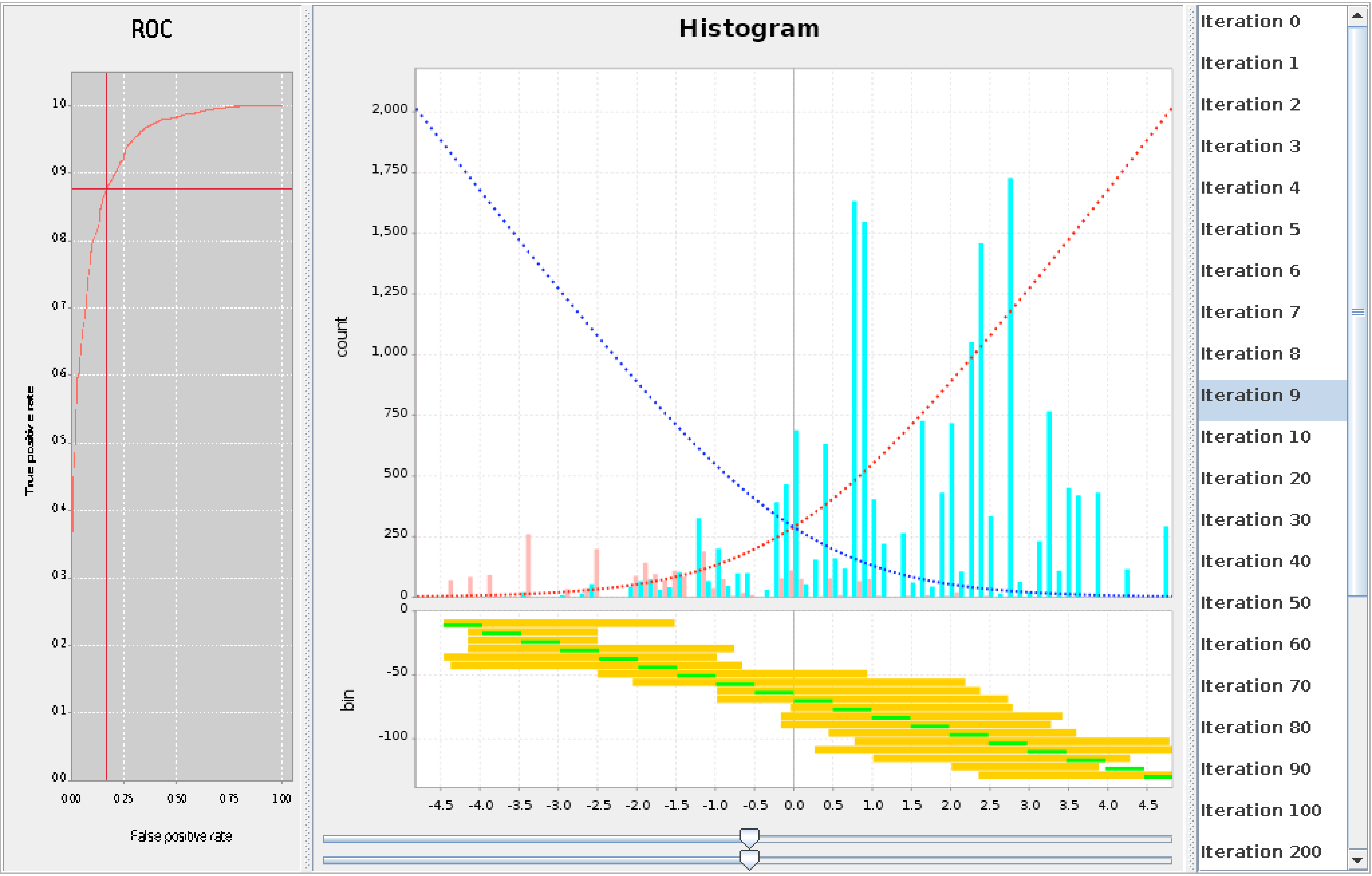
error with respect to original (noiseless) labels

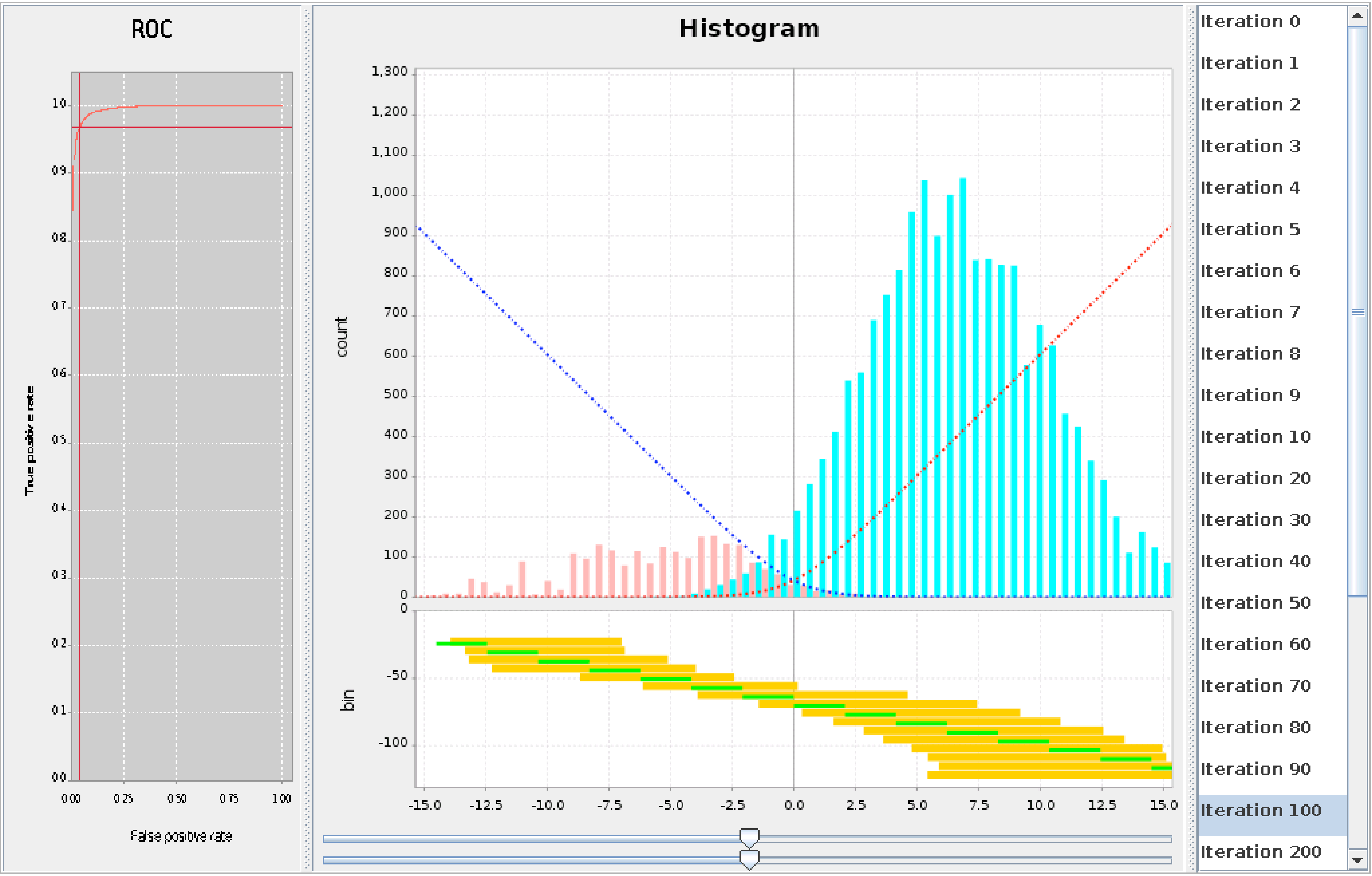
20%	22.1% ±1.2%	19.4% ±1.3%	**3.7% ±0.4%**

Logitboost
0% Noise

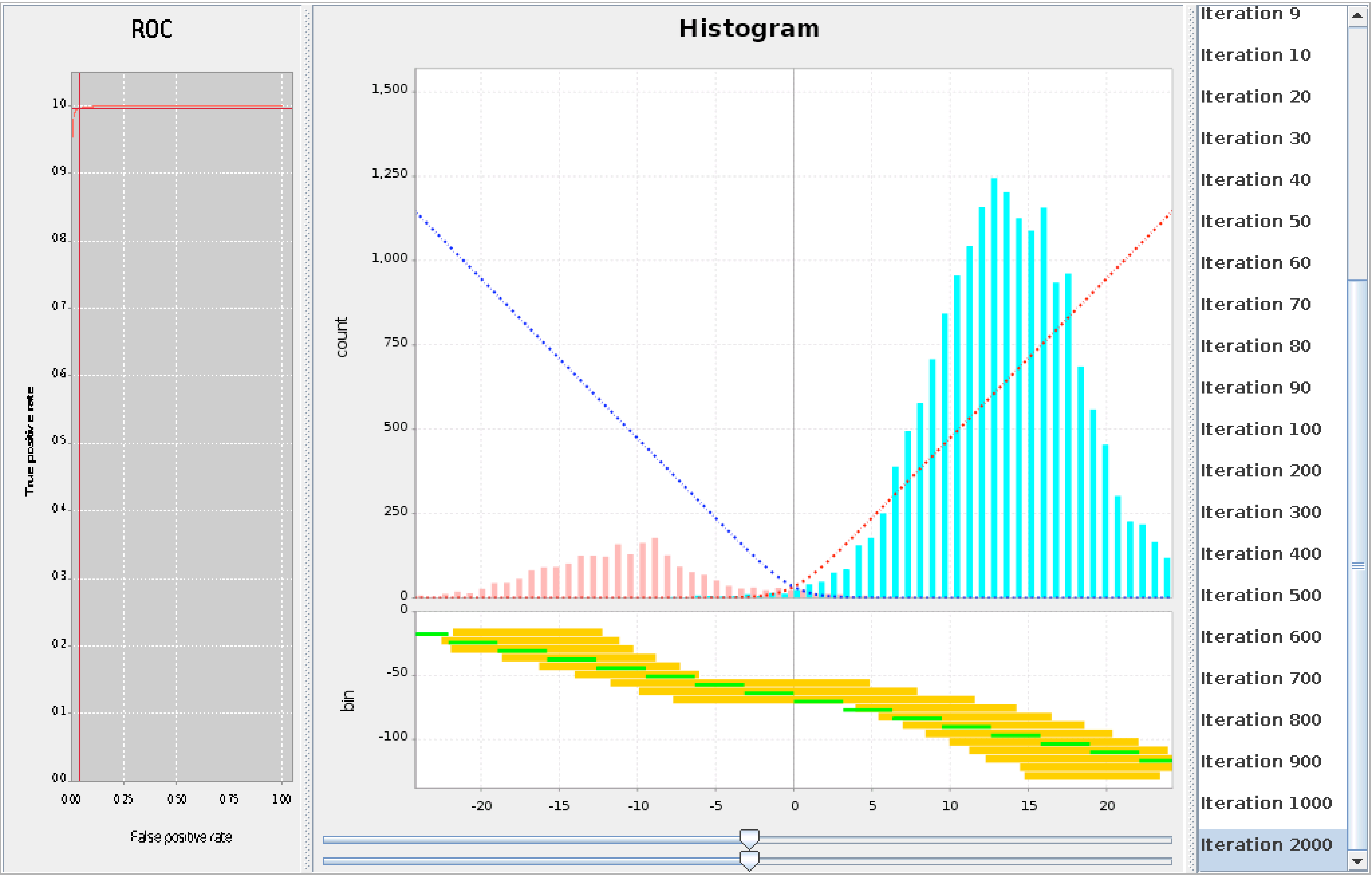


ROC
True positive rate
False positive rate
Histogram
count
bin
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9
Iteration 10
Iteration 20
Iteration 30
Iteration 40
Iteration 50
Iteration 60
Iteration 70
Iteration 80
Iteration 90
Iteration 100
Iteration 200

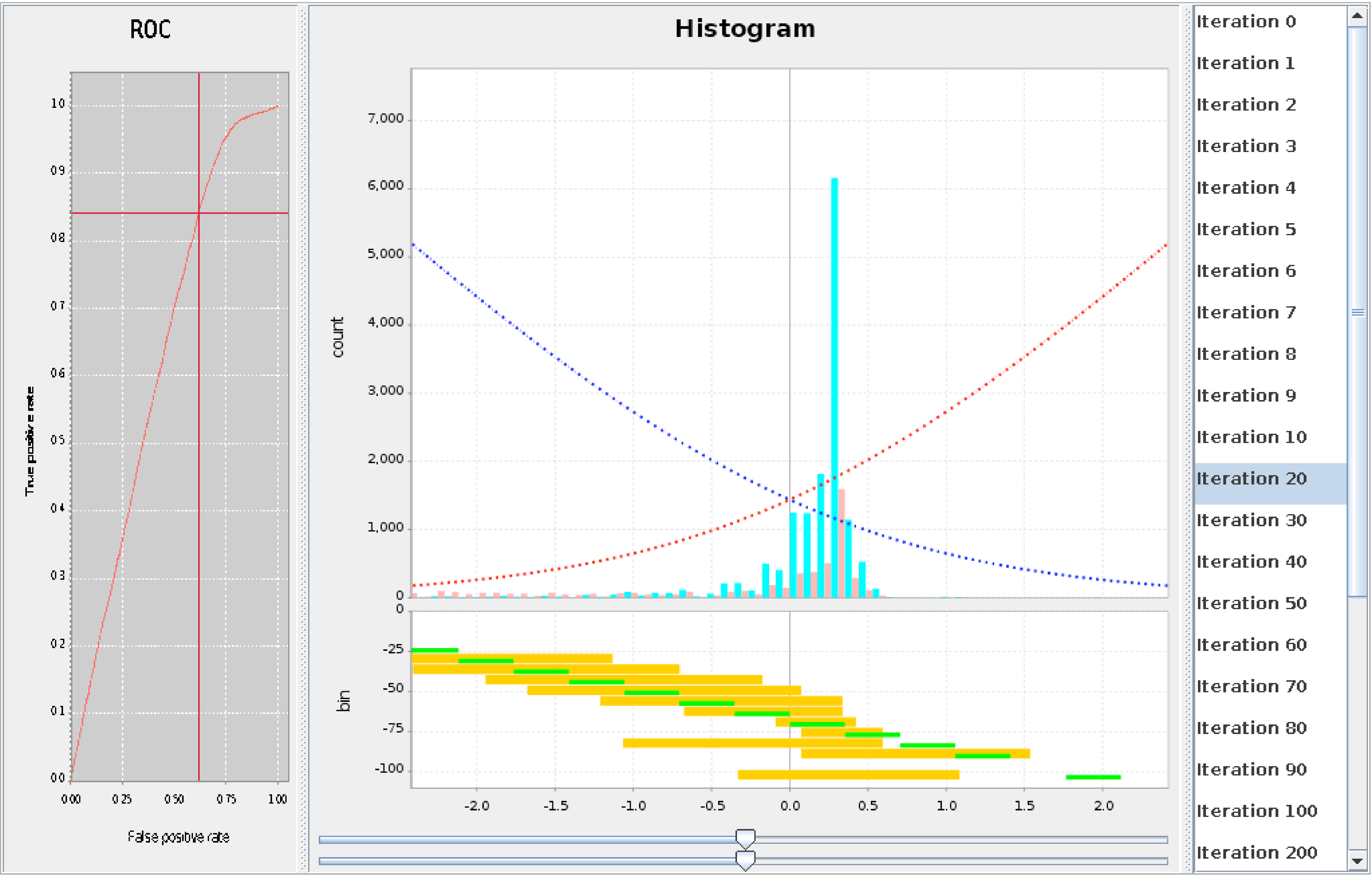

ROC
True positive rate
False positive rate
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
0.00
0.25
0.50
0.75
1.00
Histogram
count
bin
2,000
1,750
1,500
1,250
1,000
750
500
250
0
-50
-100
-4.5
-4.0
-3.5
-3.0
-2.5
-2.0
-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9
Iteration 10
Iteration 20
Iteration 30
Iteration 40
Iteration 50
Iteration 60
Iteration 70
Iteration 80
Iteration 90
Iteration 100
Iteration 200



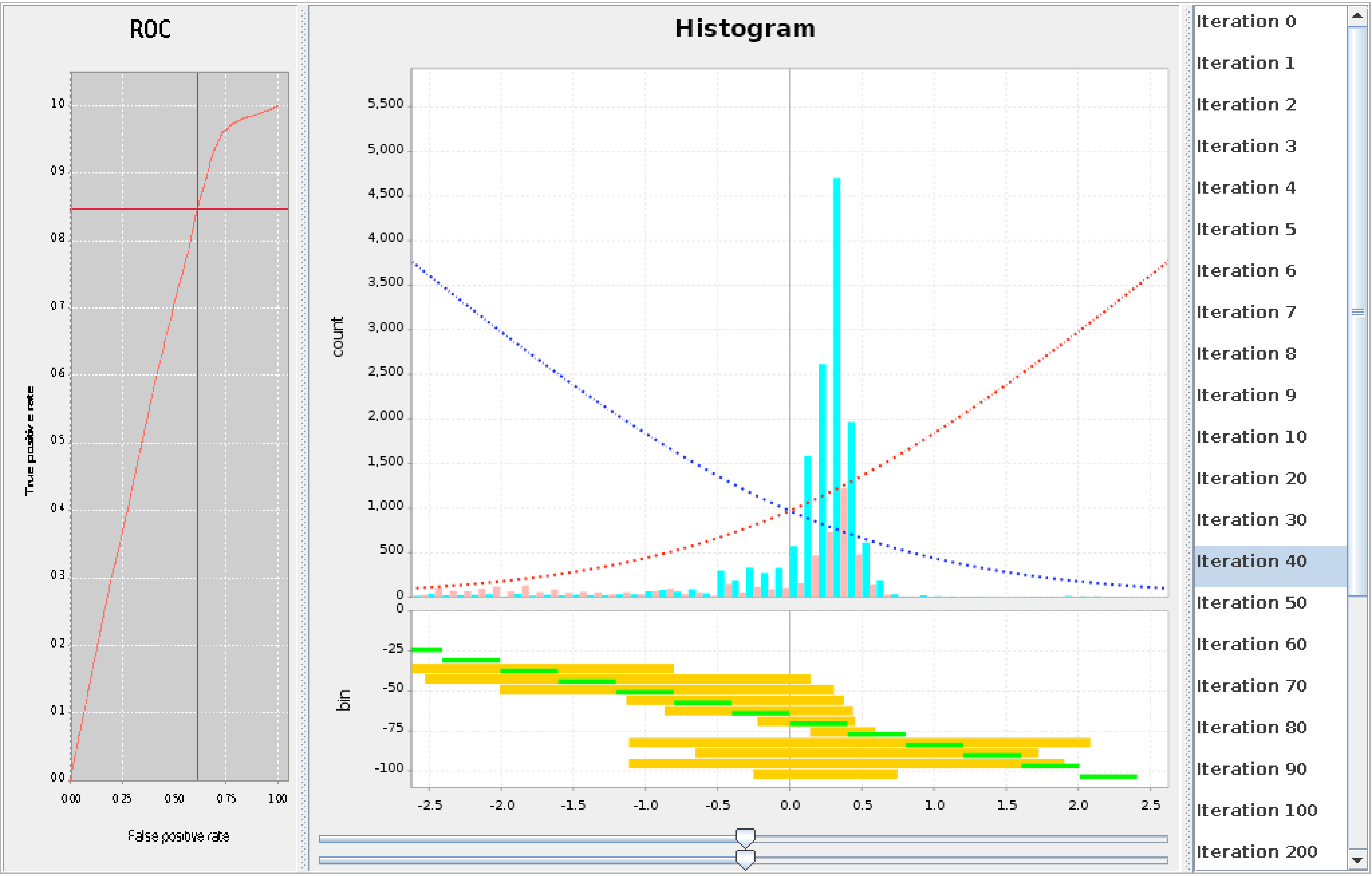
ROC
True positive rate
False positive rate
Histogram
count
bin
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9
Iteration 10
Iteration 20
Iteration 30
Iteration 40
Iteration 50
Iteration 60
Iteration 70
Iteration 80
Iteration 90
Iteration 100
Iteration 200

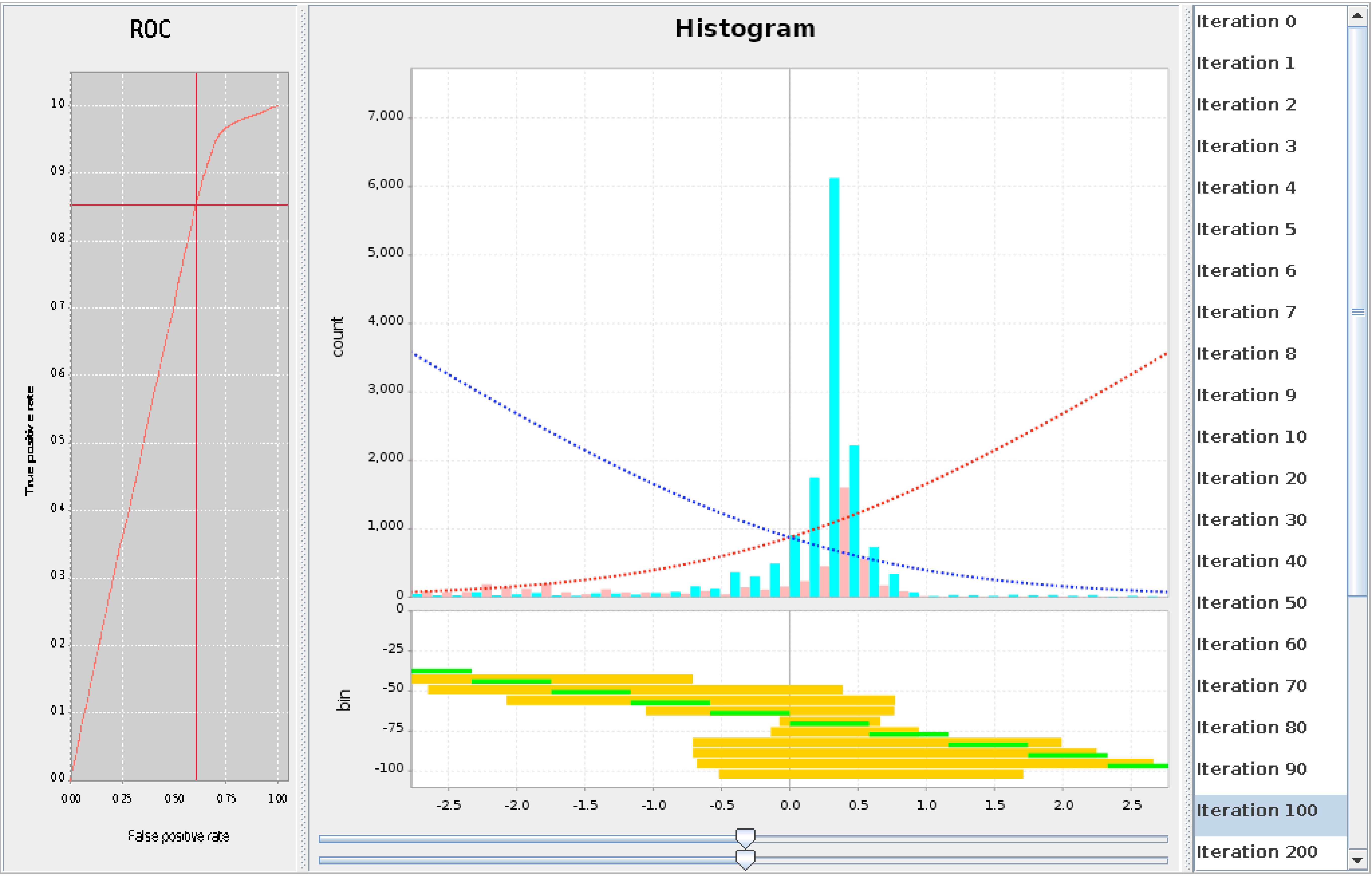

ROC
True positive rate
False positive rate
Histogram
count
bin
Iteration 9
Iteration 10
Iteration 20
Iteration 30
Iteration 40
Iteration 50
Iteration 60
Iteration 70
Iteration 80
Iteration 90
Iteration 100
Iteration 200
Iteration 300
Iteration 400
Iteration 500
Iteration 600
Iteration 700
Iteration 800
Iteration 900
Iteration 1000
Iteration 2000

Logitboost
20% Noise

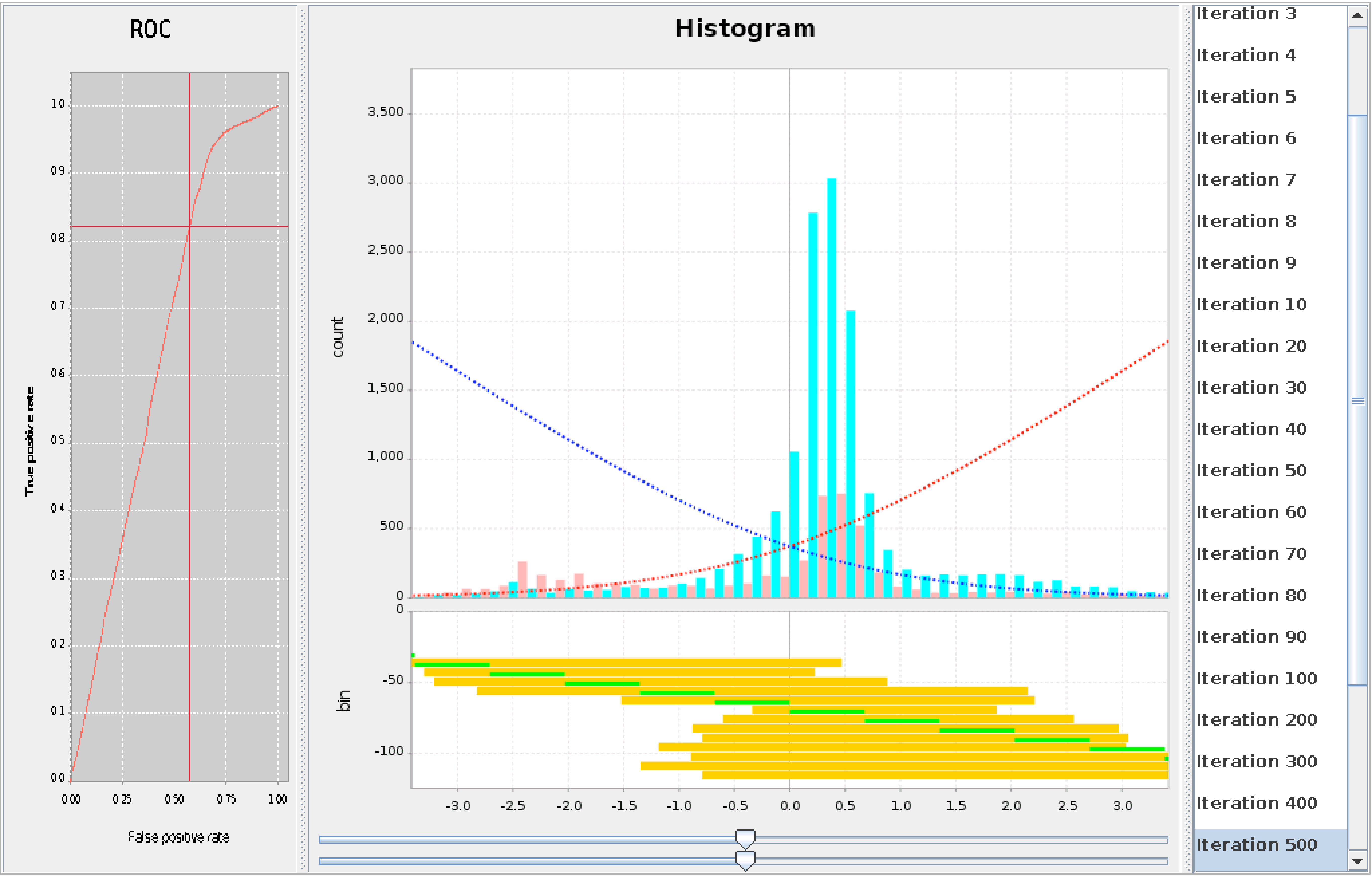


ROC
True positive rate
False positive rate
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
0.00
0.25
0.50
0.75
1.00
Histogram
count
7,000
6,000
5,000
4,000
3,000
2,000
1,000
0
bin
0
-25
-50
-75
-100
-2.0
-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5
2.0
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9
Iteration 10
Iteration 20
Iteration 30
Iteration 40
Iteration 50
Iteration 60
Iteration 70
Iteration 80
Iteration 90
Iteration 100
Iteration 200

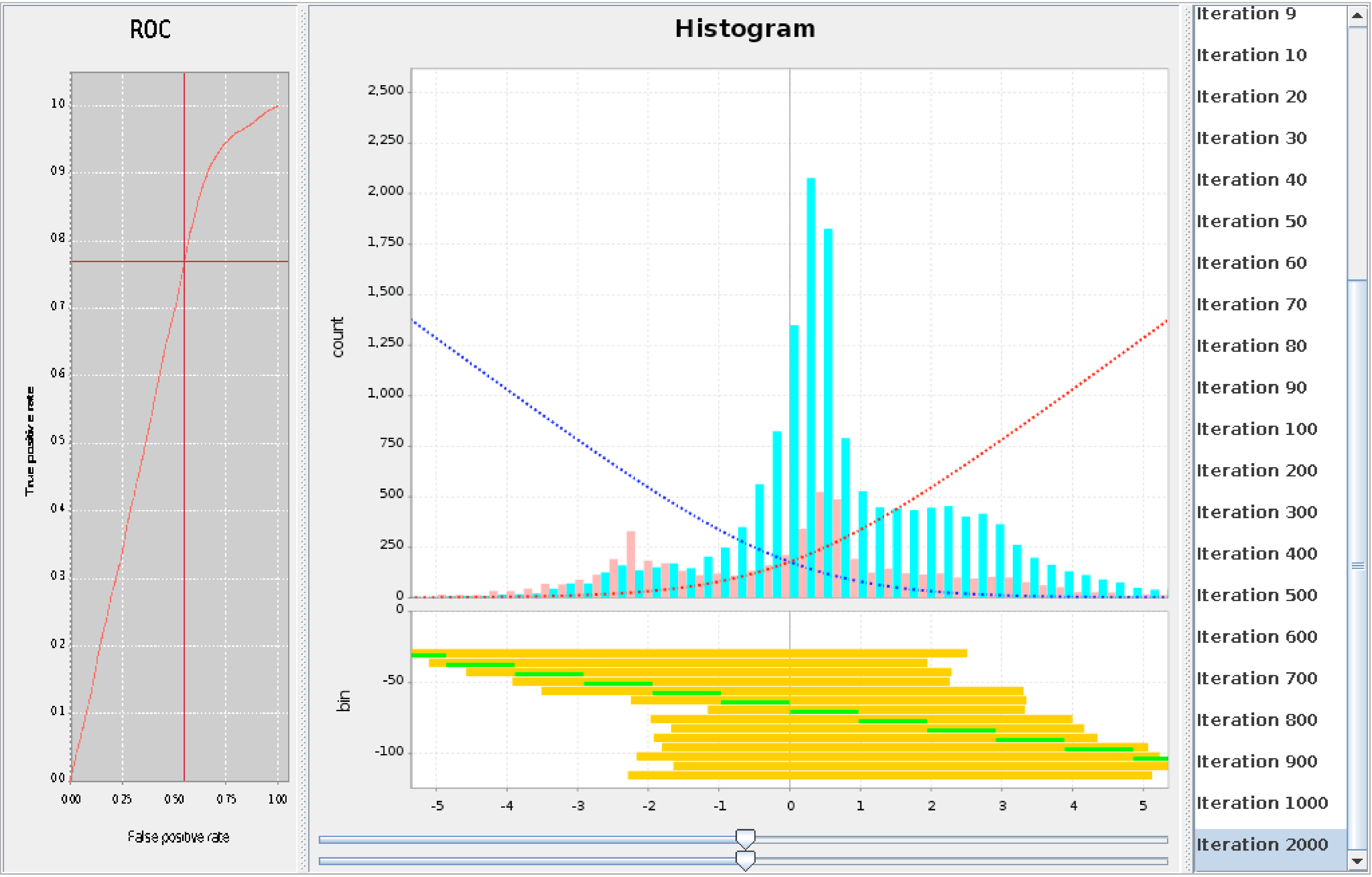

ROC
True positive rate
False positive rate
Histogram
count
bin
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9
Iteration 10
Iteration 20
Iteration 30
Iteration 40
Iteration 50
Iteration 60
Iteration 70
Iteration 80
Iteration 90
Iteration 100
Iteration 200



ROC
True positive rate
False positive rate
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
0.00
0.25
0.50
0.75
1.00
Histogram
count
7,000
6,000
5,000
4,000
3,000
2,000
1,000
0
bin
-25
-50
-75
-100
-2.5
-2.0
-1.5
-1.0
-0.5
0.0
0.5
1.0
1.5
2.0
2.5
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9
Iteration 10
Iteration 20
Iteration 30
Iteration 40
Iteration 50
Iteration 60
Iteration 70
Iteration 80
Iteration 90
Iteration 100
Iteration 200

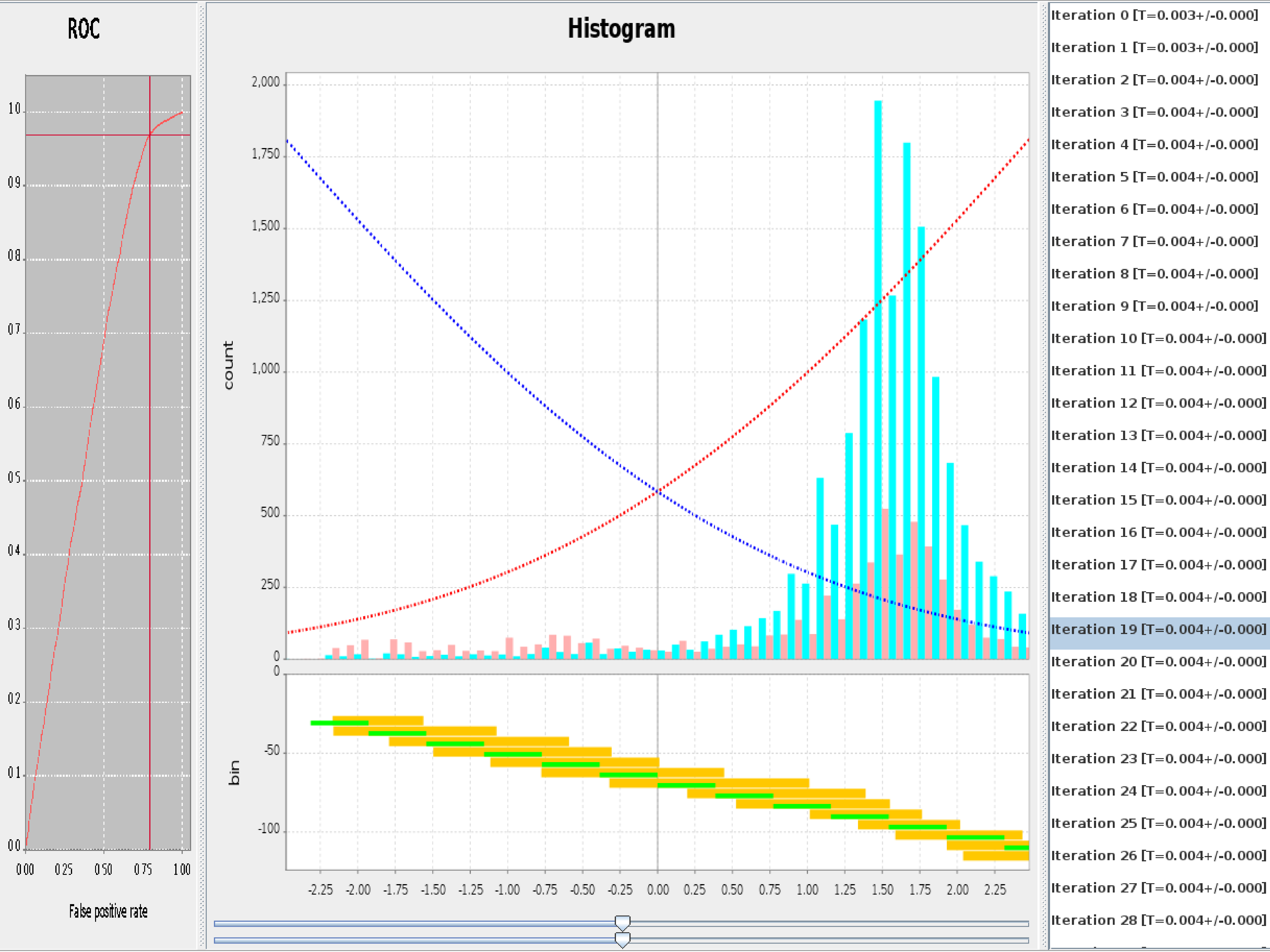


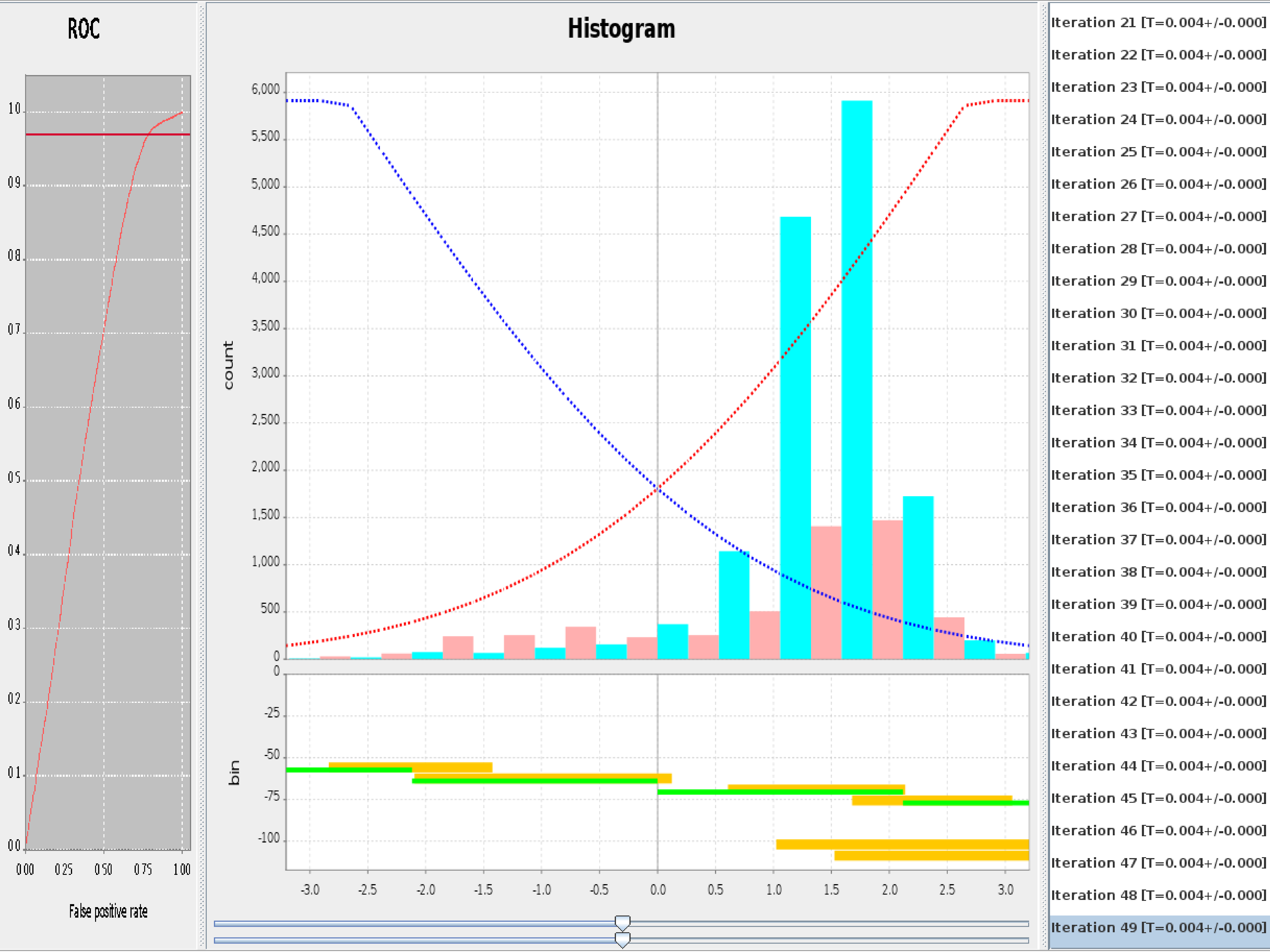
ROC
True positive rate
False positive rate
Histogram
count
bin
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
Iteration 9
Iteration 10
Iteration 20
Iteration 30
Iteration 40
Iteration 50
Iteration 60
Iteration 70
Iteration 80
Iteration 90
Iteration 100
Iteration 200
Iteration 300
Iteration 400
Iteration 500

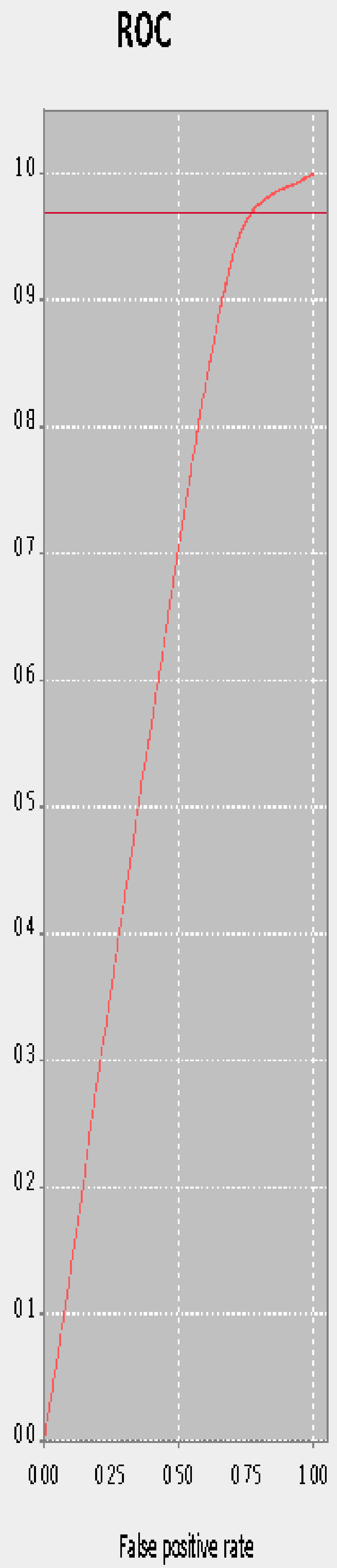


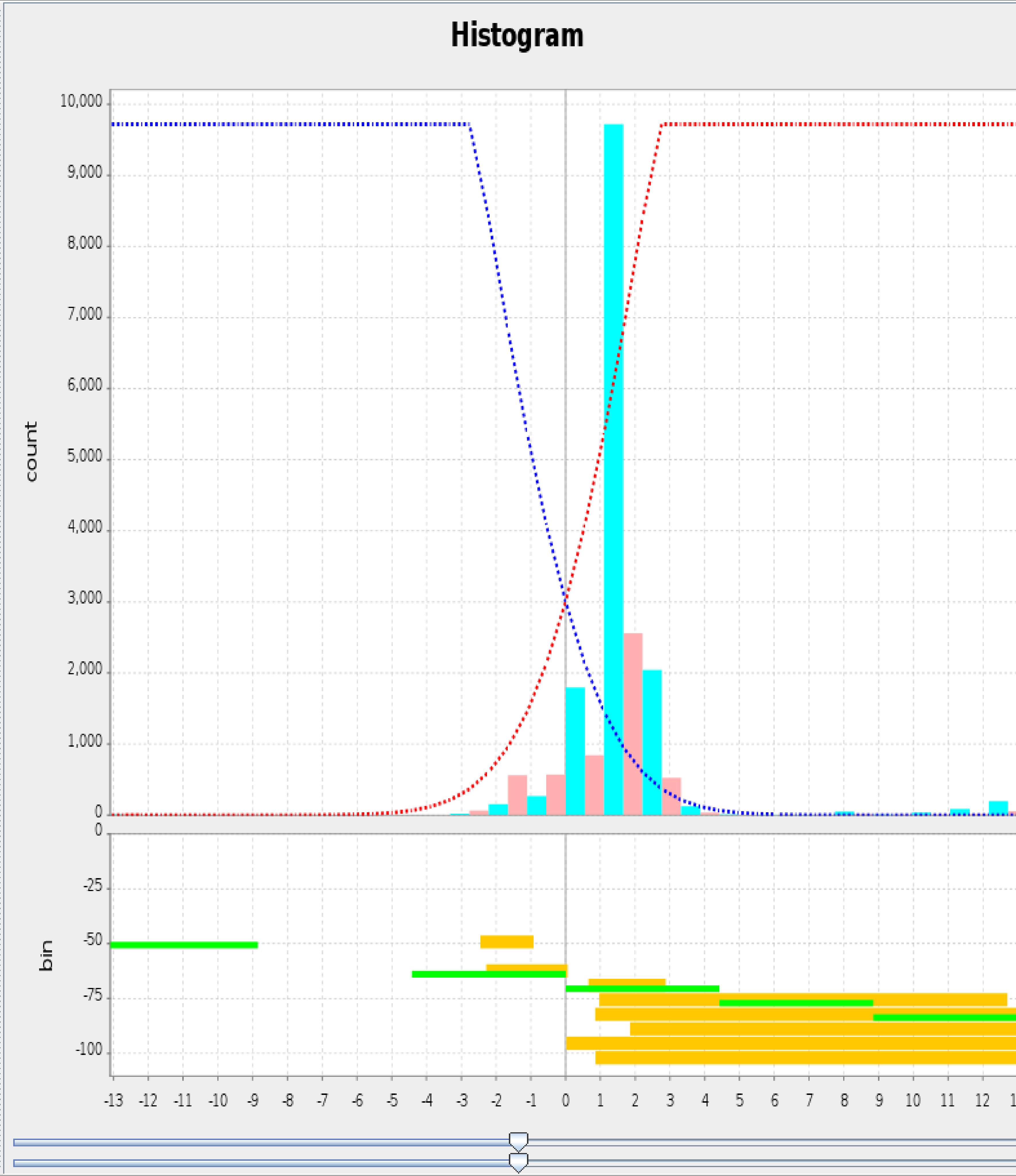
ROC
True positive rate
False positive rate
Histogram
count
bin
Iteration 9
Iteration 10
Iteration 20
Iteration 30
Iteration 40
Iteration 50
Iteration 60
Iteration 70
Iteration 80
Iteration 90
Iteration 100
Iteration 200
Iteration 300
Iteration 400
Iteration 500
Iteration 600
Iteration 700
Iteration 800
Iteration 900
Iteration 1000
Iteration 2000

Robustboost
20% Noise

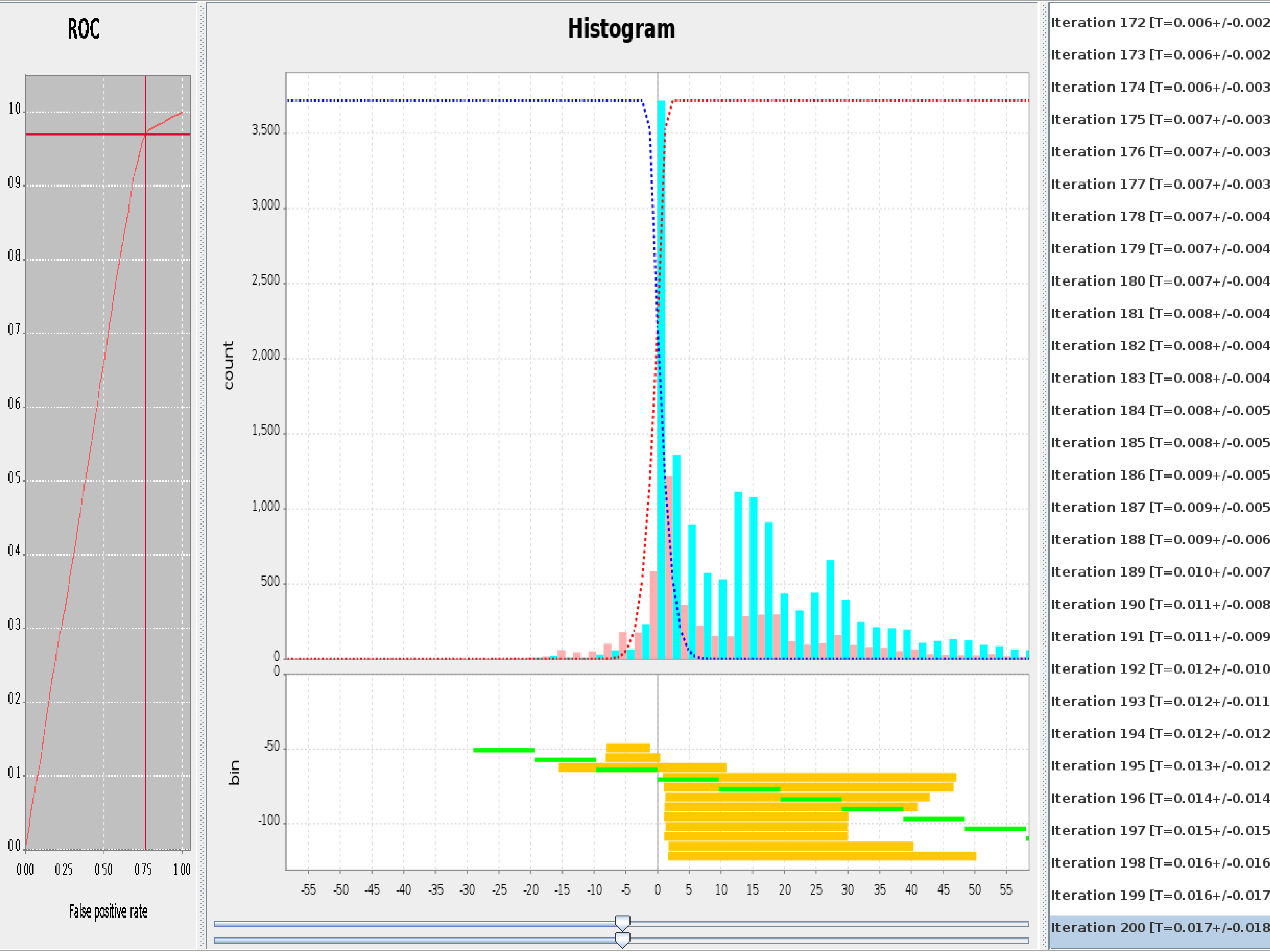

ROC
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0.00
0.25
0.50
0.75
1.00
False positive rate
Histogram
count
2,000
1,750
1,500
1,250
1,000
750
500
250
0
bin
0
-50
-100
-2.25
-2.00
-1.75
-1.50
-1.25
-1.00
-0.75
-0.50
-0.25
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
2.25
Iteration 0 [T=0.003+/-0.000]
Iteration 1 [T=0.003+/-0.000]
Iteration 2 [T=0.004+/-0.000]
Iteration 3 [T=0.004+/-0.000]
Iteration 4 [T=0.004+/-0.000]
Iteration 5 [T=0.004+/-0.000]
Iteration 6 [T=0.004+/-0.000]
Iteration 7 [T=0.004+/-0.000]
Iteration 8 [T=0.004+/-0.000]
Iteration 9 [T=0.004+/-0.000]
Iteration 10 [T=0.004+/-0.000]
Iteration 11 [T=0.004+/-0.000]
Iteration 12 [T=0.004+/-0.000]
Iteration 13 [T=0.004+/-0.000]
Iteration 14 [T=0.004+/-0.000]
Iteration 15 [T=0.004+/-0.000]
Iteration 16 [T=0.004+/-0.000]
Iteration 17 [T=0.004+/-0.000]
Iteration 18 [T=0.004+/-0.000]
Iteration 19 [T=0.004+/-0.000]
Iteration 20 [T=0.004+/-0.000]
Iteration 21 [T=0.004+/-0.000]
Iteration 22 [T=0.004+/-0.000]
Iteration 23 [T=0.004+/-0.000]
Iteration 24 [T=0.004+/-0.000]
Iteration 25 [T=0.004+/-0.000]
Iteration 26 [T=0.004+/-0.000]
Iteration 27 [T=0.004+/-0.000]
Iteration 28 [T=0.004+/-0.000]

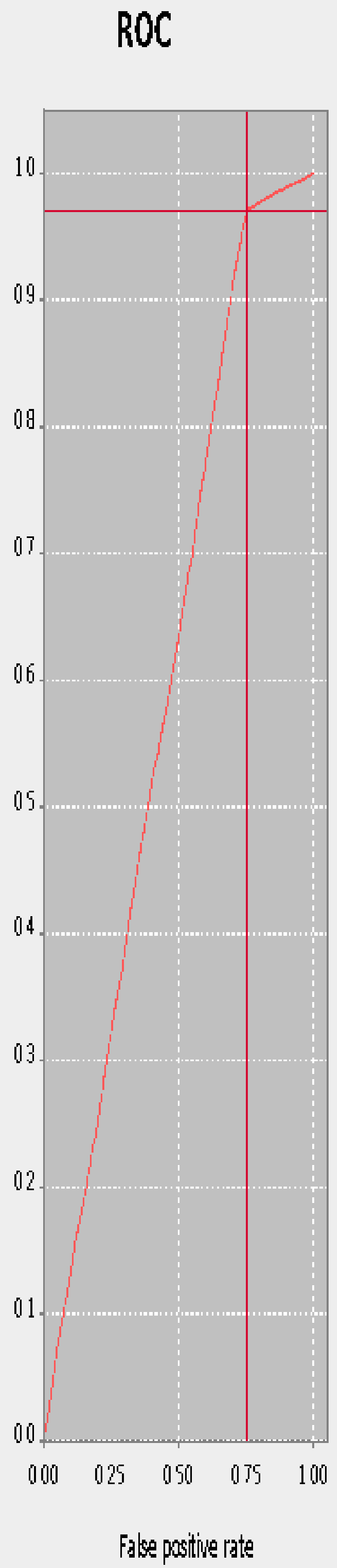

ROC
False positive rate
Histogram
count
bin
Iteration 21 [T=0.004+/-0.000]
Iteration 22 [T=0.004+/-0.000]
Iteration 23 [T=0.004+/-0.000]
Iteration 24 [T=0.004+/-0.000]
Iteration 25 [T=0.004+/-0.000]
Iteration 26 [T=0.004+/-0.000]
Iteration 27 [T=0.004+/-0.000]
Iteration 28 [T=0.004+/-0.000]
Iteration 29 [T=0.004+/-0.000]
Iteration 30 [T=0.004+/-0.000]
Iteration 31 [T=0.004+/-0.000]
Iteration 32 [T=0.004+/-0.000]
Iteration 33 [T=0.004+/-0.000]
Iteration 34 [T=0.004+/-0.000]
Iteration 35 [T=0.004+/-0.000]
Iteration 36 [T=0.004+/-0.000]
Iteration 37 [T=0.004+/-0.000]
Iteration 38 [T=0.004+/-0.000]
Iteration 39 [T=0.004+/-0.000]
Iteration 40 [T=0.004+/-0.000]
Iteration 41 [T=0.004+/-0.000]
Iteration 42 [T=0.004+/-0.000]
Iteration 43 [T=0.004+/-0.000]
Iteration 44 [T=0.004+/-0.000]
Iteration 45 [T=0.004+/-0.000]
Iteration 46 [T=0.004+/-0.000]
Iteration 47 [T=0.004+/-0.000]
Iteration 48 [T=0.004+/-0.000]
Iteration 49 [T=0.004+/-0.000]

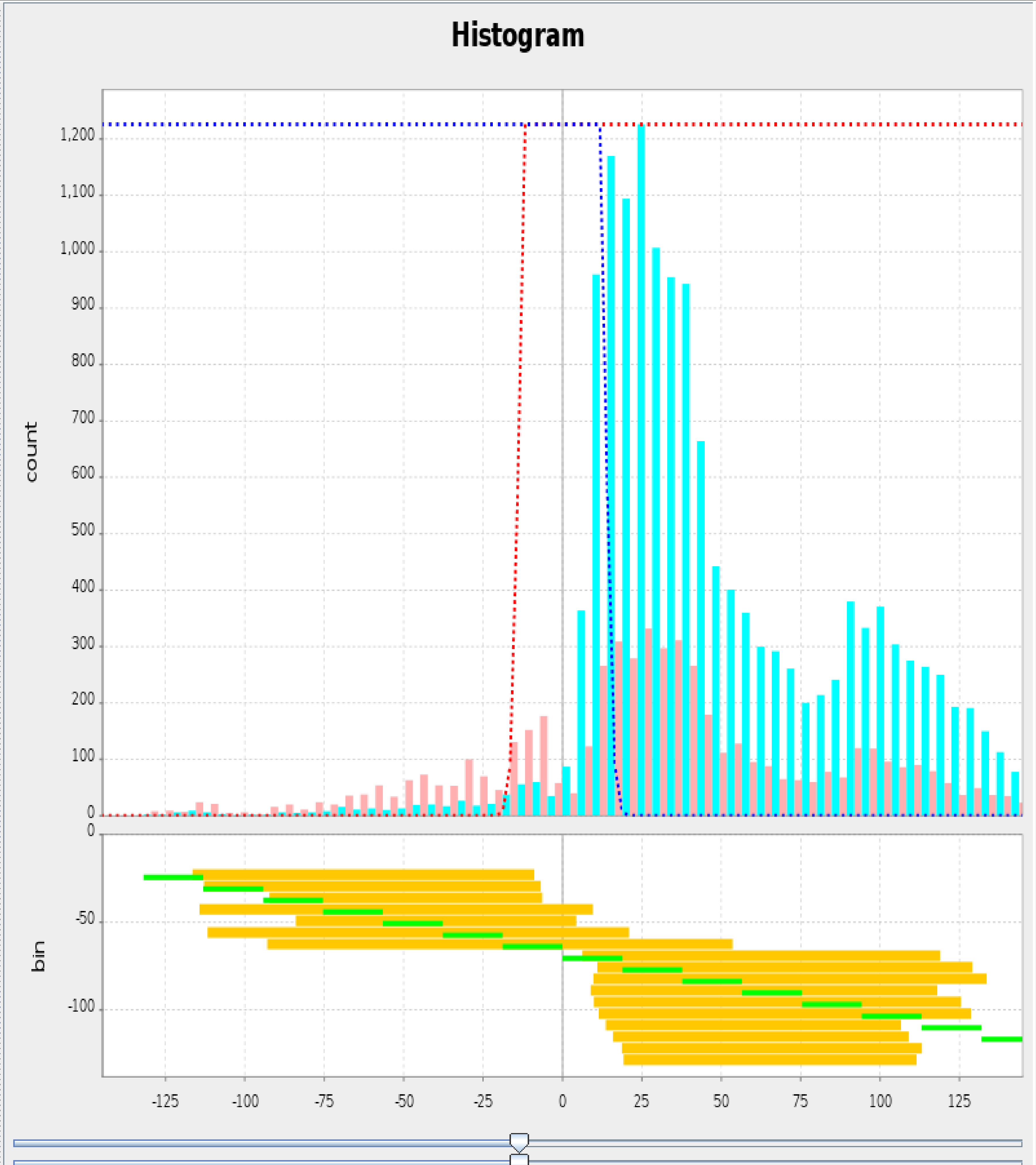




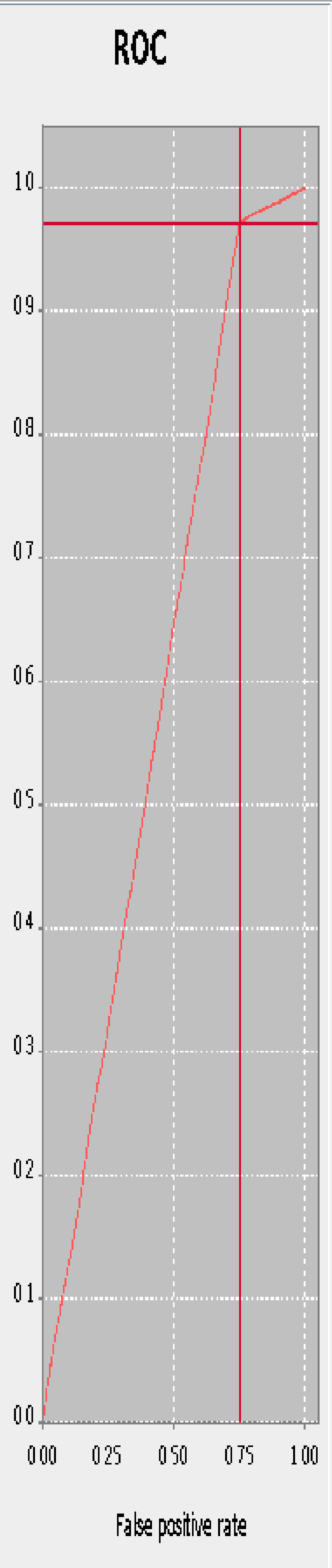
Iteration 72 [T=0.005+/-0.000]
Iteration 73 [T=0.005+/-0.000]
Iteration 74 [T=0.005+/-0.000]
Iteration 75 [T=0.005+/-0.000]
Iteration 76 [T=0.005+/-0.000]
Iteration 77 [T=0.005+/-0.000]
Iteration 78 [T=0.005+/-0.000]
Iteration 79 [T=0.005+/-0.000]
Iteration 80 [T=0.005+/-0.000]
Iteration 81 [T=0.005+/-0.000]
Iteration 82 [T=0.005+/-0.000]
Iteration 83 [T=0.005+/-0.000]
Iteration 84 [T=0.005+/-0.000]
Iteration 85 [T=0.005+/-0.000]
Iteration 86 [T=0.005+/-0.000]
Iteration 87 [T=0.005+/-0.000]
Iteration 88 [T=0.005+/-0.000]
Iteration 89 [T=0.005+/-0.000]
Iteration 90 [T=0.005+/-0.000]
Iteration 91 [T=0.005+/-0.000]
Iteration 92 [T=0.005+/-0.000]
Iteration 93 [T=0.005+/-0.000]
Iteration 94 [T=0.005+/-0.000]
Iteration 95 [T=0.005+/-0.000]
Iteration 96 [T=0.005+/-0.000]
Iteration 97 [T=0.005+/-0.000]
Iteration 98 [T=0.005+/-0.000]
Iteration 99 [T=0.005+/-0.000]
Iteration 100 [T=0.005+/-0.000

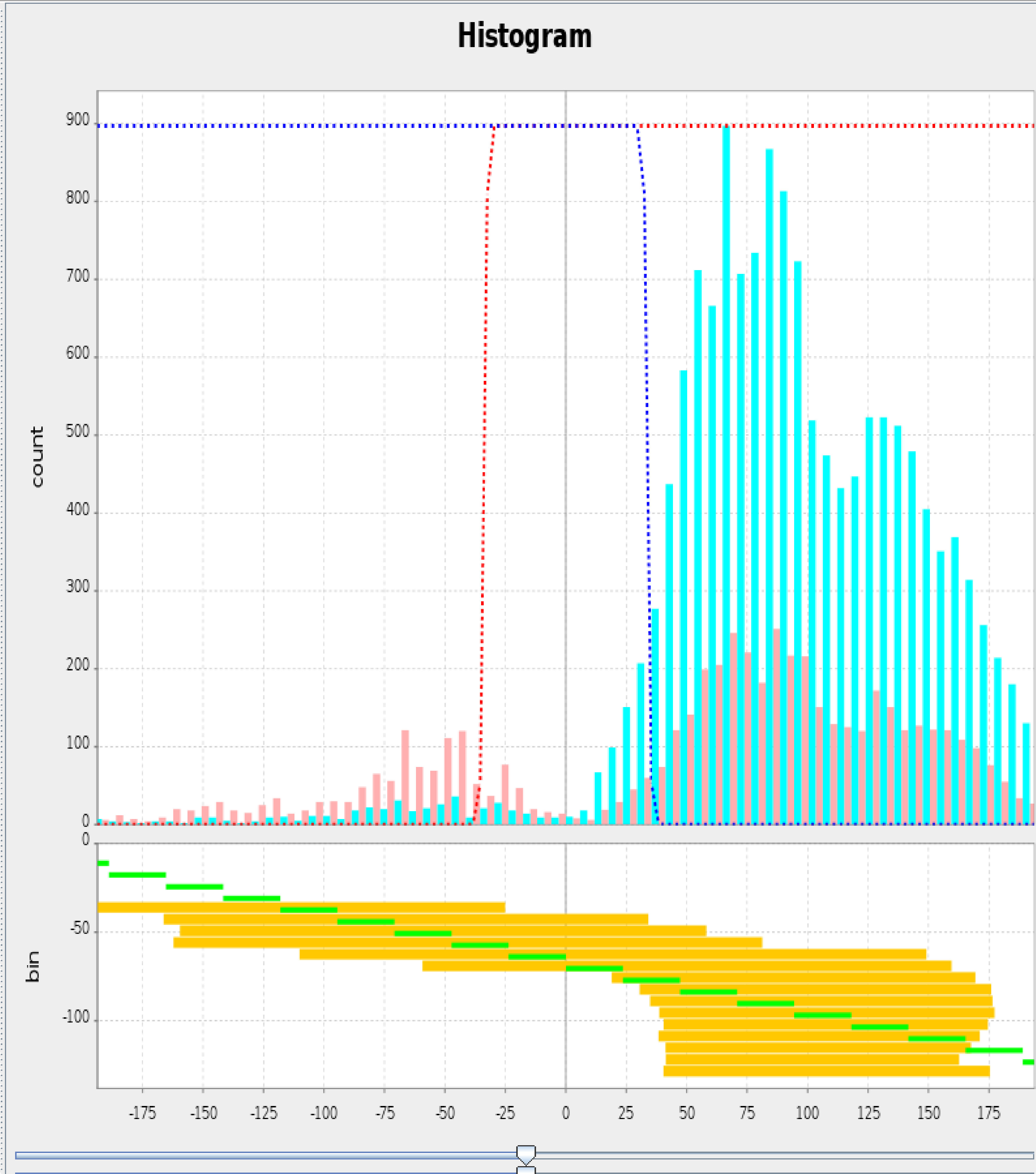

ROC
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0.00
0.25
0.50
0.75
1.00
False positive rate
Histogram
count
3,500
3,000
2,500
2,000
1,500
1,000
500
0
bin
-50
-100
-55 -50 -45 -40 -35 -30 -25 -20 -15 -10 -5 0 5 10 15 20 25 30 35 40 45 50 55
Iteration 172 [T=0.006+/-0.002
Iteration 173 [T=0.006+/-0.002
Iteration 174 [T=0.006+/-0.003
Iteration 175 [T=0.007+/-0.003
Iteration 176 [T=0.007+/-0.003
Iteration 177 [T=0.007+/-0.003
Iteration 178 [T=0.007+/-0.004
Iteration 179 [T=0.007+/-0.004
Iteration 180 [T=0.007+/-0.004
Iteration 181 [T=0.008+/-0.004
Iteration 182 [T=0.008+/-0.004
Iteration 183 [T=0.008+/-0.004
Iteration 184 [T=0.008+/-0.005
Iteration 185 [T=0.008+/-0.005
Iteration 186 [T=0.009+/-0.005
Iteration 187 [T=0.009+/-0.005
Iteration 188 [T=0.009+/-0.006
Iteration 189 [T=0.010+/-0.007
Iteration 190 [T=0.011+/-0.008
Iteration 191 [T=0.011+/-0.009
Iteration 192 [T=0.012+/-0.010
Iteration 193 [T=0.012+/-0.011
Iteration 194 [T=0.012+/-0.012
Iteration 195 [T=0.013+/-0.012
Iteration 196 [T=0.014+/-0.014
Iteration 197 [T=0.015+/-0.015
Iteration 198 [T=0.016+/-0.016
Iteration 199 [T=0.016+/-0.017
Iteration 200 [T=0.017+/-0.018



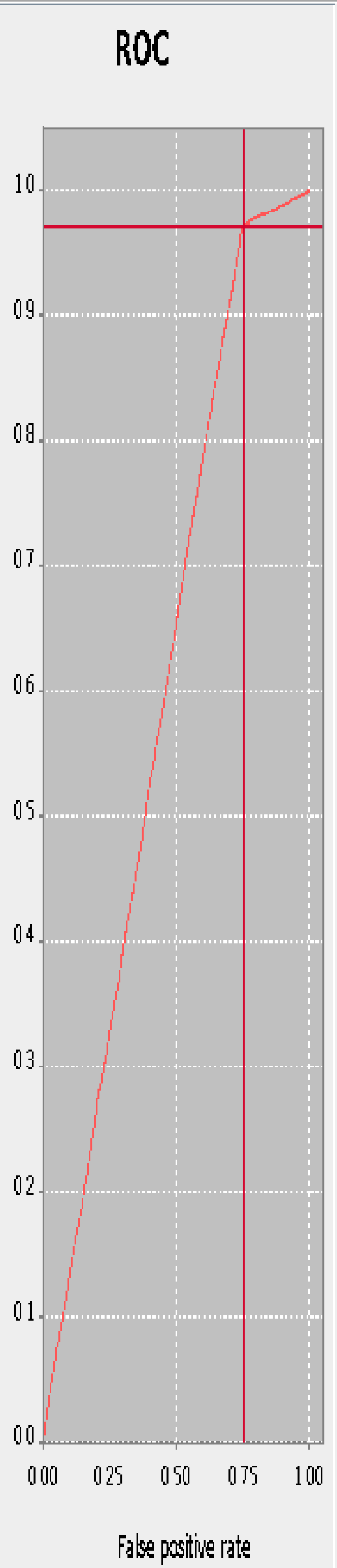


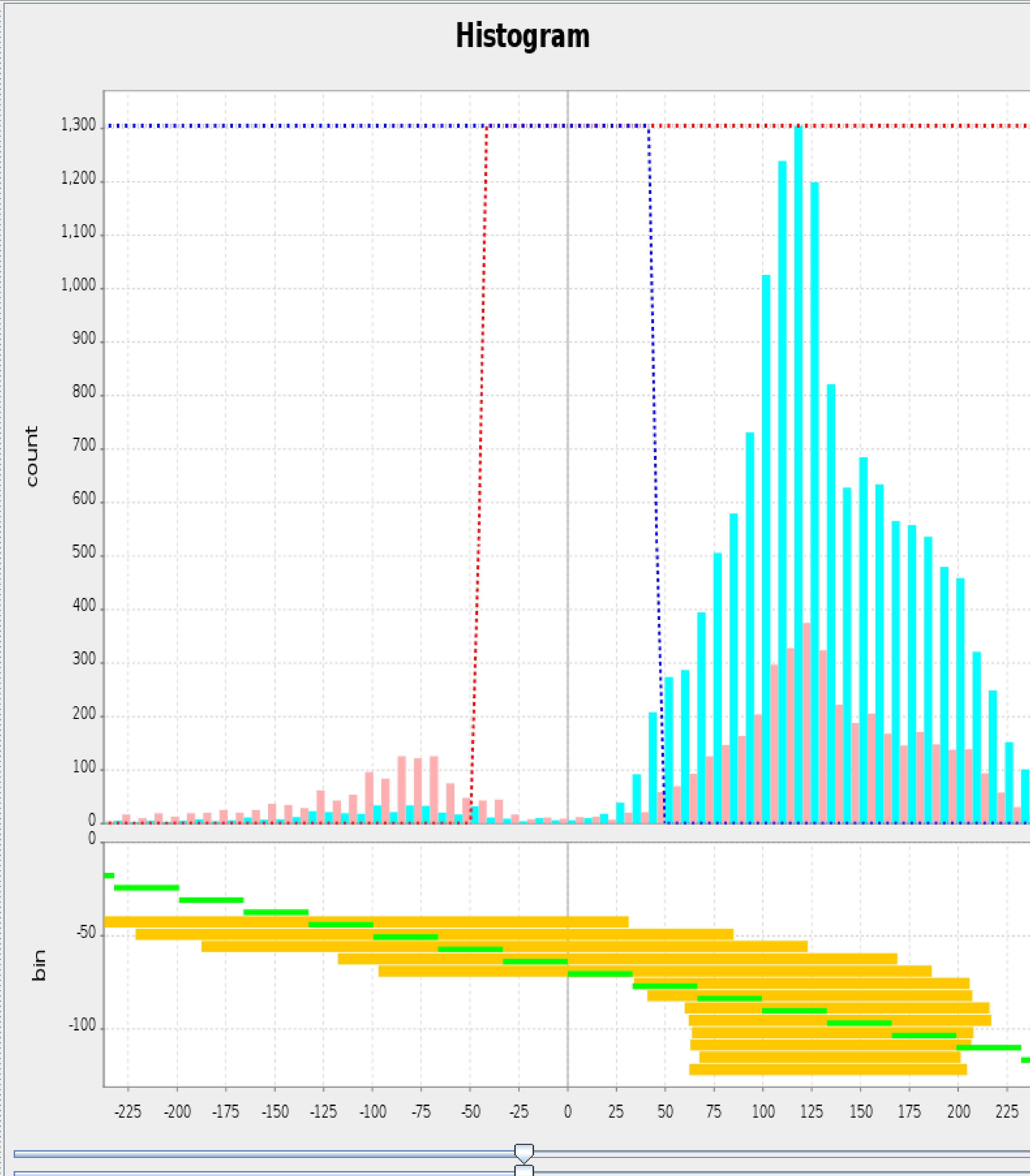
Iteration 222 [T=0.060+/-0.082
Iteration 223 [T=0.062+/-0.083
Iteration 224 [T=0.066+/-0.088
Iteration 225 [T=0.067+/-0.090
Iteration 226 [T=0.070+/-0.092
Iteration 227 [T=0.073+/-0.096
Iteration 228 [T=0.078+/-0.098
Iteration 229 [T=0.080+/-0.099
Iteration 230 [T=0.082+/-0.100
Iteration 231 [T=0.086+/-0.102
Iteration 232 [T=0.089+/-0.103
Iteration 233 [T=0.092+/-0.105
Iteration 234 [T=0.096+/-0.106
Iteration 235 [T=0.098+/-0.107
Iteration 236 [T=0.100+/-0.108
Iteration 237 [T=0.103+/-0.109
Iteration 238 [T=0.106+/-0.112
Iteration 239 [T=0.109+/-0.112
Iteration 240 [T=0.114+/-0.113
Iteration 241 [T=0.117+/-0.115
Iteration 242 [T=0.124+/-0.117
Iteration 243 [T=0.128+/-0.119
Iteration 244 [T=0.132+/-0.123
Iteration 245 [T=0.139+/-0.126
Iteration 246 [T=0.143+/-0.128
Iteration 247 [T=0.147+/-0.130
Iteration 248 [T=0.152+/-0.131
Iteration 249 [T=0.157+/-0.134
Iteration 250 [T=0.160+/-0.135





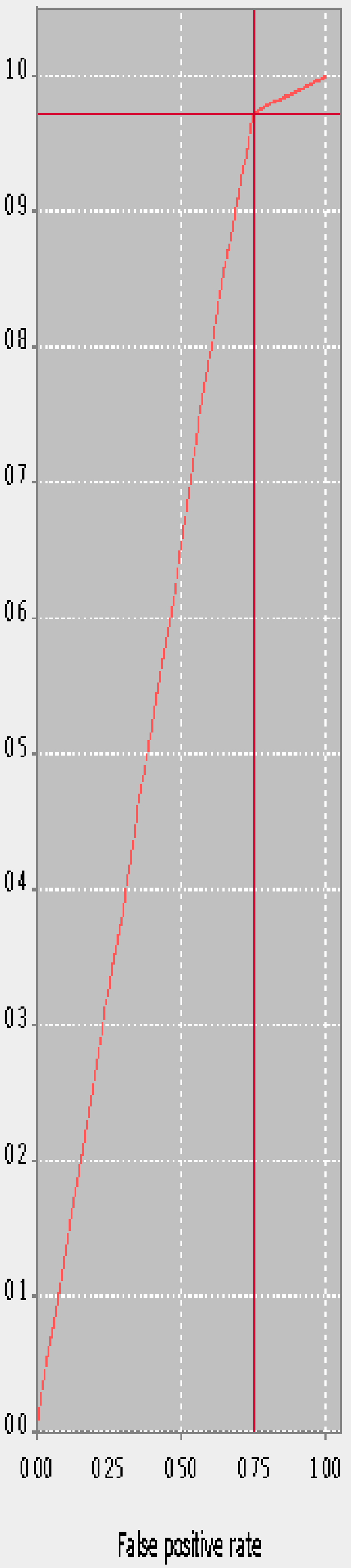
Iteration 272 [T=0.272+/-0.159
Iteration 273 [T=0.275+/-0.157
Iteration 274 [T=0.282+/-0.161
Iteration 275 [T=0.286+/-0.161
Iteration 276 [T=0.291+/-0.162
Iteration 277 [T=0.294+/-0.163
Iteration 278 [T=0.300+/-0.163
Iteration 279 [T=0.305+/-0.164
Iteration 280 [T=0.311+/-0.165
Iteration 281 [T=0.318+/-0.165
Iteration 282 [T=0.324+/-0.165
Iteration 283 [T=0.328+/-0.166
Iteration 284 [T=0.333+/-0.166
Iteration 285 [T=0.340+/-0.164
Iteration 286 [T=0.344+/-0.165
Iteration 287 [T=0.347+/-0.166
Iteration 288 [T=0.353+/-0.163
Iteration 289 [T=0.359+/-0.165
Iteration 290 [T=0.363+/-0.164
Iteration 291 [T=0.367+/-0.164
Iteration 292 [T=0.371+/-0.165
Iteration 293 [T=0.375+/-0.165
Iteration 294 [T=0.379+/-0.164
Iteration 295 [T=0.382+/-0.165
Iteration 296 [T=0.385+/-0.165
Iteration 297 [T=0.390+/-0.165
Iteration 298 [T=0.396+/-0.165
Iteration 299 [T=0.400+/-0.165
Iteration 300 [T=0.406+/-0.164



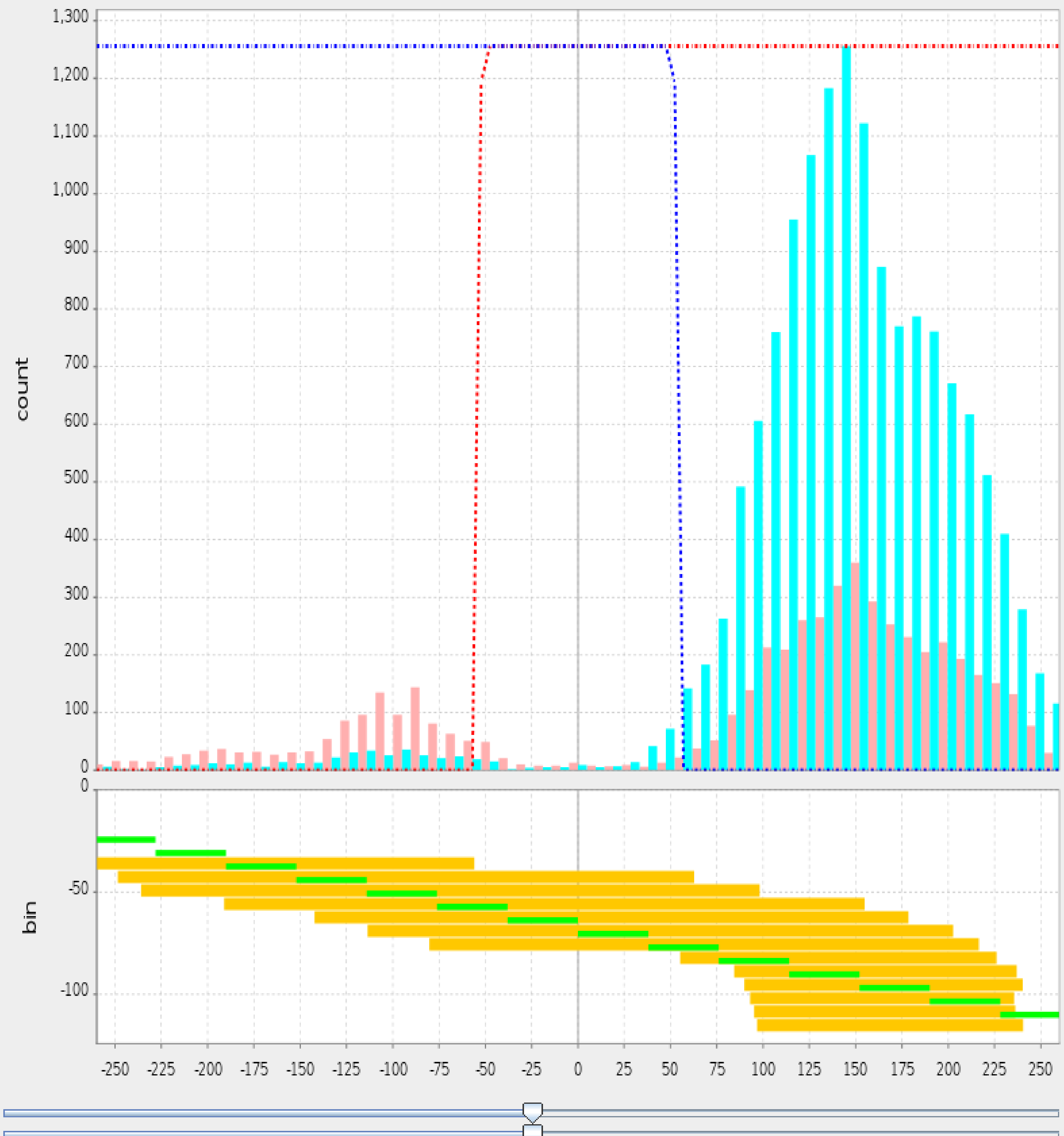


Iteration 322 [T=0.499+/-0.166
Iteration 323 [T=0.502+/-0.166
Iteration 324 [T=0.505+/-0.166
Iteration 325 [T=0.508+/-0.166
Iteration 326 [T=0.513+/-0.168
Iteration 327 [T=0.518+/-0.169
Iteration 328 [T=0.520+/-0.169
Iteration 329 [T=0.525+/-0.169
Iteration 330 [T=0.527+/-0.169
Iteration 331 [T=0.531+/-0.169
Iteration 332 [T=0.535+/-0.172
Iteration 333 [T=0.538+/-0.171
Iteration 334 [T=0.540+/-0.172
Iteration 335 [T=0.542+/-0.173
Iteration 336 [T=0.545+/-0.173
Iteration 337 [T=0.550+/-0.173
Iteration 338 [T=0.553+/-0.173
Iteration 339 [T=0.557+/-0.174
Iteration 340 [T=0.561+/-0.175
Iteration 341 [T=0.563+/-0.175
Iteration 342 [T=0.567+/-0.177
Iteration 343 [T=0.571+/-0.178
Iteration 344 [T=0.575+/-0.179
Iteration 345 [T=0.579+/-0.180
Iteration 346 [T=0.582+/-0.181
Iteration 347 [T=0.586+/-0.181
Iteration 348 [T=0.589+/-0.181
Iteration 349 [T=0.592+/-0.181
Iteration 350 [T=0.596+/-0.182

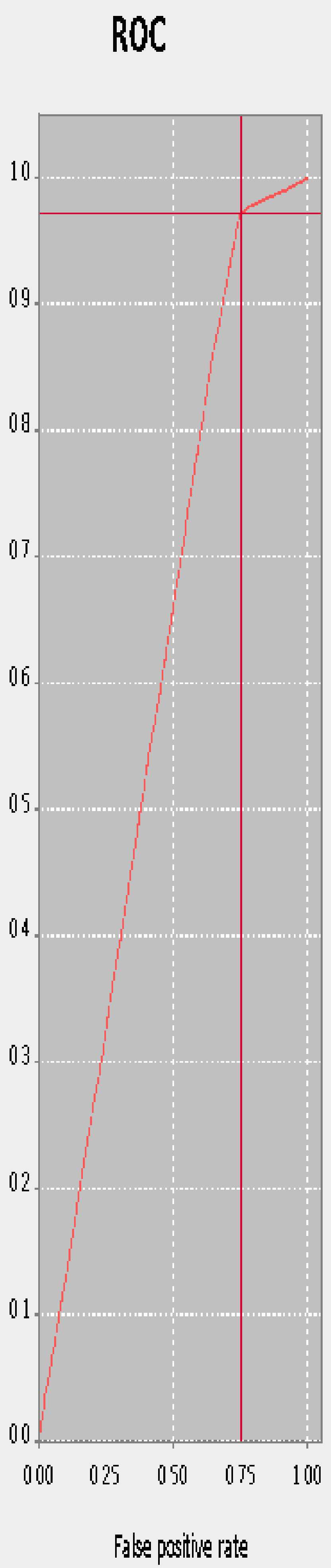
ROC

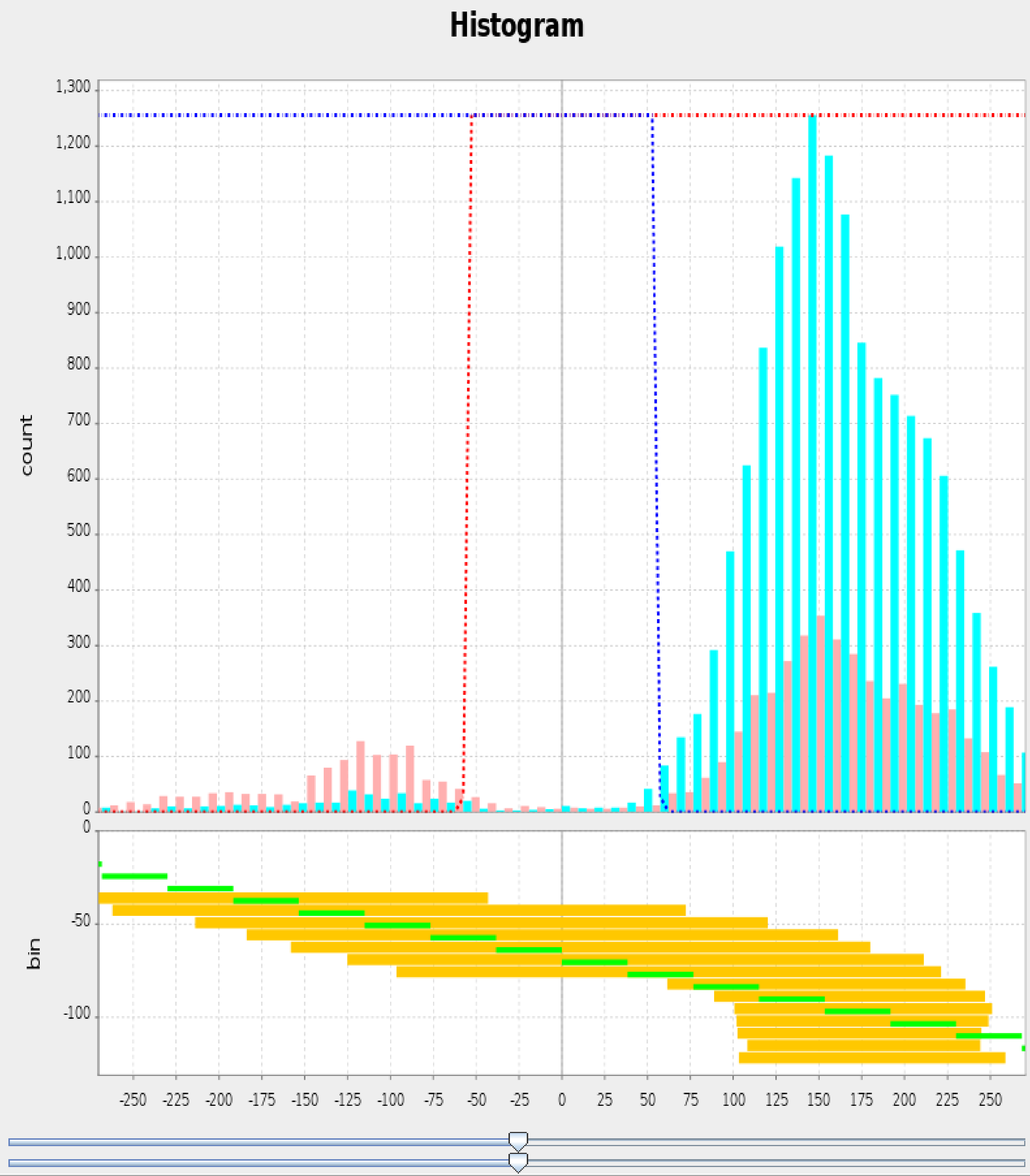


Histogram



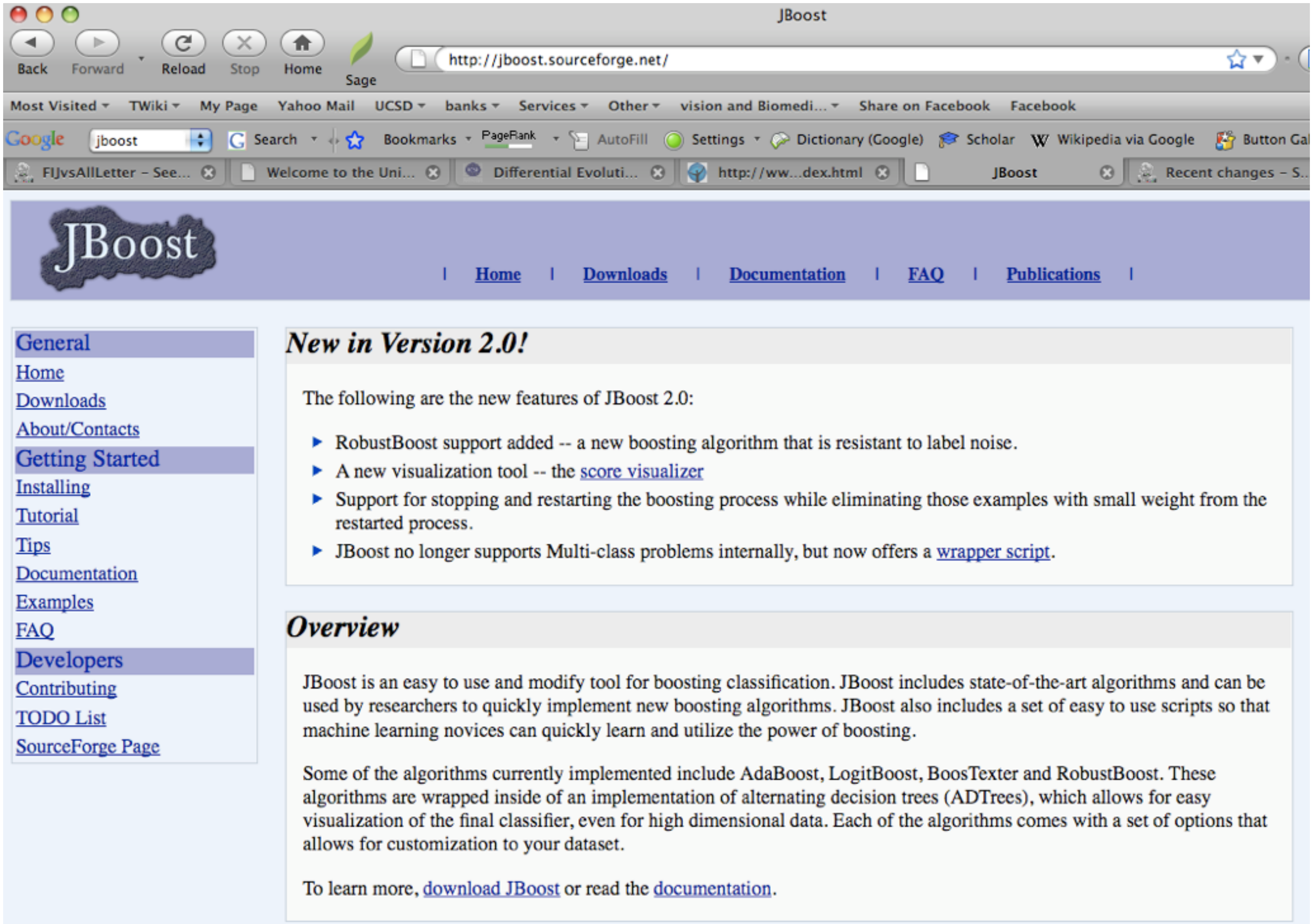
Iteration 372 [T=0.670+/-0.185
Iteration 373 [T=0.672+/-0.185
Iteration 374 [T=0.674+/-0.185
Iteration 375 [T=0.678+/-0.184
Iteration 376 [T=0.680+/-0.184
Iteration 377 [T=0.682+/-0.184
Iteration 378 [T=0.684+/-0.184
Iteration 379 [T=0.687+/-0.183
Iteration 380 [T=0.689+/-0.184
Iteration 381 [T=0.692+/-0.184
Iteration 382 [T=0.694+/-0.184
Iteration 383 [T=0.696+/-0.184
Iteration 384 [T=0.699+/-0.185
Iteration 385 [T=0.701+/-0.185
Iteration 386 [T=0.704+/-0.185
Iteration 387 [T=0.706+/-0.185
Iteration 388 [T=0.710+/-0.183
Iteration 389 [T=0.712+/-0.184
Iteration 390 [T=0.714+/-0.184
Iteration 391 [T=0.717+/-0.183
Iteration 392 [T=0.719+/-0.183
Iteration 393 [T=0.721+/-0.184
Iteration 394 [T=0.723+/-0.184
Iteration 395 [T=0.726+/-0.185
Iteration 396 [T=0.727+/-0.185
Iteration 397 [T=0.730+/-0.184
Iteration 398 [T=0.733+/-0.186
Iteration 399 [T=0.735+/-0.186
Iteration 400 [T=0.736+/-0.186





Iteration 403 [T=0.742+/-0.187
Iteration 404 [T=0.744+/-0.188
Iteration 405 [T=0.745+/-0.188
Iteration 406 [T=0.747+/-0.188
Iteration 407 [T=0.749+/-0.188
Iteration 408 [T=0.751+/-0.188
Iteration 409 [T=0.754+/-0.187
Iteration 410 [T=0.755+/-0.187
Iteration 411 [T=0.757+/-0.187
Iteration 412 [T=0.759+/-0.186
Iteration 413 [T=0.760+/-0.186
Iteration 414 [T=0.762+/-0.186
Iteration 415 [T=0.764+/-0.186
Iteration 416 [T=0.767+/-0.186
Iteration 417 [T=0.768+/-0.187
Iteration 418 [T=0.770+/-0.187
Iteration 419 [T=0.773+/-0.187
Iteration 420 [T=0.775+/-0.186
Iteration 421 [T=0.777+/-0.186
Iteration 422 [T=0.779+/-0.186
Iteration 423 [T=0.782+/-0.186
Iteration 424 [T=0.784+/-0.186
Iteration 425 [T=0.786+/-0.185
Iteration 426 [T=0.787+/-0.185
Iteration 427 [T=0.789+/-0.185
Iteration 428 [T=0.791+/-0.186
Iteration 429 [T=0.794+/-0.185
Iteration 430 [T=0.796+/-0.185
Iteration 431 [T=0.797+/-0.185

JBoost V2.0


JBoost
http://jboost.sourceforge.net/
JBoost
| Home | Downloads | Documentation | FAQ | Publications |
General
Home
Downloads
About/Contacts
Getting Started
Installing
Tutorial
Tips
Documentation
Examples
FAQ
Developers
Contributing
TODO List
SourceForge Page
New in Version 2.0!
The following are the new features of JBoost 2.0:
RobustBoost support added -- a new boosting algorithm that is resistant to label noise.
A new visualization tool -- the score visualizer
Support for stopping and restarting the boosting process while eliminating those examples with small weight from the restarted process.
JBoost no longer supports Multi-class problems internally, but now offers a wrapper script.
Overview
JBoost is an easy to use and modify tool for boosting classification. JBoost includes state-of-the-art algorithms and can be used by researchers to quickly implement new boosting algorithms. JBoost also includes a set of easy to use scripts so that machine learning novices can quickly learn and utilize the power of boosting.
Some of the algorithms currently implemented include AdaBoost, LogitBoost, BoosTexter and RobustBoost. These algorithms are wrapped inside of an implementation of alternating decision trees (ADTrees), which allows for easy visualization of the final classifier, even for high dimensional data. Each of the algorithms comes with a set of options that allows for customization to your dataset.
To learn more, download JBoost or read the documentation.