Item 236
git_comments:

1. * * Examples of using {@link ParametricVectorGenerator} for generating two dimensional data. * {@link ParametricVectorGenerator} allows to create surfaces in N-dinensional spaces where each * dimension depends on one parameter 't'. In such generator just one random producer is used, it * defines a set of values for parameter 't'.
2. * * Run example. * * @param args Args.
3. Example of butterfly-like shape.
4. Example of Archimedean spiral.
5. * Licensed to the Apache Software Foundation (ASF) under one or more * contributor license agreements. See the NOTICE file distributed with * this work for additional information regarding copyright ownership. * The ASF licenses this file to You under the Apache License, Version 2.0 * (the "License"); you may not use this file except in compliance with * the License. You may obtain a copy of the License at * * http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law or agreed to in writing, software * distributed under the License is distributed on an "AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License.
6. Example of heart shape.
7. 't' will be in [-50, 50] range
8. 't' will be in [-100, 100] range
9. * * Run example. * * @param args Args.
10. Example of regression dataset with base function $y(x) = |x^2 - 10|$.
11. * * Examples of using standard dataset generators. Standard dataset generator represents a * toy datasets that can be used for algorithms testing.
12. A set of nested rings where each ring represents a class.
13. Variance vector should be two dimensional because there are two dimensions.
14. * Licensed to the Apache Software Foundation (ASF) under one or more * contributor license agreements. See the NOTICE file distributed with * this work for additional information regarding copyright ownership. * The ASF licenses this file to You under the Apache License, Version 2.0 * (the "License"); you may not use this file except in compliance with * the License. You may obtain a copy of the License at * * http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law or agreed to in writing, software * distributed under the License is distributed on an "AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License.
15. Constructs a set of gaussians with different mean and variance values where each gaussian represents a unique class.
16. Examples of lineary separable classes, a set of uniform distributed points on plane that can be splitted on two classes by diagonal hyperplane. Each example represents a different margin - distance between points and diagonal hyperplane. If margin < 0 then points of different classes are mixed.
17. * * Run example. * * @param args Args.
18. Family that constructed by 45 degree rotation from previous family.
19. Combination of families where first family represents a complex distribution for first class and second family for second class.
20. * Licensed to the Apache Software Foundation (ASF) under one or more * contributor license agreements. See the NOTICE file distributed with * this work for additional information regarding copyright ownership. * The ASF licenses this file to You under the Apache License, Version 2.0 * (the "License"); you may not use this file except in compliance with * the License. You may obtain a copy of the License at * * http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law or agreed to in writing, software * distributed under the License is distributed on an "AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License.
21. * * Example of using distribution families. Each distribution from family represents a class. Distribution family * is a distribution hence such family can be used as element of hight-level family where this distribution will * represent one class. Such families helps to construct ditributions with complex shape.
22. Family of ring sectors.
23. * * Run example. * * @param args Args.
24. **comment:** * * Example of using primitive generators and combiners for generators.

label: code-design

25. Sum of generators where vectors from first generator are summed with corresponding vectors from second generator.
26. Vectors from ring-like distribution.
27. Vectors from uniform distribution in n-dimensional space.
28. Vectors from gaussian.
29. Vectors from distribution having filled circle shape.
30. Example of vector generator function noize.
31. Example of vector generator filtering.
32. * Licensed to the Apache Software Foundation (ASF) under one or more * contributor license agreements. See the NOTICE file distributed with * this work for additional information regarding copyright ownership. * The ASF licenses this file to You under the Apache License, Version 2.0 * (the "License"); you may not use this file except in compliance with * the License. You may obtain a copy of the License at * * http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law or agreed to in writing, software * distributed under the License is distributed on an "AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License.
33. Example of generators concatenation where each vector of first generator are concatenated with corresponding vector from second generator.
34. Example of using map function for vector generator.
35. Example of complex distribution with "axe" shape.
36. Vectors from ring's sector distribution.
37. Using of rotate for generator.
38. **comment:** * * <!-- Package description. --> * Examples for data stream generators.
    label: documentation
39. * Licensed to the Apache Software Foundation (ASF) under one or more * contributor license agreements. See the NOTICE file distributed with * this work for additional information regarding copyright ownership. * The ASF licenses this file to You under the Apache License, Version 2.0 * (the "License"); you may not use this file except in compliance with * the License. You may obtain a copy of the License at * * http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law or agreed to in writing, software * distributed under the License is distributed on an "AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License.
40. @NAME@
41. @DATA@
42. define the y axis
43. www.apache.org/licenses/LICENSE-2.0
44. draw x axis with labels and move to the bottom of the chart area
45. draw y axis with labels and move in from the size by the amount of padding
46. Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to You under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0 Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.
47. Constant.
48. * Licensed to the Apache Software Foundation (ASF) under one or more * contributor license agreements. See the NOTICE file distributed with * this work for additional information regarding copyright ownership. * The ASF licenses this file to You under the Apache License, Version 2.0 * (the "License"); you may not use this file except in compliance with * the License. You may obtain a copy of the License at * * http://www.apache.org/licenses/LICENSE-2.0 * * Unless required by applicable law or agreed to in writing, software * distributed under the License is distributed on an "AS IS" BASIS, * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. * See the License for the specific language governing permissions and * limitations under the License.
49. * * Test suite for all tests located in {@link org.apache.ignite.ml.util.generators} package.
50. * * Open browser and shows given dataset generator's data on two dimensional plane. Label of vectors * is used for ordinate representation. * * @param generator Datastream generator. * @param limit Count of

points that should be taken from dataset. * @param xIdx Index of dimension for abscissa representation. * @throws IOException
51. * * Open browser and shows given dataset generator's data on two dimensional plane. * * @param name Name of dataset for showing. * @param generator Datastream generator. * @param limit Count of points that should be taken from dataset. * @param xIdx Index of dimension for abscissa representation. * @param yIdx Index of dimension for ordinate representation. * @param isLabeled if isLabeled == true then colors will be used for separate different classes on plane. * @throws IOException
52. * * @param name Dataset name for showing. * @param values List of vectors are taken from dataset generator. * @param xIndex Index of abscissa in vector. * @param yGetter Getter of ordinate value from vector. * @param colorGetter Getter of collor for showing. * @throws IOException
53. * * Open browser and shows given dataset generator's data on two dimensional plane. Label of vectors * is used for ordinate representation. * * @param name Name of dataset for showing. * @param generator Datastream generator. * @param limit Count of points that should be taken from dataset. * @param xIdx Index of dimension for abscissa representation. * @throws IOException
54. * * Serialize xy-pair with vector to JSON representation. * * @param x X-value. * @param y Y-value. * @param clr Color.
55. * * Open browser and shows given dataset generator's data on two dimensional plane. * * @param generator Datastream generator. * @param limit Count of points that should be taken from dataset. * @param xIdx Index of dimension for abscissa representation. * @param yIdx Index of dimension for ordinate representation. * @param isLabeled if isLabeled == true then colors will be used for separate different classes on plane. * @throws IOException

git_commits:

1. **summary:** IGNITE-10793: [ML] Create comprehensive example for dataset generators
   message: IGNITE-10793: [ML] Create comprehensive example for dataset generators This closes #5999

github_issues:

github_issues_comments:

github_pulls:

1. **title:** IGNITE-10793: Examples for DataSteamGenerators
   body:
2. **title:** IGNITE-10793: Examples for DataSteamGenerators
   body:
3. **title:** IGNITE-10793: Examples for DataSteamGenerators
   body:
4. **title:** IGNITE-10793: Examples for DataSteamGenerators
   body:

github_pulls_comments:

github_pulls_reviews:

1. Does it make sense to put the library into local folder? Otherwise these examples won't work without internet connection.
2. just copy-paste from other tracers)

jira_issues:

1. **summary:** [ML] Create comprehensive example for dataset generators
   description: We need to create well documented example for generators.
2. **summary:** [ML] Create comprehensive example for dataset generators
   description: We need to create well documented example for generators.
   label: documentation
3. **summary:** [ML] Create comprehensive example for dataset generators
   description: We need to create well documented example for generators.

jira_issues_comments:

1. reviewed and merged