

Generating Data Sets with Known Hypervolume

All rights reserved, etc.

Mike Yukish, Ph.D.
Head, Manufacturing Product & Process Design Department
Applied Research Lab/Penn State University
may106@arl.psu.edu

```
makePointsBalancedDistributed[runs_, d_, distribution_] := Module[{
  currentCorner,
  corners,
  maxHVindex,
  paretos = {Table[0, {d}]},
  hvSum = 0,
  newCorners,
  newPareto,
  newHV,
  cornerList = {{Table[0, {d}], Table[1, {d}]},},
  futureParetoMaximums = {}
},
  For[i = 0, i < runs, i++,
    (* Identify the corner which has the highest hypervolume potential *)
    corners = cornerList[[All, 2]];
    futureParetoMaximums = Map[Tr[#, Times] &, corners];
    {{maxHVindex}} = Position[futureParetoMaximums, Max[futureParetoMaximums]];

    (* Operate against that corner *)
    currentCorner = cornerList[[maxHVindex]];
    (* delete the corner since it will be covered by new Pareto point *)
    cornerList = Delete[cornerList, maxHVindex];
    {newCorners, newPareto, newHV} =
      cornerBreederDistribution[currentCorner, distribution];

    (* update the tallies *)
    cornerList = Join[cornerList, newCorners];
    paretos = Append[paretos, newPareto];
    hvSum += newHV;
  ];
  {cornerList, paretos, hvSum}
];
```

```

cornerBreederDistribution[innerCorner_, distribution_] :=
Module[{location, edges, edgeSegments, newPointLocations,
  newEdgesForEachPt, ans, newCornerList, paretoPoint, hv},
  {location, edges} = innerCorner;

  (* set the length of segments along current edge,
  randomized as per distribution *)
  edgeSegments = Map[RandomVariate[distribution] × # &, edges];
  (* can now update the location of the new points *)
  newPointLocations = MapIndexed[(ans = location;
    ans[[#2[[1]]] += #1;
    ans) &, edgeSegments];
  (* each point now has edges related to it *)
  newEdgesForEachPt = MapIndexed[(ans = edgeSegments;
    ans[[#2[[1]]] = #1 - ans[[#2[[1]]];
    ans) &, edges];
  newCornerList = MapThread[{#1, #2} &, {newPointLocations, newEdgesForEachPt}, 1];
  paretoPoint = edgeSegments + location;
  hv = Times @@ edgeSegments;
  {newCornerList, paretoPoint, hv}
];

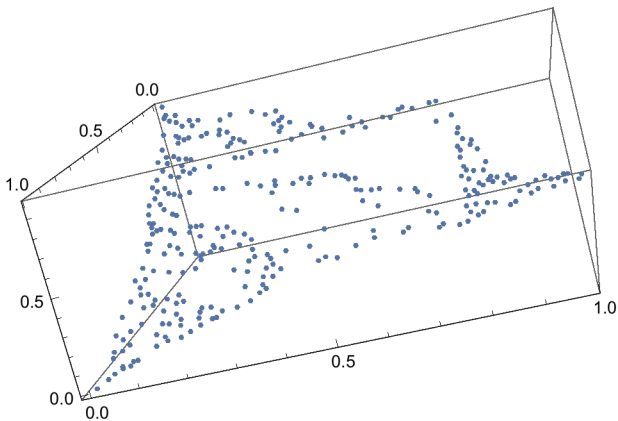
```

Test the function

```
res = makePointsBalancedDistributed[300, 3, BetaDistribution[2, 2]];
```

pareto points

```
ListPointPlot3D[res[[2]]]
```



hypervolume of the data set

```
res[[3]]
```

```
0.172848
```

Dynamic View of Hypervolumes

```

Manipulate[ (
  SeedRandom[1];
  res = makePointsBalancedDistributed[n, 3, BetaDistribution[ab[[1], ab[[2]]];
  GraphicsRow[ {
    Plot[PDF[BetaDistribution[ab[[1], ab[[2]]], x], {x, 0, 1}],
    Show[
      Graphics3D[Map[Cuboid[{0, 0, 0}, #] &, res[[2]]], ViewPoint -> {Pi, Pi/2, 2}]
      , Text[" Hypervolume " res[[3]]]]],
    {{n, 10, "Num pts"}, 1, 30},
    {{ab, {3, 3}}, {0.1, .1}, {10, 10}}
  ]

```

