This file computes and generates Figure 4 in the paper.

Define the CDF and PDF of utility distributions of agents A and B:

```
In[1]:= cdfa[x_] := Piecewise[
          {{x * 2, 0 ≤ x ≤ 1 / 4}, {1 / 2 + 2 / 3 * (x - 1 / 4), 1 / 4 ≤ x ≤ 1}, {1, x ≥ 1}}, 0];
        cdfb[x_] := Piecewise[
          {{2 / 3 * x, 0 ≤ x ≤ 3 / 4}, {1 / 2 + 2 * (x - 3 / 4), 3 / 4 ≤ x ≤ 1}, {1, x ≥ 1}}, 0];
```

Define their PDFs:

```
In[3]:= pdfa[x_] = D[cdfa[y], y] /. {y → x}
        pdfb[x_] = D[cdfb[y], y] /. {y → x}
```

Out[3]= $\begin{cases} 0 & x < 0 \\ 2 & 0 < x < \frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} < x < 1 \\ 0 & x > 1 \\ \text{Indeterminate} & \text{True} \end{cases}$

Out[4]= $\begin{cases} 0 & x < 0 \\ \frac{2}{3} & 0 < x < \frac{3}{4} \\ 2 & \frac{3}{4} < x < 1 \\ 0 & x > 1 \\ \text{Indeterminate} & \text{True} \end{cases}$

Next, we solve for the quantile functions (i.e., inverses of the CDFs), which will allow us to sample from the distributions using inverse transform sampling

```
In[5]:= Solve[y == cdfa[x], x, Reals]
        Solve[y == cdfb[x], x, Reals]
```

Out[5]= $\left\{\left\{x \to \boxed{\frac{y}{2} \text{ if } 0 < y < \frac{1}{2}}\right\}, \left\{x \to \boxed{\frac{1}{2}(-1 + 3y) \text{ if } \frac{1}{2} < y < 1}\right\}\right\}$

Out[6]= $\left\{\left\{x \to \boxed{\frac{3y}{2} \text{ if } 0 < y < \frac{1}{2}}\right\}, \left\{x \to \boxed{\frac{1+y}{2} \text{ if } \frac{1}{2} < y < 1}\right\}\right\}$

```
In[7]:= icdfa[y_] =
         Piecewise[{{y / 2, 0 ≤ y ≤ 1 / 2}, {1 / 2 * (3 y - 1), 1 / 2 ≤ y ≤ 1}}, Indeterminate]
        icdfb[y_] =
         Piecewise[{{(3 y) / 2, 0 ≤ y ≤ 1 / 2}, {1 / 2 * (1 + y), 1 / 2 ≤ y ≤ 1}}, Indeterminate]
```

Out[7]= $\begin{cases} \frac{y}{2} & 0 \le y \le \frac{1}{2} \\ \frac{1}{2}(-1 + 3y) & \frac{1}{2} \le y \le 1 \\ \text{Indeterminate} & \text{True} \end{cases}$

Out[8]= $\begin{cases} \frac{3y}{2} & 0 \le y \le \frac{1}{2} \\ \frac{1+y}{2} & \frac{1}{2} \le y \le 1 \\ \text{Indeterminate} & \text{True} \end{cases}$

Verify that the quantile functions and CDFs match:

```
In[9]:= Reduce[icdfa[cdfa[x]] == x && 0 ≤ x ≤ 1, x]
        Simplify[Reduce[cdfa[icdfa[x]] == x && 0 ≤ x ≤ 1, x]]
        Reduce[icdfb[cdfb[x]] == x && 0 ≤ x ≤ 1, x]
        Simplify[Reduce[cdfb[icdfb[x]] == x && 0 ≤ x ≤ 1, x]]
```

Out[9]= $x == \frac{1}{4} \;||\; 0 \le x \le \frac{1}{4} \;||\; \frac{1}{4} < x \le 1$

Out[10]= $0 \le x \le 1$

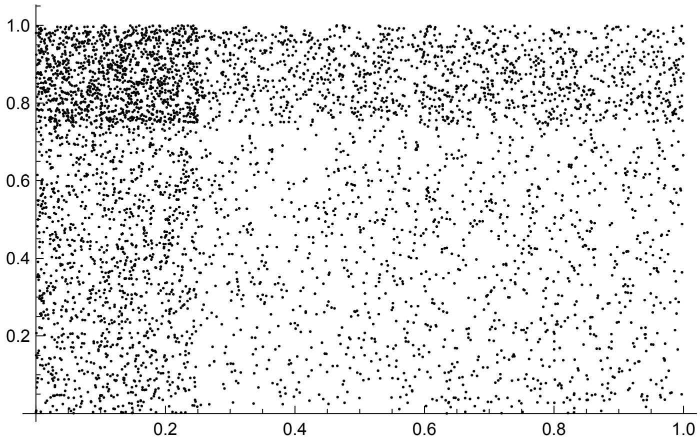
Out[11]= $x == \frac{3}{4} \;||\; 0 \le x \le \frac{3}{4} \;||\; \frac{3}{4} < x \le 1$

Out[12]= $0 \le x \le 1$

Draw the utilities of 5000 random items:

```
In[13]:= SeedRandom[0];
         random = Table[{icdfa[RandomReal[]], icdfb[RandomReal[]]}, {n, 5000}];
         itemutils = ListPlot[random, PlotStyle → Black]
```

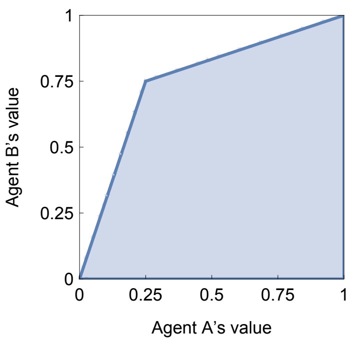
Out[15]=



Plot the region in which the maximum-percentile algorithm allocates to agent a: whenever CDF_A(utility_A) is larger than CDF_B(utility_B):

```
In[16]:= percentileregion = RegionPlot[cdfa[ua] ≥ cdfb[ub], {ua, 0, 1},
           {ub, 0, 1}, Frame → True, PlotRangePadding → 0, ImageSize → Small,
           FrameLabel → {"Agent A’s value", "Agent B’s value"},
           FrameTicks → {{{0, 0.25, 0.5, 0.75, 1}, None}, {{0, 0.25, 0.5, 0.75, 1}, None}},
           PlotStyle → RGBColor[{0.8105251, 0.8520337, 0.9129394}]]
```

Out[16]=



Now, we compute the equalizing multipliers. Since there are only two agents, we can assume that agent B’s multiplier is one and only have to find a multiplier m for agent A such that agent A’s probability of receiving an item is 1/2 (which immediately implies that agent B's probability is also 1/2):

In[17]:=
```
pra[m_] = Integrate[pdfa[x] * cdfb[m * x], {x, 0, 1}]
```

Out[17]=
$$\begin{cases} \frac{m}{4} & 0 \le m \le \frac{3}{4} \\ \frac{-5+4\,m}{4\,m} & m > 4 \\ \frac{1+4\,m+m^2}{12\,m} & 3 < m < 4 \\ \frac{-13+16\,m+m^2}{24\,m} & m == 3 \\ \frac{-15+24\,m+m^2}{36\,m} & 1 < m < 3 \\ \frac{2+4\,m+3\,m^2}{24\,m} & m == 4 \\ \frac{9-24\,m+25\,m^2}{36\,m} & \frac{3}{4} < m \le 1 \\ 0 & \text{True} \end{cases}$$

In[18]:=
```
Solve[pra[m] == 1 / 2, m, Reals]
N[%]
```

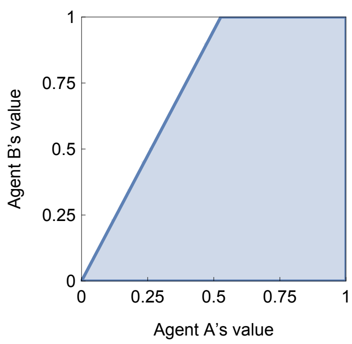
Out[18]= $\{\{m \to -3 + 2\sqrt{6}\}\}$

Out[19]= {{m → 1.89898}}

We can now plot the region in which the multiplier algorithm allocates items to agent A, that is, where ma times agent A's utility is larger than agent B's utility:

In[20]:=
```
multiplierregion = RegionPlot[(-3 + 2 √6) * ua ≥ ub, {ua, 0, 1},
  {ub, 0, 1}, Frame → True, PlotRangePadding → 0, ImageSize → Small,
  FrameLabel → {"Agent A's value", "Agent B's value"},
  FrameTicks → {{{0, 0.25, 0.5, 0.75, 1}, None}, {{0, 0.25, 0.5, 0.75, 1}, None}},
  PlotStyle → RGBColor[{0.8105251, 0.8520337, 0.9129394}]]
```

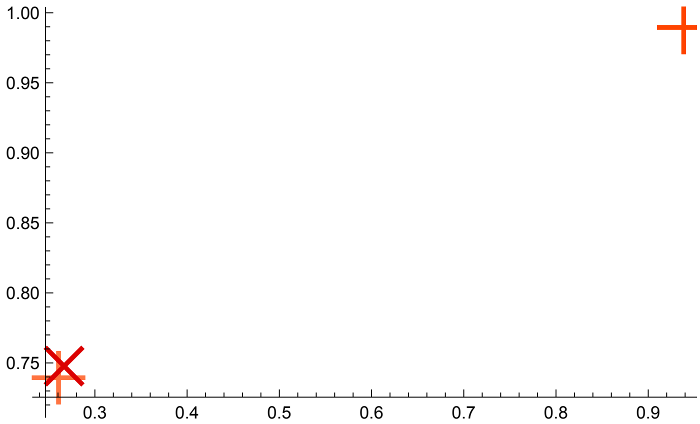
Out[20]=



Finally, for the maximum-percentile figure, we want to highlight three items: two items given to agent A and with utilities as close to the median for both agents, and one item given to agent B with utilities at close to the top percentile for both agents:

In[21]:=
```
randomto1 = Select[random, cdfa[#[[1]]] ≥ cdfb[#[[2]]] &];
randomto1s = SortBy[randomto1, (cdfa[#[[1]]] - 1 / 2) ^2 + (cdfb[#[[2]]] - 1 / 2) ^2 &];
randomto2 = Select[random, cdfa[#[[1]]] < cdfb[#[[2]]] &];
randomto2s = SortBy[randomto2, (cdfa[#[[1]]] - 1) ^2 + (cdfb[#[[2]]] - 1) ^2 &];
cross =
  Graphics[{Thickness[.09], Line[{{-1, 0}, {1, 0}}], Line[{{0, -1}, {0, 1}}]}];
markers = ListPlot[{#} & /@ Append[Take[randomto1s, 2], randomto2s[[1]]],
  PlotMarkers → {{cross, .13}, {Rotate[cross, 45 Degree], .13}, {cross, .13}},
  PlotStyle → {ColorData[2][3], ColorData[2][1], ColorData[2][2]}]
```

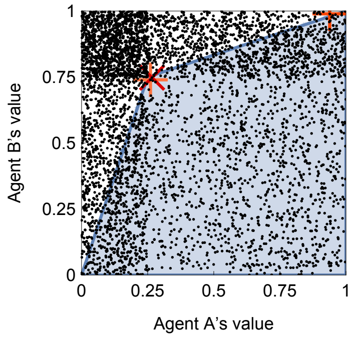
Out[26]=



Export the maximum-percentile panel of the figure to the current directory:

In[27]:=
```
SetDirectory@NotebookDirectory[];
percentileplot = Show[{percentileregion, markers, itemutils}]
Export["percentile_plot1.pdf", percentileplot];
```

Out[28]=



Export the multiplier panel of the figure:

In[30]:=
```
multiplierplot = Show[{multiplierregion, itemutils}]
Export["percentile_plot2.pdf", multiplierplot];
```

Out[30]=