Algorithm 1 Experimental as Market (EXAM)

Input: n the number of subjects, m the number of treatments, $(c_t) \in \mathbb{N}$ treatment t's pseudo capacity with $\sum_t c_t = n$, (w_{it}) subject i's WTP for treatment t, (e_{ti}) treatment t's predicted treatment effect for subject i, b the budget constraint

Output: (p^*_{it}) treatment t's assignment probability for subject i, (α^*, β^*_t) parameters determining treatment t's equilibrium price of the form $\pi^*_{te} = \alpha^* e + \beta^*_t$, error_{min} minimized market clearing error relative the total capacity of treatments

function InitialAlpha( )
    $\alpha \leftarrow$ generate random number $\sim \text{Uniform}(-b, 0)$ ▷ set the value of α
    return α

function InitBeta( )
    $\beta_{t_0} \leftarrow 0$
    for each $t = t_1, ..., t_m$ **do**
        $\beta_t \leftarrow$ generate random number $\sim \text{Uniform}(-b, b)$ ▷ set the initial value of β_t
    return (β_t) ▷ return an m-dimensional vector

function Price(α, (β_t)) ▷ get the price of treatment t
    for each $i, t = t_1, ..., t_m$ **do**
        $\pi_{te_{ti}} = \alpha e_{ti} + \beta_t$
    return $(\pi_{te_{ti}})$ ▷ return the $n \times m$ price matrix

function Demand($(\pi_{te_{ti}})$) ▷ get subject i's demand for treatment t
    for each i **do** ▷ perform utility maximization for each subject i
        $(p_{it})_t \leftarrow \arg\max_{(p_{it})_t \in P} \sum_t w_{it} p_{it}$ s.t. $\sum_t \pi_{te_{ti}} p_{it} \leq b$
    return (p_{it}) ▷ return the $n \times m$ demand matrix

function ExcessDemand((p_{it})) ▷ get the excess demand for treatment t
    for each $t = t_1, ..., t_m$ **do**
        $d_t \leftarrow \sum_i p_{it} - c_t$
    return (d_t) ▷ return the m-dimensional excess demand vector

function ClearingError((d_t)) ▷ get the market clearing error
    if $d_t < 0$ for all t **then**
        return 0
    else
        error $\leftarrow \sqrt{\sum_t d_t^2} / \sum_t c_t$
        return error ▷ return the market clearing error

$\delta_\beta \leftarrow b/50$ ▷ scaling factor for β_t's to set new prices

function BetaNew((β_t, d_t)) ▷ recalibrate β_t's to set new prices
  for each $t = t_1, ..., t_m$ **do**
    $\beta_t^{new} \leftarrow \beta_t + d_t \delta_\beta$
  return (β_t^{new})

function ClearMarket( ) ▷ the main function
  $\alpha \leftarrow$ InitialAlpha( )
  $(\beta_t) \leftarrow$ InitBeta( )
  $(\pi_{te_{ti}}) \leftarrow$ Price($\alpha, (\beta_t)$)
  $(p_{it}) \leftarrow$ Demand($(\pi_{te_{ti}})$)
  $(d_t) \leftarrow$ ExcessDemand((p_{it}))
  error $\leftarrow$ ClearingError((d_t))
  error$_{min} \leftarrow$ error ▷ initialize the minimum of clearing error
  ClearingThreshold $\leftarrow 0.01$ ▷ threshold for the market clearing error
  IterationThreshold $\leftarrow 10$ ▷ threshold for iteration times
  iterations $\leftarrow 0$ ▷ initialize iteration time count
  while True **do**
  if iterations > IterationThreshold **then**
    $\alpha \leftarrow$ InitialAlpha( ) ▷ start new equilibrium research
    $(\beta_t) \leftarrow$ InitBeta( )
    iterations $\leftarrow 0$
  else
    $(\beta_t) \leftarrow$ BetaNew($(\beta_t), (d_t)$)
  $(\pi_{te_{ti}}) \leftarrow$ Price($\alpha, (\beta_t)$)
  $(p_{it}) \leftarrow$ Demand($(\pi_{te_{ti}})$)
  $(d_t) \leftarrow$ ExcessDemand((p_{it}))
  error $\leftarrow$ ClearingError((d_t))
  if error < error$_{min}$ **then**
    error$_{min} \leftarrow$ error
    $\alpha^* \leftarrow \alpha$ ▷ the new prices reduce the error
    $(\beta_t^*) \leftarrow (\beta_t)$
    $(p_{it}^*) \leftarrow (p_{it})$
  if error$_{min}$ < ClearingThreshold **then**
    break
  iterations += 1
  return $((p_{it}^*), \alpha^*, (\beta_t^*)$, error$_{min})$ ▷ return the outputs