Lista # 2 - Escolha dinâmica discreta

Nessa lista vamos estimar um modelo dinâmico de escolha discreta e usar nossos resultados para fazer simulações contrafactuais. O modelo é inspirado no modelo clássico de reposição de máquinas apresentado em Rust (1987). Todas as dicas se referem a implementação em Matlab, mas serão úteis também em outras implementações. Fiquem a vontade para usar R ou Python se preferirem. Além da solução por escrito, vocês devem também entregar os códigos.

Temos uma firma que atua em N mercados distintos de operação independente, indexados $n = 1, \ldots, N$. Para produzir seu produto em cada mercado a firma utiliza uma máquina que se deprecia rapidamente e precisa ser reposta, paralisando as operações da firma no mercado. No ínicio de cada período a firma precisa decidir se troca a máquina. Caso a firma decida não trocar a máquina no início do período t, o lucro da firma é dado por

$$\theta x_n(1 - \alpha a_{nt}) + \varepsilon_{0nt},$$

em que x_n é uma medida do tamanho do mercado n, a_{nt} é a idade da máquina da firma no mercado n no período t e ε_{0nt} é um choque.

Caso a firma decida trocar a máquina no período t seu lucro é somente $-\phi + \varepsilon_{1nt}$, em que ϕ é o custo de reposição da máquina e ε_{1nt} é também um choque. Suponha que ε_{0nt} e ε_{1nt} são iid e com distribuição valor extremo do tipo 1.

Quando a firma troca sua máquina em t, a idade da sua máquina no período seguinte é 0, ou seja, $a_{n,t+1} = 0$. Caso não haja troca, a máquina envelhece no período seguinte até um limite máximo de 10, mais precisamente, $a_{n,t+1} = \min\{a_{n,t} + 1, 10\}$. Firmas maximizam em suas decisões o valor presente esperado do fluxo de lucros descontado $\mathbb{E}\sum_{t=0}^{\infty} \rho^t \pi_{nt}$. Nesse exercício, assumimos que conhecemos a taxa de desconto verdadeira: $\rho = 0.9$.

Vocês tem acesso a uma base de dados em que cada linha representa um mercado n em um período t, com as seguintes variáveis:

- Market: indicador de mercado (n);
- MarketSize: tamando do mercado (x_n);
- t: indicador de período (t);
- age: idade da máquina (a_{nt}).

Parte I: *Setup*

1. Equações de Bellman.
   (a) Escreva as equações de Bellman que representam o problema dinâmico da firma na sua versão completa como função de todas as variáveis de estado, ou seja, $V(x_n, a_{nt}, \varepsilon_{0nt}, \varepsilon_{1nt})$.

(b) Escreva as equações de Bellman no formato integrado, ou seja, apenas como função das variáveis de estado x_n e a_{nt}: $V(x_n, a_{nt})$.

(c) Escreva a fórmula para as *Conditional Choice Probabilities* (CCPs) de se escolher trocar a máquina no período t no mercado n como função das variáveis de estado e da função valor encontrada no item anterior.

2. Crie uma nova variável, c_{nt}, que represente a escolha feita no período t. Ou seja, $c_{nt} = 1$, caso a troca tenha sido feita no período t e $c_{nt} = 0$, caso contrário.
Dica: Notem que não será possível criar a variável c_{nt} para todos os períodos. Observações em que não é possível conhecer c_{nt} não serão usadas em nossa análise.

3. Estime um modelo de probabilidade linear para a decisão de troca da máquina:

$$c_{nt} = \beta_0 + \beta_1 x_n + \beta_2 a_{nt} + v_{nt}.$$

Reporte o vetor estimado $\hat{\beta}$ e seus erros-padrão. Interprete o resultado. Os sinais das estimativas fazem sentido econômico?

Parte II: Estimação

Nessa parte vamos estimar o vetor de parâmetros estruturais do modelo (θ, α, ϕ) por *Maximum Likelihood* (ML) utilizando o algorítmo de *Nested Fixed Point* (NFXP). A ideia é que para cada avaliação da função objetivo de ML ao longo da otimização, precisaremos encontrar as funções valores por iteração, construir as CCPs e avaliar a verossimilhança na amostra.

1. Crie uma função que compute a *log-likelihood* tendo como entradas os dados e o vetor de parâmetros desconhecidos. Por exemplo, $LogLikelihood(Par, Data)$.

(a) Para o vetor de parâmetros dados na entrada, precisaremos primeiro computar a função valor $V(x, a)$ por iteração e depois $CCP(x, a)$ utilizando a expressão encontrada na Parte I.
Dicas: Comecem criando um grid para todos os valores possíveis de x (note que nos dados x assume um número relativamente pequeno de valores) e outro para a. Como temos duas variáveis de estado, é mais fácil fazer a iteração utilizando uma orientação matricial para V, e consequentemente CCP, mas não é necessário. Escrevam um while loop que computa a função valor para todos os pontos dos grids utilizando o método da contração. Fiquem atentos ao problema quando $a = 10$. Após a convergência da contração, compute as CCPs.

(b) O próximo passo é usar as CCPs encontradas para computar a *likelihood* das observações.
Dicas: Aqui vocês precisam encontrar para cada observação a entrada correspondente na matriz (ou vetor) CCP. A maneira mais fácil e rápida é criando uma nova variável, vamos chamá-la de 'state', na base de dados (isso deve ser feito fora da função LogLikelihood) que informa o índice daquela obsevação na matriz (ou vetor) V e CCP. Por exemplo, se uma observação tem $x = 1$ e $a = 7$, temos que

procurar qual a entrada na matriz CCP correspondente. Suponha que a entrada correspondente a $x = 1$ e $a = 7$ em CCP seja a 107. Então a variável 'state' para essa observação deve informar o número 107. Com essa variável definida, se chamamos CCP(state) obtemos o vetor de CCPs para cada observação na nossa amostra.

2. Utilize uma função pronta (por exemplo, *fminunc* no Matlab) com otimização baseada em gradiente para encontrar o vetor de parâmetros que maximiza a função objetivo $LogLikelihood(Par, Data)$.
   Dica: Você precisará informar um vetor inicial para busca. Utilize $(\theta, \alpha, \phi) = (1, 0.1, 1)$. *Qualquer vetor numa região razoável deve funcionar.*

Parte III: Contrafactual

Apenas como um exemplo, vamos avaliar como as políticas da firma mudam depois de uma alteração do custo de reposição.

1. Utlizando os parâmetros estimados, compute as CCPs. Esse será o cenário 'baseline'.

2. Agora, reduza o custo de reposição em 50%. Recompute as CCPs.

3. Fixe $x = 1$ e reporte o vetor de CCPs para os dois cenários. Isto é, em cada cenário haverá uma CCP para cada idade possível da máquina. Compare os dois cenários.