K-means clustering and Bonhomme, Lamadon, Manresa (2019)

Nicolò Russo

Outline

1. Introduction to clustering
2. K-means clustering
3. MATLAB implementation of K-means
4. "Discretizing unobserved heterogeneity" by Bonhomme, Lamadon, Manresa (2019)

Definition and example

- **Def.**: separation of the data into groups (**clusters**) based on patterns in the data
- Not prediction, but understanding of the data
- **Example**: market segmentation

Supervised learning

- Process:
  1. Teach the machine using labeled **training data**
  2. Provide the machine with new unlabeled data
  3. Algorithm analyzes new data and produces the correct outcome
- Example:
  1. Classification

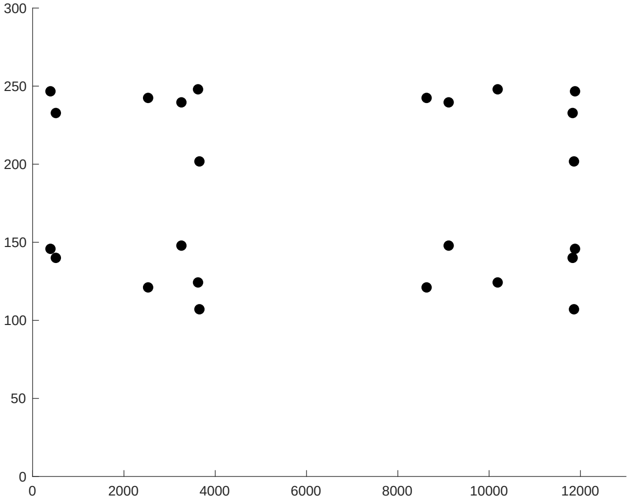
Unsupervised learning

- Process:
  1. Provide the machine with **unlabeled** data
  2. Algorithm acts on information without guidance
  3. Algorithm groups data according to similarities, patterns, and differences
- Examples:
  1. Clustering
  2. Association

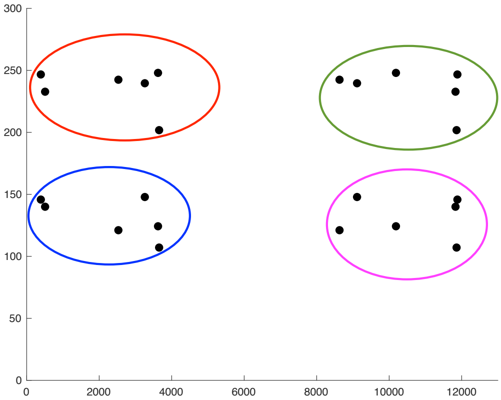
Properties of clusters

1. All the data points in a cluster should be similar to each other
2. Data points in different clusters should be as different as possible

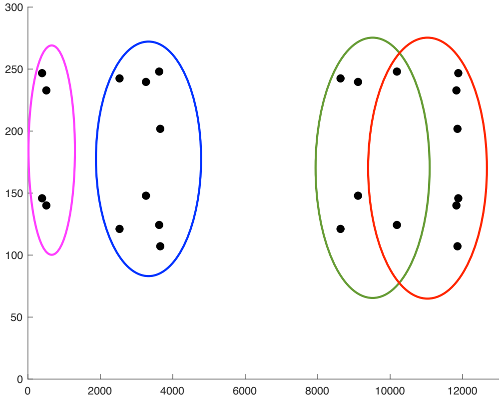
Properties of clusters



Properties of clusters



(a) Appropriate clustering

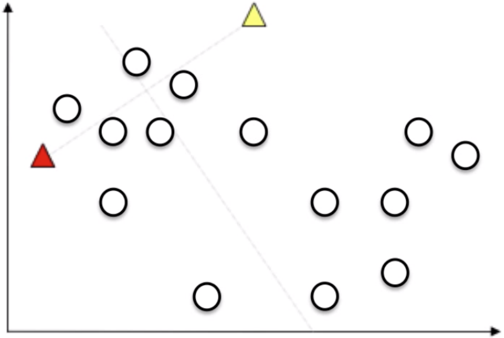


(b) Inappropriate clustering

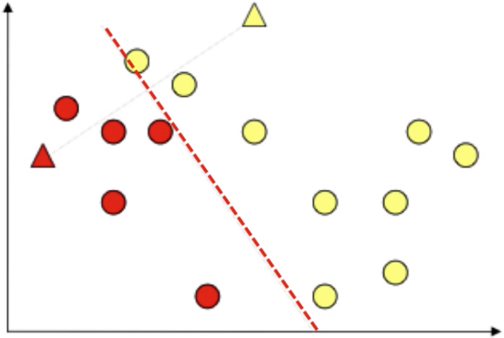
K-means overview

- Clusters the data in a pre-specified number of subpopulations (K)
- Polythetic and hard clustering method
- Associates each cluster to a **centroid** (a prototypical instance in the data)
- Algorithm:
  1. Compares distance between data points and centroids
  2. Assigns each data point to a specific cluster

K-means visually



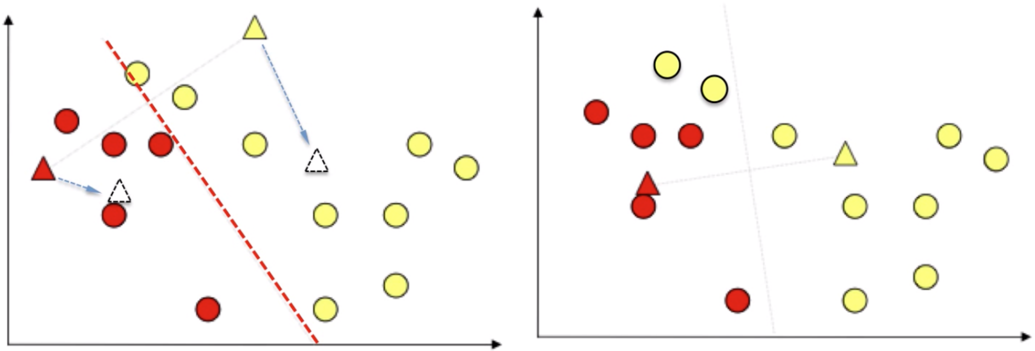
Data set



1st iteration - 1st half

Source: *Victor Lavrenko, University of Edinburgh*

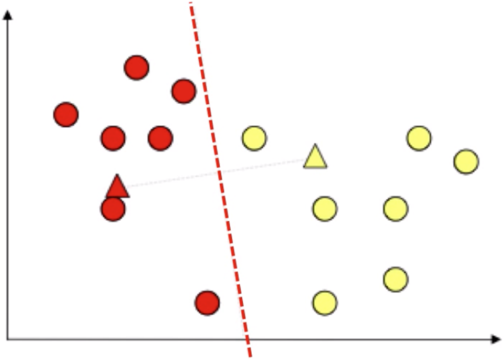
K-means visually



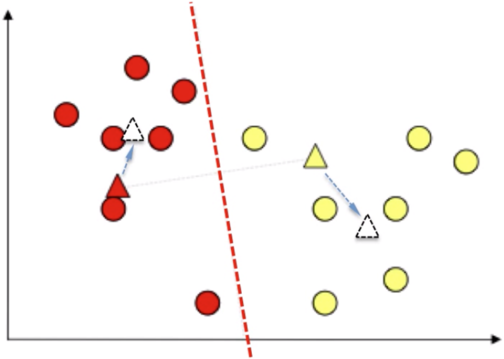
1st iteration - 2nd half Old assignment - new centroids

Source: Victor Lavrenko, University of Edinburgh

K-means visually



2nd iteration - 1st half



2nd iteration - 2nd half

<u>Source:</u> Victor Lavrenko, University of Edinburgh

K-means algorithm

Inputs: K; data points $x_1, \ldots, x_n$ where x_i is a vector

1. Place K centroids $c_1, \ldots, c_K$ at **random** location.
2. Repeat until convergence:
   2.1 For each point x_i:
      2.1.1 Find nearest centroid c_j using

$$\arg\min_j D(x_i, c_j)$$

      2.1.2 Assign the point x_i to cluster j
   2.2 For each cluster $j = 1, \ldots, K$:
      - Compute centroids as

$$c_j(a) = \frac{1}{n_j} \sum_{x_i \to c_j} x_i(a), \quad \forall a = 1, \ldots, d$$

K-means objective function

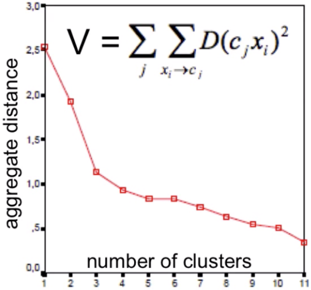
- Minimize aggregate intra-cluster distance:

$$V = \sum_{j=1}^{K} \sum_{x_i \to c_j} D(c_j, x_i)^2$$

- K-means always converges to a **local minimum**

Optimal number of clusters

- Run K-means for different K and plot aggregate intra-cluster distance V
- Analyze the **scree plot**



- K is chosen "where the mountain ends and the rubble begins"

Lloyd's algorithm for K-means

- **Goal**: predict K centroids and a label μ^i for each data point
- **Algorithm**:
  1. Initialize cluster centroids $c_1, c_2, \ldots, c_K \in \mathbb{R}^n$ randomly.
  2. Repeat until convergence:

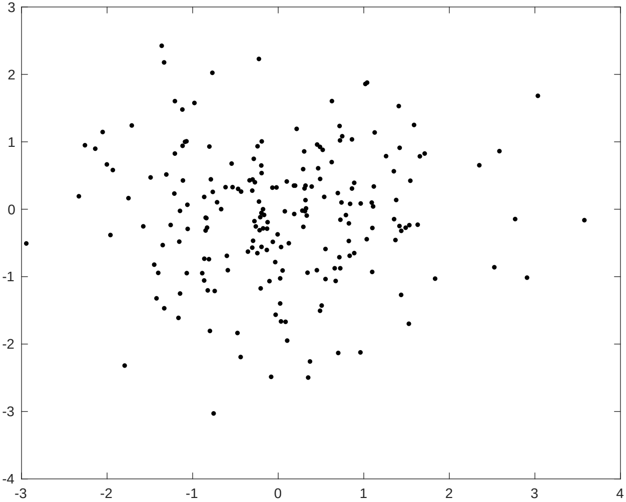
     2.1 For every $i = \{1, \ldots, n\}$, set

     $$\mu^i := \arg\min_j ||x^i - c_j||^2$$

     2.2 For every $j = \{1, \ldots, K\}$ set:

     $$c_j = \frac{\sum_{i=1}^n \mathbb{1}\{\mu^i = j\} x^i}{\sum_{i=1}^n \mathbb{1}\{\mu^i = j\}}$$

Randomly generated data, $n = 200$



MATLAB Syntax

```
opts = statset('Display','final');
[idx_opt,C_opt,sum_opt] = ...
    kmeans(X,K,'Distance','sqeuclidean','Replicates',Z,'Options',opts);
```

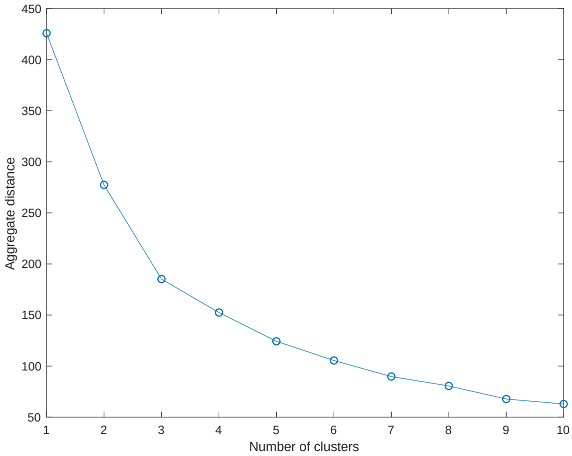
- `idx_opt` $:= n \times 1$ vector of cluster indices
- `C_opt` $:= K \times a$ matrix of centroids
- `sum_opt` $:= K \times 1$ vector of within-cluster sum of points-to-centroid distances

MATLAB Syntax

```
opts = statset('Display','final');
[idx_opt,C_opt,sum_opt] = ...
    kmeans(X,K,'Distance','sqeuclidean','Replicates',Z,'Options',opts);
```

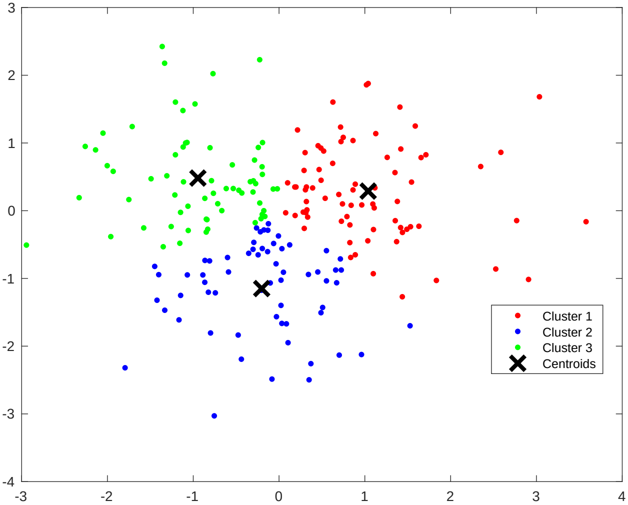
- `X` ⇒ Data matrix
- `K` ⇒ Number of clusters
- `'Distance', 'sqeuclidean'` ⇒ Use Euclidean distance
- `'Replicates', Z` ⇒ Number of initial random assignments
- `'Options', opts` ⇒ Displays the final output

Scree plot



▸ Code

Final clustering with $K = 3$



Overview

- **GOAL:** develop discrete estimators when unobserved heterogeneity is not discrete
- Study **two-step grouped fixed-effects (GFE)** estimators for panel data
  1. K-means clustering to classify individuals into groups
  2. Estimate model with group-specific heterogeneity
- Analyze asymptotic properties of GFE estimators
- Extend two-step approach to improve performance
- Illustration in a dynamic discrete choice model of migration and probit model

What the mainstream does

- **Fixed-effects approaches in nonlinear panel data models**
  - No restrictions on the form of unobserved heterogeneity
  - **BUT** large number of parameters, difficulties with time-varying heterogeneity
  - Arellano and Hahn (2007)
- **Discrete approaches**
  - Individual heterogeneity as a small number of unobserved types
  - **BUT** need restrictions on the form of unobserved heterogeneity
  - Keane and Wolpin (1997)

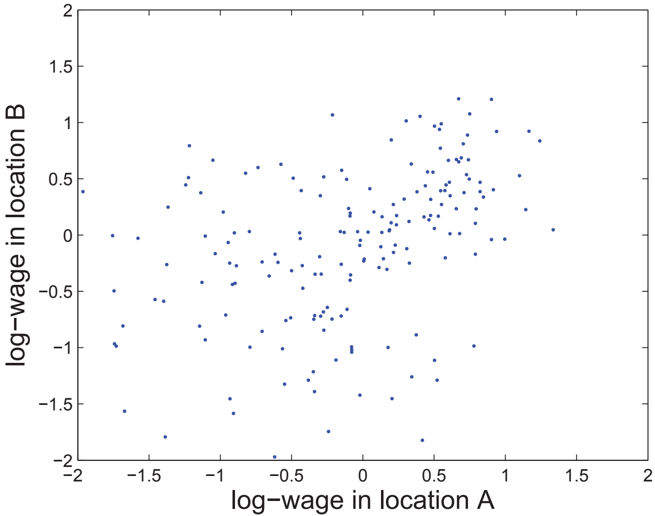
What this paper does

- Considers discrete estimators
- Studies the properties in nonlinear models
- **Main contribution:** no restrictions on individual unobserved heterogeneity

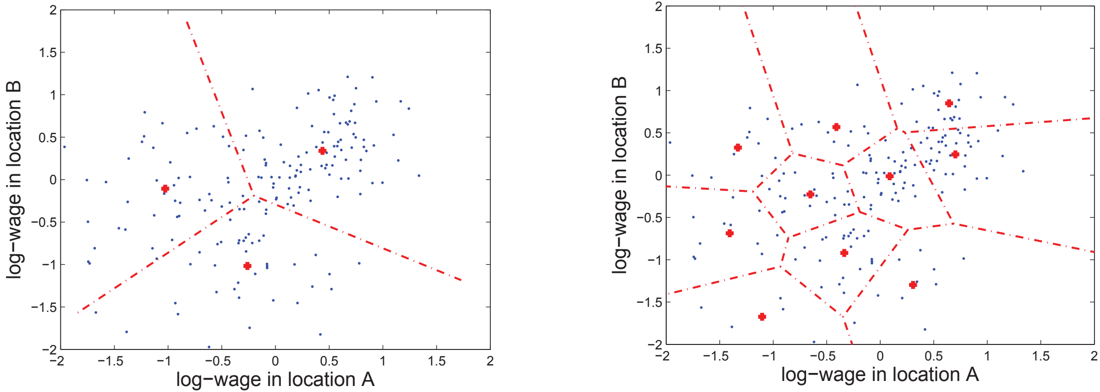
Role of K-means clustering

- Used in the first step of GFE
- Groups together individuals whose **unobserved** types are the most similar
  - No assumptions on heterogeneity needed!
  - Just need to choose K

Role of K-means clustering - Example



Role of K-means clustering - Example



(a) $K = 3$ (b) $K = 10$

Another example

Conditional densities

- $f_i(\alpha_{i0}, \theta_0)$:= conditional density of Y_i on X_i
    - α_{i0}:= individual-specific vectors
    - θ_0 := vector of common parameters
- Focus on densities with the form:

$$\ln f_i(\alpha_{i0}, \theta_0) = \sum_{t=1}^{T} \ln f(Y_{it} | Y_{i,t-1}, X_{it}, \alpha_{it0}, \theta_0)$$

- Densities of exogenous covariates

$$\ln g_i(\mu_{i0}) = \sum_{t=1}^{T} \ln g(X_{it} | X_{i,t-1}, \mu_{it0})$$

Main assumption 1

Assumption 1: unobserved heterogeneity (α_{it0} and μ_{it0} for $t = 1, \ldots, T$) depends on a low-dimensional vector of latent types.

- Discrete heterogeneity as a dimension reduction device
- No specification of mapping between underlying types and heterogeneity
- Let K-means capture the underlying structures

Formal statement

Main assumption 2

Assumption 2: there are individual-specific moments from which the underlying types can be approximated.

- External measurements of the types or constructed from the panel data
- Requires an injectivity condition

Formal statement

Estimator - 1st step: Clustering

- Approximate individual moments h_i and assign clusters using K-means:

$$(\hat{h}, \hat{k}_1, \dots, \hat{k}_N) = \underset{(\tilde{h}, k_1, \dots, k_N)}{\arg\min} \sum_{i=1}^{N} ||h_i - \tilde{h}(k_i)||^2$$

- Use Lloyd's algorithm to perform K-means

Estimator - 2nd step: Estimation

- $\hat{k}_i :=$ cluster assignments
- Two-step GFE estimator:

$$(\hat{\theta}, \hat{\alpha}) = \arg\max_{(\theta,\alpha)} \sum_{i=1}^{N} \ln f_i(\alpha(\hat{k}_i), \theta)$$

where $\alpha = (\alpha(1)', \dots, \alpha(K)')'$

Illustration: Dynamic Discrete Choice Model - Setting

- Model of location choices over J possible alternatives
- Continuum of agents i:
  - Differ in permanent type $\alpha_i \in \mathbb{R}^J$, which determines wage in each location
- **"Detrended" log-wages in location *j*:** $\ln W_{it}(j) = \alpha_i(j) + \varepsilon_{it}(j)$
- **Flow utility of being in location *j* at time *t***: $U_{it}(j) = \rho W_{it}(j) + \xi_{it}(j)$
- **Cost of moving between location *j* and *j'***: $c_i(j)$

Illustration: DDC Model - Data

- NLSY79: males at least 22 years old in 1979
- $J = 2$ large regions: North-East and South (A) and Nort-Central and West (B)
- 1889 workers, observed for an average of 12.3 years

Illustration: DDC Model - Estimation

Two steps:

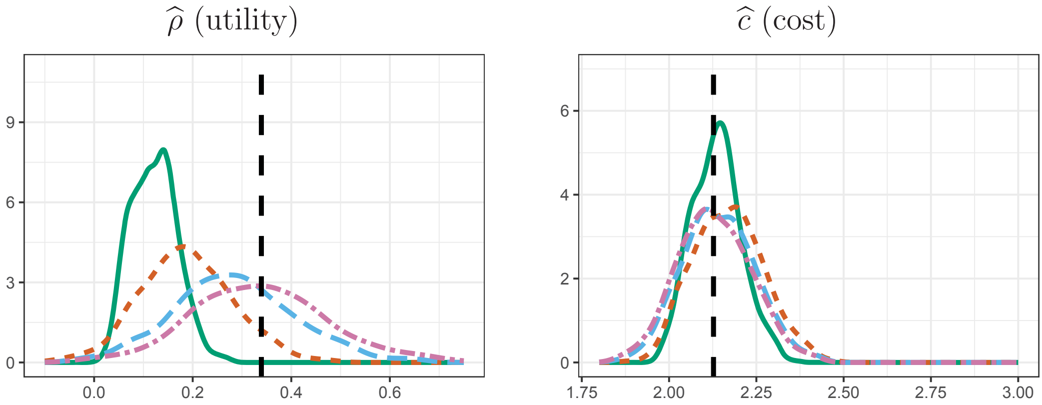
1. Given an i.i.d. sample $(W_{i1}, \ldots, W_{iT}, j_{i1}, \ldots, j_{iT})$ estimate $\alpha_i(j_{it})$:

$$(\hat{\alpha}, \hat{k}_1, \ldots, \hat{k}_N) = \underset{(\tilde{\alpha}, k_1, \ldots, k_N)}{\arg\min} \sum_{i=1}^{N} \sum_{t=1}^{T} (\ln W_{it} - \tilde{\alpha}(k_i, j_{it}))^2$$

2. Maximize the log-likelihood of choices

$$(\hat{\theta}, \hat{c}) = \underset{(\theta, c)}{\arg\max} \sum_{i=1}^{N} \sum_{t=1}^{T} \sum_{j=1}^{J} \mathbb{1}\{j_{it} = j\} \ln Pr(j_{it} = j | j_{i,t-1}, \mathcal{J}_{i,t-1}, \hat{\alpha}(\hat{k}_i, \mathcal{J}_{i,t-1}), c(\hat{k}_i, \mathcal{J}_{i,t-1}), \theta)$$

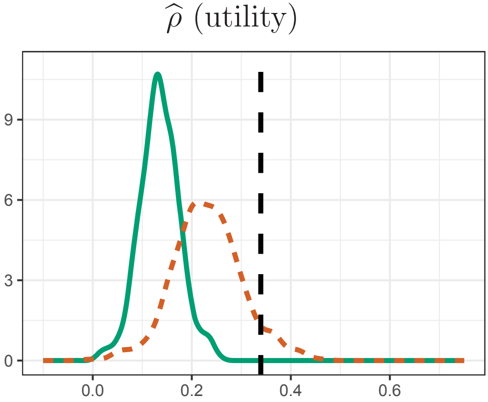
Illustration:DDC - GFE estimates

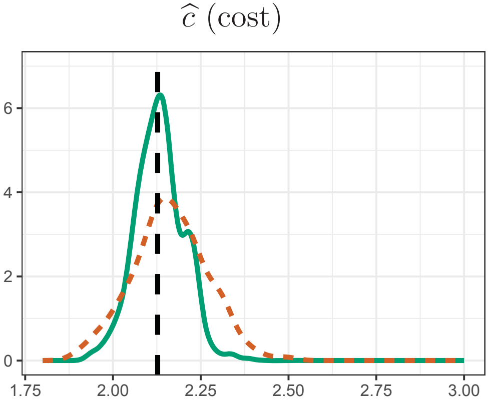


Solid is two-step GFE, dotted is bias-corrected, dashed is iterated once and biased corrected, dashed-dotted is iterated three times and bias corrected.

The vertical line is the true parameter value.

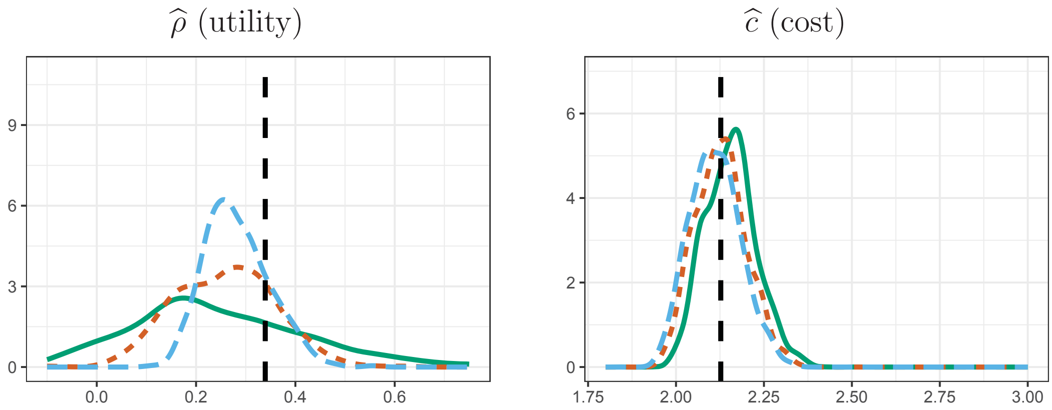
Illustration: DDC - FE estimates





Solid is fixed-effects, dotted is bias-corrected fixed effects

Illustration: DDC - RE estimates



Solid is $K = 2$, dotted is $K = 4$, dashed is $K = 8$ groups

Key takeaways

- Illustration: dynamic discrete choice model
  1. Discrete GFE when unobserved heterogeneity is continuous
  2. Good performance at low computational cost
  3. Potential role for GFE estimators in structural models
- In general:
  1. Use of discrete estimators as a dimension reduction device
  2. Role of K-means

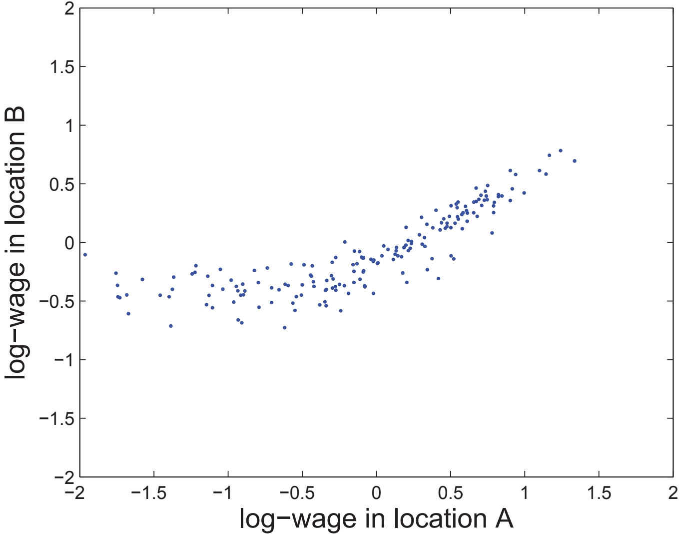
Appendix

Code for scree plot

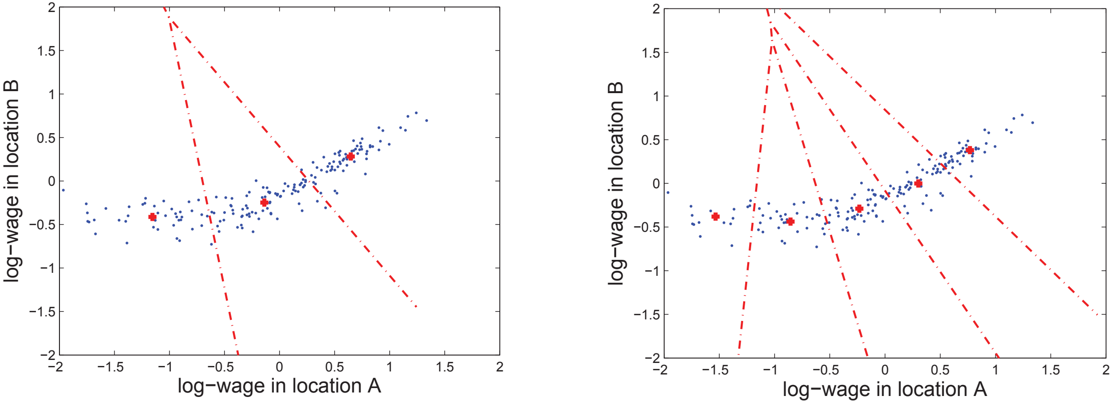
```
%First, we initialize a grid for different values of K we will try
K_grid=1:10;
%Then, we initialize a vector which will contain the aggregate distances.
%We will have one of such distances for every value of K we try.
agg_sum_total=zeros(length(K_grid),1);
opts = statset('Display','final');
for i=K_grid
    [idx,C,sumd]=kmeans(X,i,'Distance','sqeuclidean','Replicates',5,...
        'Options',opts);
    agg_sum_total(i)=sum(sumd);
end
```

Back

Another example



Another example



(a) $K = 3$ (b) $K = 5$

Back

Assumption 1 - Formal statement

Assumption 1: (underlying dimension) There exist vectors ξ_{i0} of dimension d, vectors λ_{t0} of dimension d_λ, and two functions α and μ, such that $\alpha_{i,t0} = \alpha(\xi_{i0}, \lambda_{t0})$ and $\mu_{it0} = \mu(\xi_{i0}, \lambda_{t0})$.

Back

Assumption 2 - Formal statement

Assumption 2: (injective moments) There exist vectors h_i, and a function φ, such that $plim_{S\to\infty} h_i = \varphi(\xi_{i0})$, and $\frac{1}{N}\sum_{i=1}^{N} ||h_i - \varphi(\xi_{i0})||^2 = O_p(1/S)$ as N, S tend to infinity. Moreover, there exists a function ψ such that $\xi_{i0} = \psi(\varphi(\xi_{i0}))$.

Back