Data and computation 2

Alfred Galichon (New York University)

January 2023

Preliminary stuff

- Schedule: Jan 3-19,2023. Lectures and recitations are 75min in length and meet in KMC 2-80.
    - Lectures: 4 times a week, Tue and Thu, from 930am-1045am and from 11am-1215pm.
    - Recitations: 2 times a week, Tue and Fri, from 2pm-315pm. **Exception: First recitation (Tue 1/3) will be moved to Friday 1/6, 330pm-445pm**.
- Course description: This course is the sequel to Data and Computation 1, and in the same spirit brings students up to speed with modern tools to manage economic data at a graduate level. The course is centered on the Python/numpy ecosystem, and will introduce topics such as 1) replicability through containerization, high performance computing, and cloud computing, 2) logistic regression and model selection, 3) computational market design, 4) network problems, and 5) dynamic discrete choice problems.
- Course material: https://github.com/alfredgalichon/nyu-econ-ga-4022

Introducting the teaching team

- **Alfred Galichon**, lecturer: professor of economics and of mathematics at NYU
- **Giovanni Montanari**, TA: graduate student in economics at NYU

Remarks on this course

- Teaching format: mix of theory and lab.
- Programming: our demos will be done Python with Google Colab.
- Questions?

This course

- is focused on models of demand, matching models, and optimal transport methods, with various applications pertaining to labor markets, economics of marriage, industrial organization, matching platforms, networks, and international trade, from the crossed perspectives of theory, empirics and computation
- will introduce tools from economic theory, mathematics, econometrics and computing, on a needs basis, without any particular prerequisite other than those for a first year graduate sequence in Economics
- aims at providing a bridge between theory and practice, and has an unusual teaching format: each teaching "block" will be made of a mix of theory and coding (in Python), based on an empirical application related to the theory just seen.
- is taught in a workshop, artisanal, kind of style.
- aims at providing a bridge between theory and practice, with a mix of theory and coding (in Python), based on an empirical application related to the theory just seen.

Readings (optional)

- Various lectures from math+econ+code available at `https://github.com/math-econ-code/mec_optim` and `https://github.com/math-econ-code/mec_equil`
- Materials from the Software Carpentry organization available at `https://software-carpentry.org/`
- Various lectures from QuantEcon available at `https://python.quantecon.org/`
- Python Data Science Handbook by Jake Vanderplas `https://jakevdp.github.io/PythonDataScienceHandbook/`
- *Optimal Transport Methods in Economics*, by Alfred Galichon
- *Computational Optimal Transport: With Applications to Data Science*, by Peyré and Cuturi
- *Statistical Rethinking* by Richard McElreath
- *Linear Algebra Done Right*, 3rd edition by Sheldon Axler
- *All of Statistics* by Larry Wasserman

Outline: week 1

Week 1: regression and GLM
L01. Logistic regression refreshers. Link with Poisson MLE.
L02. GLMs and computation using scikit learn
L03. Regularization and an application.
L04. Fixed effects and the gravity equation

R01. Upgrading the tools: git, docker
R02: Getting started on HW1

Outline: week 2

Week 2: market design

L05. Basics of linear optimization

L06. Optimal assignments and the Becker model

L07. Estimating matching models

L08. Matching and bargaining

R03. Optimization, gradient descent, automatic differentiation

R04: Getting started on HW2

Outline: week 3

Week 3: network problems, dynamic choice problems
L09. Topology on networks
L10. Transportation on networks
L11. Finite-horizon dynamic discrete choice
L12. Stationary discrete choice

R05. Urban economics toolkit
R06: Getting started on HW3

Getting started

- From your browser, go to https://colab.research.google.com/
- In the menu that appears, choose the 'github' tab
- Enter 'alfredgalichon/nyu-econ-ga-4022' in the box when prompted
- Choose the relevant notebook

Section 1

L1: logistic regression reminders

Reference:

- [McF] D. McFadden (1981). "Econometric Models of Probabilistic Choice," in C.F. Manski and D. McFadden (eds.), *Structural analysis of discrete data with econometric applications*, MIT Press.
- [DCMS] K. Train. (2009). *Discrete Choice Methods with Simulation*. 2nd Edition. Cambridge University Press.

This lecture

1. Choice probabilities
2. The logit model
3. Link with Poisson regression

Discrete choice models

- Assume a consumer is facing a number of options $j \in \mathcal{J}_0 = \mathcal{J} \cup \{0\}$, where $j = 0$ is a default option. The consumer is drawing a utility shock which is a vector $\varepsilon = \left(\varepsilon_0, \ldots, \varepsilon_{|\mathcal{J}|}\right) \sim \mathbf{P}$ such that the utility of option j is $U_j + \varepsilon_j$, while the outside option yields utility ε_0.
- U is called vector of *systematic utilities*; ε is called vector of *utility shocks*.
- We assume thoughout that $\mathbf{P}$ has a density with respect to the Lebesgue measure, and has full support.
- The preferred option is the one which attains the maximum in

$$\max_{j \in \mathcal{J}} \left\{ U_j + \varepsilon_j, \varepsilon_0 \right\}.$$

Choice probabilities, definition

- Let $s_j = \sigma_j(U)$ be the probability of choosing option j, where σ is given by
$$\sigma_j(U) = \Pr(U_j + \varepsilon_j \geq U_z + \varepsilon_z \text{ for all } z \in \mathcal{J}_0).$$
- Note that if $s = \sigma(U)$, then $s_j > 0$ for all $j \in \mathcal{J}_0$ and $\sum_{j \in \mathcal{J}_0} s_j = 1$.
- Note that because the distribution $\mathbf{P}$ of ε is continuous, the probability of being indifferent between two options is zero, and hence we could have indifferently replaced weak preference $\geq$ by strict preference $>$. Without this, choice probabilities may not have been well defined.

Choice probabilities, properties

- $\sigma_j(U)$ is increasing in U_j.
- $\sigma_j(U)$ is weakly decreasing in $U_{j'}$ for $j' \neq j$.
- If one replaces (U_j) by $(U_j + c)$, for a constant c, one has $\sigma(U + c) = \sigma(U)$.

Normalization

- Because of the last property, we can normalize the utility of one of the alternatives. We will normalize the utility of the utility associated to $j = 0$, and hence take
$$U_0 = 0.$$
- Thus in the sequel, σ will be seen as a mapping from $\mathbb{R}^{\mathcal{J}}$ to the set of $(s_j)_{j \in \mathcal{J}}$ such that $s_j > 0$ and $\sum_{j \in \mathcal{J}} s_j < 1$, and the choice probability of alternative $j = 0$ is recovered by
$$s_0 = 1 - \sum_{j \in \mathcal{J}} s_j.$$

Choice probabilities inversion

- In many settings, the econometrician observes the market shares s_j and wants to deduce the corresponding vector of systematic utilities. That is, we would like to solve:
  Problem. *Given a vector s with positive entries satisfying* $\sum_{j\in\mathcal{J}} s_j < 1$, *characterize and compute the set*

$$\sigma^{-1}(s) = \left\{ U \in \mathbb{R}^{\mathcal{J}} : \sigma(U) = s \right\}.$$

- This problem is called "demand inversion," or "conditional choice probability inversion," or "identification problem." It is a central issue in econometrics/industrial organization and will be a key building block for matching models.

The Daly-Williams-Zachary theorem

- Define the expected indirect utility of consumers by

$$G(U) = \mathbb{E}\left[\max_{j \in \mathcal{J}} (U_j + \varepsilon_j, \varepsilon_0)\right]$$

  In the discrete choice literature, this is called *McFadden's surplus function*.

- As the expectation of the maximum of terms which are linear in U, G is convex function in U (strictly convex in fact), and

$$\frac{\partial G}{\partial U_j}(U) = \Pr(U_j + \varepsilon_j \geq U_z + \varepsilon_z \text{ for all } z \in \mathcal{J}_0).$$

  But the right-hand side is simply the probability s_j of chosing option j; therefore, we get:
  Theorem (Daly-Zachary-Williams). *The map σ coincides with the gradient of G, that is*

$$\sigma(U) = \nabla G(U). \tag{1}$$

Gumbel distribution

- Assume that **P** is the distributions of i.i.d. standard type I extreme value random variables, a.k.a. standard Gumbel distributions, which has c.d.f.

$$F(z) = \exp\left(-\exp\left(-x+\gamma\right)\right)$$

  where $\gamma = 0.5772...$ (Euler's constant). The mean of this distribution is zero.
- Basic fact from extreme value theory: if ε_1,...,ε_n are i.i.d. Gumbel distributions, then $\max\left\{u_i + \varepsilon_i\right\}$ has the same distribution as $\log\left(\sum_{i=1}^{n} \exp u_i\right) + \epsilon$, where ϵ is also a Gumbel. (Proof of this fact in class).
- Note that the literature usually calls "standard Gumbel" the distribution with c.d.f. $\exp\left(-\exp\left(-x\right)\right)$; but that distribution has mean γ, which is why we slightly depart from the convention.

Social welfare function and choice probabilities, logit model

- Then
$$G(U) = \log\left(1 + \sum_{j\in\mathcal{J}} \exp(U_j)\right)$$
where $s_0 = 1 - \sum_{j\in\mathcal{J}} s_j$.
- As a result, the choice probability of alternative j is proportional to the exponential of the systematic utility associated with U, that is
$$\sigma_j(U) = \frac{\exp U_j}{1 + \sum_{j'\in\mathcal{J}} \exp(U_{j'})}$$
which is sometimes called a *Gibbs distribution*.
- Assume that the random utility shock is scaled by a factor T. Then
$$\sigma_j(U) = \frac{\exp(U_j/T)}{1 + \sum_{j'\in\mathcal{J}} \exp(U_{j'}/T)}$$
which is sometimes called the *soft-max operator*, and converges as $T \to 0$ toward
$$\max_{j\in\mathcal{J}} \{U_j, 0\}.$$

Logistic regression

- Assume individual i associates the following utility to decision j

$$\sum_k \Phi_{ij}^k \lambda_k + \varepsilon_{ij}$$

where ε_{ij} are iid Gumbel distributions, i.e. of c.d.f. $\exp\left(-\exp\left(-x\right)\right)$.

- The conditional probability that i chooses j is

$$\pi_{ij} = \frac{\exp\left(\sum_k \Phi_{ij}^k \lambda_k\right)}{\sum_{j'} \exp\left(\sum_k \Phi_{ij'}^k \lambda_k\right)}$$

and therefore the conditional log-likelihood associated with j is the logistic regression

$$l_{ij}\left(\lambda\right) = \log \pi_{ij} = \sum_k \Phi_{ij}^k \lambda_k - \log \sum_{j'} \exp\left(\sum_k \Phi_{ij'}^k \lambda_k\right)$$

Logistic regression as MLE

- Then, if $J(i)$ is the actual choice of i, and $\hat{\pi}_{ij} = 1\{j = J(i)\}$, the sample log-likelihood is

$$l(\lambda) = \sum_i l_{iJ(i)}(\lambda) = \hat{\pi}^\top \Phi\lambda - \sum_i \log \sum_{j'} \exp\left((\Phi\lambda)_{ij'}\right)$$

  and the logistic regression can be expressed as

$$\max_\lambda \left\{ \hat{\pi}^\top \Phi\lambda - \sum_i \log \sum_{j'} \exp\left((\Phi\lambda)_{ij'}\right) \right\}$$

- The first order conditions are

$$\left(\hat{\pi} - \pi^\lambda\right)^\top \Phi = 0 \text{ where } \pi^\lambda_{ij} = \frac{\exp\left((\Phi\lambda)_{ij}\right)}{\sum_{j'} \exp\left((\Phi\lambda)_{ij'}\right)}$$

  that is

$$\sum_i \pi^\lambda_{ij} \Phi^k_{ij} = \sum_i \hat{\pi}_{ij} \Phi^k_{ij}$$

  which interprets as predicted moments=observed moments.

Computation of the logistic regression

- See notebook. We will run once the DIY (do-it-yourself) approach by optimizing ourselves the log-likelihood. Later on, we shall use packages, such as scikit-learn. But for that we will need to understand the link with generalized linear models.

Section 2

L2: Generalized linear models

Reference:

- [McCN] McCullagh and Nelder (1989). *Generalized Linear Models*. Chapman and Hall/CRC

Generalized linear models

- In many setting, an economic model will allow to make predictions on the conditional mean of a dependent random variable y given explanatory random vector x.
- In the case of linear regression, we have

$$E\left[y|x\right] = x^{\top}\beta,$$

however, we shall encounter situations where it will be useful to be more general.

- This leads us to *generalized linear models* (GLM), which specify

$$E\left[y|x\right] = g^{-1}\left(x^{\top}\beta\right)$$

where $g : \mathbb{R} \to \mathbb{R}$ is an increasing and continuous function called *link function*.

- Often we shall specify in addition $Var\left(y|x\right) = V\left(g^{-1}\left(x^{\top}\beta\right)\right)$.

Example 1: least squares

- In least squares (OLS), have

$$y = x^{\top}\beta + \epsilon$$

  with $E\left[\epsilon|x\right] = 0$, in which case $g\left(z\right) = z$.
- Additionally, assuming $E\left[\epsilon^2|x\right] = \sigma^2$, we have

$$Var\left(y|x\right) = \sigma^2.$$

Example 2: Poisson regression

- Recall a Poisson distribution with parameter $\theta \in (0, +\infty)$ has probability mass
$$\pi_{z|\theta} = \frac{e^{-\theta}\theta^{z}}{z!}$$
over $z \in \{0, 1, 2, ...\}$. It has expectation and variance θ.
- Assume that conditional on x, y has a Poisson distribution of parameter $\theta = \exp\left(x^{\top}\beta\right)$. Then
$$E\left[y|x\right] = \exp\left(x^{\top}\beta\right)$$
so in this case $g = \ln$.
- Note that we get
$$var\left(y|x\right) = \exp\left(x^{\top}\beta\right)$$
which may be overrestrictive (more on this later).

Poisson regression (ctd)

- Sample log-likelihood

$$\sum_i -\exp\left(x_i^\top \beta\right) + x_i^\top \beta y_i - \ln\left(y_i!\right)$$

  and therefore, max likelihood yields the Poisson regression

$$\max_\beta \left\{ \sum_i -\exp\left(x_i^\top \beta\right) + x_i^\top \beta y_i \right\}$$

- First order conditions give

$$\sum_i \left(y_i - \exp\left(x_i^\top \beta\right)\right) x_i = 0$$

ML inference in Poisson regression

- Recall that if $E_{P_n} \log p(\beta, z)$ is the log-likelihood of the sample, and setting $l(\beta, z) = \log p(\beta, z)$ we get

$$\begin{aligned} E_{P_n}\left[\partial_\beta l(\beta_n, z)\right] &= 0 \\ E_P\left[\partial_\beta l(\beta, z)\right] &= 0 \end{aligned}$$

thus

$$E_P\left[\partial_\beta l(\beta_n, z)\right] - E_P\left[\partial_\beta l(\beta, z)\right] = E_P\left[\partial_\beta l(\beta_n, z)\right] - E_{P_n}\left[\partial_\beta l(\beta_n, z)\right]$$

therefore

$$(\beta_n - \beta) E_P\left[\partial^2_\beta l(\beta_n, z)\right] = -\frac{1}{\sqrt{n}} g_n\left(\partial_\beta l(\beta, z)\right)$$

where $g_n f = \sqrt{n}\,(E_{P_n} f - E_P f)$.

- Thus

$$\beta_n - \beta = -\frac{1}{\sqrt{n}}\left(E_P\left[\partial^2_\beta l(\beta, z)\right]\right)^{-1} g_n\left(\partial_\beta l(\beta, z)\right)$$

ML inference in Poisson regression (ctd)

- Hence

$$V(\beta_n - \beta) = \frac{1}{n}\left(E_P\left[\partial^2_\beta l(\beta, z)\right]\right)^{-1} \times E_P\left(\partial_\beta l(\beta, z)\left(\partial_\beta l(\beta, z)\right)^\top\right) \times \left(E_P\left[\partial^2_\beta l(\beta, z)\right]\right)^{-1}$$

- And because at the ML parameter

$$E_P\left(\partial_\beta l(\beta, z)\left(\partial_\beta l(\beta, z)\right)^\top\right) = E_P\left[\partial^2_\beta l(\beta, z)\right],$$

we have thus

$$V(\beta_n - \beta) = \frac{1}{n}\left(E_P\left[\partial^2_\beta l(\beta, z)\right]\right)^{-1}.$$

Estimation of GLM

- Actually, we don't need to assume that $y \sim \mathit{Poisson}\left(\exp(x^\top \beta)\right)$ to estimate β.
- Consider X the matrix obtained by stacking the rows $x_i^\top$ on top of each other. Compute

$$\max_\beta \left\{ y^\top X\beta - 1^\top \exp\left(X\beta\right) \right\}$$

and define $\bar{y} = \exp\left(X\beta\right)$ the predictor of y. One has

$$\sum_i y_i X_{ik} = \sum_i \bar{y}_i X_{ik} \ \forall k$$

and therefore β is obtained by matching the predicted moments with the observed ones

$$\mathbb{E}\left[y_i x_i\right] = \mathbb{E}\left[\bar{y}_i x_i\right].$$

Inference in GLM

- While the point estimate is unchanged wrt the Poisson regression, the inference is changed as soon as one departs from the assumption that $Var(y|x) = x^\top \beta$. Assume $Var(y|x) = V(y|x)$.
- The estimation of β is now seen as what is called an *M-estimation* procedure

$$\max_\beta \frac{1}{n} \sum_{i=1}^n F(z_i, \theta).$$

- The derivation done for MLE applies replacing $\partial_\beta l(\beta, z_i) = \partial_\beta \log p(\beta, z_i)$ by $\partial_\beta l(\beta, z_i) = \left(y_i - \exp\left(x_i^\top \beta\right)\right) x_i$ with the provision that $E_P\left[\partial^2_\beta l(\beta, z)\right] \neq E_P\left[\partial l(\beta, z)\, \partial l(\beta, z)^\top\right]$. Hence

$$\begin{aligned} V(\beta_n - \beta) = \frac{1}{n} & \left(E_P\left[\partial^2_\beta l(\beta, z)\right]\right)^{-1} \\ & \times E_P\left(\partial_\beta l(\beta, z)\left(\partial_\beta l(\beta, z)\right)^\top\right) \\ & \times \left(E_P\left[\partial^2_\beta l(\beta, z)\right]\right)^{-1} \end{aligned}$$

Inference in GLM (ctd)

- We have therefore

$$E_P\left[\partial^2_\beta l\left(\beta,z\right)\right]=E\left[\exp\left(x^\top\beta\right)xx^\top\right]$$

and

$$\begin{aligned}E_P\left[\partial_\beta l\left(\beta,z\right)\left(\partial_\beta l\left(\beta,z\right)\right)^\top\right]&=E\left[\left(y-\exp\left(x^\top\beta\right)\right)^2xx^\top\right]\\&=E\left[V\left(y|x\right)xx^\top\right].\end{aligned}$$

Computation of GLM in R

In R, compute using
`glm(fomrmula , family="poisson", data)`
See an example at
https://stats.idre.ucla.edu/r/dae/poisson-regression/

GLM and duality

- Consider $y \in \mathbb{R}_{+}^{n}$, $\beta \in R^{k}$ and X a $n \times k$ matrix

Theorem (GLM duality). The primal problem

$$\max_{\beta} \left\{ y^{\top} X\beta - 1^{\top} \exp\left(X\beta\right) \right\}$$

has dual

$$\min_{z \in \mathbb{R}_{+}^{n}} z^{\top} \left(\ln z - 1\right)$$
$$s.t. X^{\top} \left(z - y\right) = 0.$$

Proof. Write the Lagrangian for the problem

$$\begin{aligned} & \min_{z \geq 0} \max_{\beta} z^{\top} \left(\ln z - 1\right) - \left(z - y\right)^{\top} X\beta \\ & = \max_{\beta} y^{\top} X\beta + \min_{z \geq 0} \left\{ z^{\top} \left(\ln z - 1\right) - z^{\top} X\beta \right\} \end{aligned}$$

has $\ln z = X\beta$ and $z^{\top} \left(\ln z - 1\right) - z^{\top} X\beta = -z^{\top} 1 = -1^{\top} \exp\left(X\beta\right)$ and hence this is

$$\max_{\beta} y^{\top} X\beta - 1^{\top} \exp\left(X\beta\right).$$

Multinomial logit model (ctd)

- As a result, if $J(i)$ is the actual choice of i, and $\hat{\pi}_{ij} = 1\{j = J(i)\}$, the logistic regression can be expressed as

$$l(\lambda) = \hat{\pi}^\top \Phi \lambda - \sum_i \log \sum_j \exp\left((\Phi\lambda)_{ij} \right)$$

- This is *almost*, but *not quite* the form of a GLM – notice the log. To make the precise connection with GLM/Poisson regression, we need to introduce *individual fixed effects*. For that, we need refreshers on vectorization and Kronecker products.

Vectorization

- We need to represent $(\Phi\lambda)_{ij}$ as a vector of dimension $\mathbb{R}^{n^2}$, an operation called *vectorization*.
- To do this, we can either append rows (row-major order, or C ordering); or append columns (column-major order, or Fortran ordering). As R uses the C convention, we shall adopt the later. A 2x2 matrix A is therefore represented as

$$vec\left(A\right) = \left(A_{11}, A_{21}, A_{12}, A_{22}\right).$$

Kronecker products

- For A an $n \times m$ matrix and B an $p \times q$ matrix, the Kronecker product $A \otimes B$ is the $np \times mq$ matrix defined as

$$A \otimes B = \begin{pmatrix} a_{11}B & ... & a_{1m}B \\ ... & ... & ... \\ a_{n1}B & ... & a_{nm}B \end{pmatrix}.$$

- A very important identity is

$$vec\left(BXA^{\top}\right) = \left(A \otimes B\right) vec\left(X\right).$$

Logistic regresssion as a GLM

- Introduce a fixed effect u_i and let $\beta = \left(\lambda^\top, u^\top\right)^\top$. We rewrite $(\lambda, u) \to \left((\Phi\lambda)_{ij} - u_i\right)_{ij}$ in a matrix form by defining

$$X = \begin{pmatrix}\Phi & -1_n \otimes I_n\end{pmatrix}$$

where $\otimes$ is the Kronecker product and we have

$$X\beta = vec\left(\left((\Phi\lambda)_{ij} - u_i\right)_{ij}\right).$$

- The Poisson regression of $\hat{\pi}_{ij}$ on X yields

$$\max_{\lambda,u}\left\{-\sum_{ij}\exp\left((\Phi\lambda)_{ij} - u_i\right) + \sum_{ij}\hat{\pi}_{ij}\left((\Phi\lambda)_{ij} - u_i\right)\right\}$$

therefore

$$\max_{\lambda,u}\left\{-\sum_{ij}\exp\left((\Phi\lambda)_{ij} - u_i\right) + \sum_{ij}\hat{\pi}_{ij}(\Phi\lambda)_{ij} - \sum_{i} u_i\right\}.$$

Logistic regresssion as a GLM (ctd)

- Taking first order conditions in u_i we get

$$\sum_j \exp\left((\Phi\lambda)_{ij} - u_i \right) = 1$$

- Therefore, $u_i = \log \sum_j \exp\left((\Phi\lambda)_{ij} \right)$ and the problem becomes the MLE in the multinomial logit model

$$\max_{\lambda, u} \left\{ \sum_{ij} \hat{\pi}_{ij} (\Phi\lambda)_{ij} - \sum_i \log \sum_j \exp\left((\Phi\lambda)_{ij} \right) \right\}.$$

- To summarize: **logistic regression = GLM + fixed effect**.

Discrete choice application

- See Jupyter notebook.