Introduction

Contents of the course,
logic, sets, functions
SB: A1, 13.5

General info

- Lecturer: Mitri Kitti (mitri.kitti@gmail.com)
- Web-page: www.mitrikitti.fi/mathcamp.html
- Material
  - handouts by Juuso Välimäki on course webpage
  - Book: Simon and Blume, *Mathematics for Economists*, relevant chapters indicated in slides (SB: XX)
  - exercises and slides can be found in the course website

Contents

- Basics: logic, sets, functions
- Elementary analysis
  - topology: metrics, open and closed sets
  - normed vector spaces
  - continuity of functions, Weierstrass theorem
- Linear algebra
  - linear functions
  - rank of a matrix, determinant
  - eigenvalues and eigenvectors
  - application: linear dynamic systems
- Multivariate calculus
  - differential and partial derivatives
  - Taylor series, linearizations and log-linearizations
  - implicit function theorem
  - application: stability of nonlinear dynamic systems

Contents and schedule

- Unconstrained optimization
  - first and second order optimality conditions
  - envelope theorem
- Convex analysis
  - convex sets and concave functions, quasiconcavity
  - concave and quasiconcave optimization
- Nonlinear programming
  - first order necessary conditions
  - comparative statics
  - sufficient optimality conditions

Contents and schedule

- Introduction to Programming
- Programming with Matlab
  - basics, linear algebra
  - conditional statements and loops
  - writing functions
- Basics on Dynare

Logic

- A proposition is a statement which can be declared true or false without ambiguity
- A propositional (or sentential) function is a statement which becomes a proposition when we replace the variable with a specified value
  - "$2+2=4$" is a proposition while "$x<2$" is a propositional function
- Propositional functions lead to propositions when the variables are quantified
  - for x (in the domain of discourse) is denoted as $\forall$
  - there is an x is denoted as $\exists$, if there is uniqueness then we denote $\exists!$

Logic

- Connectives (for constructing new propositions)
  - and ($\wedge$), or ($\vee$), negation ($\neg$)
- Implication "$\Rightarrow$"
  - we write $A \Rightarrow B$ if proposition B is true when proposition A is true, we also say that A is sufficient condition for B and B is necessary condition for A
  - if $A \Rightarrow B$ and $B \Rightarrow A$ we write $A \Leftrightarrow B$
- Note: $\neg(\forall x,\ P(x)) \Leftrightarrow \exists x, \neg P(x)$

Methods of proof

- Direct proof
  - showing that conclusion (C) follows from hypothesis (H)
- Proof by contrapositive
  - to show $H \Rightarrow C$ it is shown that $\neg C \Rightarrow \neg H$
- Proof by contradiction
  - assume $\neg(H \Rightarrow C)$ and show that $\neg(H \Rightarrow C) \wedge H$ is a contradiction
- Proof by construction
- Proof by decomposition

Sets

- Naive set theory: set is a collection of objects
- Membership relation $x \in A$ (element x belongs to set A)
  - set inclusion $B \subseteq A$ if $x \in B \Rightarrow x \in A$, proper inclusion $B \subset A$: $B \subseteq A$ and $B \neq A$
- The usual binary operations
  - union $A \cup B$
  - intersection $A \cap B$
  - complement A^c, the set difference (complement of A relative to B) of B and A is $B \setminus A$ (elements of B that are not elements of A)
  - Cartesian product $A \times B$
- De Morgan laws $(A \cap B)^c = A^c \cup B^c$ and $(A \cup B)^c = A^c \cap B^c$

Sets

- Examples
  - real numbers $\mathbb{R}$, integers $\mathbb{Z}$, positive integers $\mathbb{N}$, rational numbers $\mathbb{Q}$
  - real coordinate sets $\mathbb{R}^n$: n-times Cartesian product of $\mathbb{R}$
- Power set $\mathcal{P}(A)$ or 2^A: the set of all subsets of A
- Arbitrary unions and complements, index set I
  - union over I: $\bigcup_{i \in I} A_i$
  - intersection over I: $\bigcap_{i \in I} A_i$

Real coordinate spaces

- The sets $\mathbb{R}^n$, $n \geq 1$ are called real coordinate spaces
  - we write an element $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$ with $x_i \in \mathbb{R}$, $i = 1, \ldots, n$ also as a column array (vector) $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$
- Inner product of two elements of $\mathbb{R}^n$: $x \cdot y = x^T y = \sum_i x_i y_i$ (here x^T denotes the transpose of x)
  - x is orthogonal to y if $x \cdot y = 0$
  - note: $\cos(\angle x, y) = x \cdot y / (\|x\| \|y\|)$, where $\|x\| = \sqrt{x \cdot x}$
- Least upper bound of a set
  - an element $u \in \mathbb{R}$ is a least upper bound of S if $x \leq u$ for all $x \in S$ and for any $v \in \mathbb{R}$ such that $x \leq v$ for all $x \in S$ it holds that $u \leq v$, we denote $u = \sup S$

Real coordinate spaces

- Infima and suprema
  - greatest lower bound $\inf S$ respectively as $\sup$
  - axiom: any $\emptyset \neq S \subset \mathbb{R}$ with an upper bound has a least upper bound
- Notations
  - zero vector $\mathbf{0} = (0, \ldots, 0)$
  - positive vectors $x \geq \mathbf{0} \Leftrightarrow x_i \geq 0 \forall i = 1, \ldots, n$
  - strictly positive vectors $x \gg \mathbf{0} \Leftrightarrow x_i > 0 \forall i = 1, \ldots, n$
  - positive and strictly positive orthants $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n : x \geq \mathbf{0}\}$ and $\mathbb{R}^n_{++} = \{x \in \mathbb{R}^n : x \gg \mathbf{0}\}$

Functions

- ▲ Function $f : X \mapsto Y$ is a rule that assigns an element $f(x) \in Y$ to each element $x \in X$
  - formally f is a subset of $X \times Y$ such that $\forall x \in X \exists y \in Y : (x, y) \in f$ and $(x, y) \in f \Leftrightarrow \forall z \neq y : (x, z) \notin f$
  - graph of a function is $\{(x, f(x)) \in X \times Y : x \in X\}$
  - note: for each x there is unique $y \in X$, if there were several y's corresponding to $x \in X$, f would be a correspondence (set valued mapping), i.e., $f : X \mapsto \mathcal{P}(Y)$
  - often the notions of map and mapping are used as synonyms of function

Functions: domain and range

- X is the domain of the function and Y is the range
  - note: in these notes the domain is often assumed to be $\mathbb{R}^n$, although we could in most cases assume the domain to be a (open) subset of $\mathbb{R}^n$
- For $A \subset X$ the image of A is $f(A)$
- Pre-image (or inverse image) of $B \in Y$ is $f^{-1}(B) = \{x \in X : \exists y \in B,\ f(x) = y\}$
- ▲ If $f(X) = Y$ then f is surjection (onto)
- ▲ If for each $y \in Y$, $f^{-1}(y)$ consists of at most one element of X, then f is injective (one-to-one) mapping of X to Y

Bijections

- ▲ If f is both surjective and injective, then it is a bijection
  - f is a bijection if and only if f^{-1} is a function
- Examples:
  - identity function $\mathrm{id}(x) = x$ is a bijection
  - exponential function is not bijection between $\mathbb{R}$ and itself but it is a bijection between $\mathbb{R}$ and $\mathbb{R}_{++}$
- ■ If X and Y are finite sets, then there is a bijection between X and Y if and only if the number of elements in the two sets is the same
  - this can be taken as the definition for the same number of elements (cardinality of sets)

Economic models

- Purposes of economic modeling
  - explaining how economies work and how economic agents interact
  - forming an abstraction of observed data
  - means of selection of data
  - other uses: forecasting, policy analysis
- Simplification in modeling
  - choice of variables and relationships relevant for the analysis
- Endogenous and exogenous variables
  - when an economic model involves functions in which some of the variables are not affected by the choice of economic agents, these variables are called exogenous variables
  - exogenous variables are fixed parameters in the model
  - respectively, endogenous variables are those that are affected by economic agents, they are determined by the model

Economic models

- Example: model of a consumer
  - elements: choice of a consumption bundle $x \in \mathbb{R}^n_+$ over budget set $\{x \in \mathbb{R}^n : p \cdot x \leq I, x \geq \mathbf{0}\}$ where $p \gg \mathbf{0}$ and $I > 0$
  - utility function $u : \mathbb{R}^n_+ \mapsto \mathbb{R}$ (more generally preference relation)
  - variables p and I are exogenous and x is endogenous

Elements of Analysis

Some topology, continuity,
Weierstrass theorem
SB: 10–12, 29, 30.1

Metric spaces

Definition

A set X is said to be a metric space if for any two points p and q of X there is associated a real number $d(p,q)$, called the distance from p to q, such that:

(i) $d(p,q) > 0$ if $p \neq q$; $d(p,p) = 0$,
(ii) $d(p,q) = d(q,p)$,
(iii) $d(p,q) \leq d(p,r) + d(r,q)$, for any $r \in X$.

- any function with these three properties is called a distance function or a metric

Examples of metrics

- Example 1: any set X can be made into a metric space by defining $d(x,x) = 0$ and $d(x,y) = 1$ when $x \neq y$
- Example 2: $X = \mathbb{R}^n$, $d(p,q) = \sqrt{\sum_{k=1}^n |p_k - q_k|^2}$, $p = (p_1, \ldots, p_n)$, $q = (q_1, \ldots, q_n)$
- Example 3: $X =$ the set of real valued bounded functions defined on a set S, $d(f,g) = \sup_{x \in S} |f(x) - g(x)|$.
- Example 4: $X =$ the set of infinitely lengthy sequences of real numbers, $d(p,q) = \sup_k |p_k - q_k|$, $p = (p_1, p_2, \ldots)$, $q = (q_1, q_2, \ldots)$

Examples of metrics

- Why would an economist be interested in metric spaces in examples 3 and 4?
  - In some settings the choice variable of a decision maker can be a function or an infinite sequence
- Application for example 3: $S = [0,1]$ with $q \in S$ representing the the quality of a product, function $f(q) =$ price of quality $q \in S$ product
- Application for example 4: infinitely many periods, in each periods there is price for a product, $X =$ sequences of prices

Vector spaces

Definition

A set X is a vector space if the sum of two vectors and the product of a scalar and a vector can be defined

- the sum "+" should satisfy the usual properties of a sum: associativity ($(x+y)+z = x+(y+z)$), commutativity ($x+y = y+x$), there are identity (zero) and inverse (-vector) elements
- the scalar product satisfies distributivity ($a(x+y) = ax+ay$, $(a+b)x = ax+bx$) and compatibility ($a(bx) = (ab)x$) conditions and there is an identity element ($1x = x$)

- Example 1: real coordinate space $\mathbb{R}^n$
- Example 2: X functions on S, define sum and scalar product pointwise

Vector spaces

Definition

Normed vector space X, norm $\|\cdot\|$ (function that measures the length)

- $\|x\| \geq 0$ and $\|x\| = 0$ if and only if $x = \mathbf{0}$
- $\|ax\| = |a|\|x\|$ for all scalars a
- $\|x+y\| \leq \|x\| + \|y\|$ (triangle inequality)

- Example: $X = \mathbb{R}^n$ with $\|x\| = \sqrt{\sum_{k=1}^n x_k^2}$ (Euclidean norm), note $d(x,y) = \|x-y\|$ is a distance

Open sets

- ▲ X metric space, r-neighborhood of x, $N_r(x) = \{y \in X : d(x,y) < r\}$
- ▲ Interior point, x is an interior point of X if there is $r > 0$ such that $N_r(x) \subseteq X$
  - we denote the interior of a set as $\text{int}(S)$

Definition

$S \subseteq X$ is an open set if every point of S is an interior point, i.e., $S = \text{int}(S)$

Open sets

1. X is open as well as $\emptyset$
2. any union of open sets is open
3. intersection of finitely many open sets is open

⋆ these three properties are the definition of a topology!

- Open sets in $\mathbb{R}^n$
  - Norm equivalence theorem: all norms in $\mathbb{R}^n$ define the same topology (the set of open sets)
  - example 1: strictly positive orthant $\mathbb{R}^n_{++}$
  - example 2: open half spaces $\{x \in \mathbb{R}^n : p \cdot x < a\}$

Closed sets

Definition

A set S is closed if S^c (complement) is open

- Equivalent definition
  - closure of S: $x \in \bar{S}$ if for all $\varepsilon > 0$ we have $N_\varepsilon(x) \cap S \neq \emptyset$
  - S is closed if (and only if) $S = \bar{S}$
- Examples
  - positive orthant $\mathbb{R}^n_+$
  - hyperplanes $\{x \in \mathbb{R}^n : p \cdot x = a\}$ and closed half-spaces $\{x \in \mathbb{R}^n : p \cdot x \lesseqgtr a\}$
  - level curves of a continuous function: $\{x \in \mathbb{R}^n : f(x) = \alpha\}$
  - (lower and upper) level sets of a continuous function: $\{x \in \mathbb{R}^n : f(x) \lesseqgtr \alpha\}$
- ▲ Boundary of a set $= \partial S$, $x \in \partial S$ if $N_\varepsilon(x) \cap S \neq \emptyset$ and $N_\varepsilon(x) \cap S^c \neq \emptyset$
  - remark: $\text{int}(S) = S \setminus \partial S$

Sequences and convergence

- A sequence $x^1, x^2, \ldots$, where $x^k \in X$, $k = 1, 2, \ldots$ (X is a metric space), is said to converge to $x \in X$ if for every $\varepsilon > 0$, there is an integer N such that $k \geq N$ implies $d(x^k, x) < \varepsilon$
  - notation for a sequence $\{x^k\}$
  - formally a sequence in S is a function on the set $\mathbb{N} = \{1, 2, \ldots\}$, if $X : \mathbb{N} \mapsto S$ is a sequence then $X(n) = x^n$
  - if $\{x^k\}$ converges to x we denote $x^k \to x$ (as $k \to \infty$) or $\lim\limits_{k\to\infty} x^k = x$
  - x is called the limit point of the sequence
- ■ A set S is closed if and only if $\{x^k\} \subset S$, $x^k \to x$ implies that $x \in S$

Compact sets

Definition

A subset S of a metric space X is said to be compact if every sequence $\{x^k\} \subset S$ has a convergent subsequence with a limit point in S

- given an infinite sequence $\{k_j\}$ of integers with $k_1 < k_2 < k_3 \ldots$ we say that $\{x^{k_j}\}$ is a subsequence of $\{x^k\}$
- compactness can be defined for topological spaces (without metric): S is compact if for arbitrary collection of open sets $\{U_\alpha\}_{\alpha \in A} \subseteq X$ such that $S \subseteq \bigcup_{\alpha \in A} U_\alpha$ there is a finite set $I \subseteq A$ such that $S \subseteq \bigcup_{\alpha \in J} U_\alpha$

Compact sets in $\mathbb{R}^n$

Theorem

Every compact set of a normed vector space is bounded and closed

- $S \subset X$ is bounded if there is $M > 0$ such that $\|x\| \leq M$ for all $x \in S$
- ⋆ For $\mathbb{R}^n$ the converse of the previous theorem is also true, i.e., every bounded and closed set is compact, i.e., $S \subset \mathbb{R}^n$ is compact if and only if it is bounded and closed
- ⋆ is equivalent to Bolzano-Weierstrass theorem: every bounded sequence in $\mathbb{R}^n$ has a convergent subsequence

Continuous functions

Definition

Let X and Y be metric spaces, f is continuous at p if for every $\varepsilon > 0$ there is a $\delta > 0$ such that $d_X(x, p) < \delta \Longrightarrow d_Y(f(x), f(p)) < \varepsilon$

- If f is continuous at any $x \in X$ then f is continuous on X
- Equivalent definitions of continuity
  - $f^{-1}(V)$ is open (closed) for every open (closed) set $V \in X$
  - $\lim\limits_{k\to\infty} f(x^k) = f(p)$ for any sequence $\{x^k\}$ such that $p = \lim\limits_{k\to\infty} x^k$

Lipschitz continuity

- X and Y metric spaces, $f : X \mapsto Y$ is Lipschitz continuous if the is $L \geq 0$ (Lipschitz constant) such that $d_Y(f(x^1), f(x^2)) \leq L d_X(x^1, x^2)$
- If $L \in (0, 1)$ the function is called a contraction
- f is locally Lipschitz on X if any $x \in X$ has an open neighborhood in which f is Lipschitz

Weierstrass theorem

Theorem

Continuous function over a compact set attains its extrema (minimum and maximum)

- S compact, f continuous on S, then there is $\bar{x}$ such that $f(\bar{x}) = \inf_{x \in S} f(x)$ (and the same for supremum)

Linear algebra

Matrices, determinant, inversions
SB: 6–9, 26, 27

Introduction

- Some economic models have a linear structure
- Nonlinear models can be approximated by linear models
- System of linear equations, general form

$$\begin{array}{rcl} a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n & = & b_1 \\ \vdots & \vdots & \vdots \\ a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n & = & b_m \end{array} \Leftrightarrow Ax = b$$

  - a_{ij}, b_j, $i = 1, \ldots, m$, $j = 1, \ldots, n$, are parameters (exogenous variables) and $x_1, \ldots, x_n$ are (endogenous) variables

Example: two products

- Demand $Q_i^d = K_i P_1^{\alpha_{i1}} P_2^{\alpha_{i2}} Y^{\beta_i}$, $i = 1, 2$
  - P_i price of product i, Y income
  - α_{ij} elasticity of demand of product i for price j
  - β_i income elasticity of product i
- Supply $Q_i^s = M_i P_i^{\gamma_i}$, $i = 1, 2$
  - γ_i is the price elasticity of supply
- Equilibrium $Q_i^d = Q_i^s$, $i = 1, 2$
  - Notations $q_i^d = \ln Q_i^d$, $q_i^s = \ln Q_i^s$, $p_i = \ln P_i$, $y = \ln Y$, $m_i = \ln M_i$, $k_i = \ln K_i$

Example: two products

- By taking a logarithm of supply, demand, and the equilibrium condition we get a linear system

$$\begin{array}{rcl} q_i^d & = & k_i + \alpha_{ii} p_i + \alpha_{ij} p_j + \beta_i y, \\ q_i^s & = & m_i + \gamma_i p_i, \\ q_i^s & = & q_i^d. \end{array}$$

- after some algebraic manipulation this results to a system of the form

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

Vector spaces

Definition

$W \subseteq V$ is a subspace of V (vector space) if (*i*) $v, w \in W \Rightarrow v + w \in W$ and (*ii*) $\alpha v \in W$ for all scalars α and $v \in W$

- Let V be a vector space and $v^1, \ldots, v^n \in V$ and $\alpha_1, \ldots, \alpha_n$ scalars. The expression $\alpha_1 v^1 + \cdots + \alpha_n v^n$ is called a linear combination of $v^1, \ldots, v^n$
  - the set of all linear combination of $v^1, \ldots, v^n$, denoted by span$(v^1, \ldots, v^n)$, this set is a subspace of V

Linearly dependent vectors

Definition

Vectors $v^1, \ldots, v^n \in V$ are linearly dependent if there are scalars $\alpha_1, \ldots, \alpha_n$ not all equal to zero such that $\alpha_1 v^1 + \cdots + \alpha_n v^n = \mathbf{0}$

- If vectors are not linearly dependent they are linearly independent
- If vectors $v^1, \ldots, v^n$ are linearly independent and span$(v^1, \ldots, v^n) = V$ then they form a basis of V
- If $v \in V$ is written as $v = \alpha_1 v^1 + \cdots + \alpha_n v^n$, then $(\alpha_1, \ldots, \alpha_n)$ are the coordinates of v w.r.t. the given basis

Vector and affine spaces

- If V has a basis consisting of n elements we say that the dimension of V is n
- Any two basis of a vector space have the same number of elements
- Example: coordinate vectors e_i, $i = 1, \ldots, n$ of $\mathbb{R}^n$

Definition

Translation $x^0 + V$ of a vector space V is called an affine space (set)

- dimension of an affine space $x^0 + V$ is the dimension of the vector space V

Matrices

▲ $m \times n$ (real) matrix A, we denote $A \in \mathbb{R}^{m \times n}$, is an array

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix},$$

where $a_{ij} \in \mathbb{R}$, $i = 1, \ldots, m$, $j = 1, \ldots, m$

- notation: $A_i = (a_{i1}, \ldots, a_{in})$ (ith row), $A^j = (a_{1j}, \ldots, a_{mj})$ (jth column)

- Transpose of a matrix A^T is defined by $(A^T)_{ij} = A_{ji}$
- Special matrices: square, identity matrix (I), symmetric, diagonal, upper (lower) triangular

Invertible matrices

- Multiplication of matrices, $A \in \mathbb{R}^{n\times m}$, $B \in \mathbb{R}^{m\times s}$, the product AB is defined as

$$(AB)_{ij} = \sum_{k=1}^{n} a_{ik} b_{kj} = A_i \cdot B^j$$

Definition

$A \in \mathbb{R}^{n\times n}$ is invertible (nonsingular) if there is $A^{-1} \in \mathbb{R}^{n\times n}$ such that $AA^{-1} = A^{-1}A = I$

- Some results
  - Vectors $v^1, \dots, v^k \in \mathbb{R}^n$ with $A^j = v^j$, $j = 1 \dots, k$ are linearly dependent if and only if the system $Ax = \mathbf{0}$ has a nonzero solution $x \in \mathbb{R}^k$
  - $A \in \mathbb{R}^{n\times n}$ is nonsingular if and only if its columns (or rows) are linearly independent

Positive and negative definite matrices

Definition

A symmetric matrix $A \in \mathbb{R}^{n\times n}$ is positive semidefinite if $x^T Ax \geq 0$ for all $x \in \mathbb{R}^n$, and positive definite if $x^T Ax > 0$ for all $x \neq \mathbf{0}$

- matrix A is negative (semi)definite if $-A$ is positive (semi)definite
- if a matrix is neither neg. or pos. sem. def., then it is indefinite
- note: $x \cdot Ax$ is a *quadratic form* and we also talk about definiteness of quadratic forms

Linear equations

- Linear *system* of equations $Ax = b$
  - here $A \in \mathbb{R}^{m\times n}$, and $x \in \mathbb{R}^n$, $b \in \mathbb{R}^m$
  - if there are more equations than unknowns, $m > n$, the system is said to be over-determined
  - note: $Ax = b \Leftrightarrow x_1 A^1 + \dots + x_n A^n = b$
  - system is said to be homogeneous if $b = \mathbf{0}$
- Solutions of linear systems
  - system is solvable if and only if b can be expressed as a linear combination of columns of A
  - in other words, system is solvable if and only if $b \in \text{span}(A^1, \dots, A^n) = \mathcal{R}(A)$ (range of A)
  - the set of solutions of $Ax = \mathbf{0}$ is called the null space of A and it is denoted by $\mathcal{N}(A)$

Linear equations

- rank(A) = dimension of the image space $\{y \in \mathbb{R}^m : y = Ax, x \in \mathbb{R}^n\}$, where $A \in \mathbb{R}^{m\times n}$
- column rank = dimension of span$(A^1, \dots, A^n)$
- row rank = dimension of span$(A_1, \dots, A_m)$
- rank = column rank = row rank
- rank is at most $\min(m, n)$, if rank$(A) = \min(m, n)$ we say that A has full rank
- if rank$(A) = m$ it has full row rank, if rank$(A) = n$ it has full column rank

Linear mappings

Definition

A function $F : V \mapsto W$ is linear if $F(\alpha x + \beta y) = \alpha F(x) + \beta F(y)$ for all $x, y \in V$ and $\alpha, \beta \in \mathbb{R}$

- If a function is of the form $F(x) = L(x) + b$, where $L : V \mapsto W$ is linear and $b \in W$, then F is an affine function
- Example: rotation and scaling
- Kernel of F = ker$(F) = \{v \in V : F(v) = \mathbf{0}\}$
- Image of F = Im$(F) = \{w \in W : \exists v \in V,$ such that $w = F(v)\}$
  - when considering a matrix as a linear function, then kernel coincides with the null space and image with the range

Linear mappings

Theorem

If $F : \mathbb{R}^n \mapsto \mathbb{R}^m$ is a linear function, then there is a matrix $A \in \mathbb{R}^{m\times n}$ such that $F(x) = Ax$ for all $x \in \mathbb{R}^n$

Theorem (Fundamental theorem of linear algebra)

dim(V) = dim(ker(F)) + dim(Im(F))

- for $A \in \mathbb{R}^{m\times n}$ we have $n = \text{dim}(\mathcal{N}(A)) + \text{rank}(A)$

Theorem

If x^0 is a solution of $Ax = b$ then any other solution x can can be written as $x = x^0 + w$ where $w \in \mathcal{N}(A)$

- in other words the set of solutions is the affine set $x^0 + \mathcal{N}(A)$

Linear equations: number of solutions

- Determining the size of the solution set for $Ax = b$ with $A \in \mathbb{R}^{m\times n}$
  - if rank$(A) = m$, then $Ax = b$ has a solution for all $b \in \mathbb{R}^m$
  - $\Leftrightarrow$ A (or $f(x) = Ax$) is surjective if and only if A has full row rank
  - if $n = m$, A is surjective if and only if it is injective
  - if rank$(A) < m$, then $Ax = b$ has a solution only for $b \in \mathcal{R}(A)$
  - if rank$(A) = n$, then $\mathcal{N}(A) = \{\mathbf{0}\}$ and $Ax = b$ has at most one solution solution for all $b \in \mathbb{R}^m$
  - $\Leftrightarrow$ A is injective if and only if $\mathcal{N}(A) = \{\mathbf{0}\}$
  - $\Leftrightarrow$ A is injective if and only if it has full column rank
  - if rank$(A) < n$, then if $Ax = b$ has any solution at all, the set of solutions is an affine subspace of dimension $n -$ rank(A)
  - pos. (neg.) def. matrix is nonsingular

Determinants

- Geometric idea: scale factor for measure
  - $n = 2$, determinant tells how the area is scaled under the linear transformation, $\det(A) = a_{11}a_{22} - a_{12}a_{21}$

Definition (Laplace's formula)

- let A_{ij} denote the matrix obtained from $A \in \mathbb{R}^{n\times n}$ by deleting ith row and jth column
- minor $M_{ij} = \det(A_{ij})$
- cofactor $C_{ij} = (-1)^{i+j} M_{ij}$
- for any row (i) $\det(A) = \sum_{j=1}^{n} a_{ij} C_{ij}$
- for any column (j) $\det(A) = \sum_{i=1}^{n} a_{ij} C_{ij}$

Theorem

A is nonsingular if and only if $\det(A) \neq 0$

Cramer's rule

- Method for solving $Ax = b$
  - let B_i denote the matrix that is obtained from A by replacing ith column with b
  - assuming $\det(A) \neq 0$ the system has a unique solution that is given by $x_i = \det(B_i)/\det(A)$
- Example IS-LM analysis
  - national income model, net national product Y, interest rate r, marginal propensity to save s, marginal efficiency of capital a, investment $I = I^0 - ar$, money balances needed m, government spending G, money supply M_s

$$\begin{aligned} sY + ar &= I^0 + G \\ mY - hr &= M_s - M^0 \end{aligned}$$

Example: IS-LM analysis

- Endogenous variables Y and r can be solved by using Cramer's rule

$$Y = \frac{\begin{vmatrix} I^0 + G & a \\ M_s - M^0 & -h \end{vmatrix}}{\begin{vmatrix} s & a \\ m & -h \end{vmatrix}} = \frac{(I^0 + G)h + a(M - M^0)}{sh + am}$$

$$r = \frac{\begin{vmatrix} s & I^0 + G \\ m & M_s - M^0 \end{vmatrix}}{\begin{vmatrix} s & a \\ m & -h \end{vmatrix}} = \frac{(I^0 + G)m - s(M - M^0)}{sh + am}$$

Determinants and definiteness of a matrix

- kth order leading principal minor of $A \in \mathbb{R}^{n \times n}$ is the determinant of $k \times k$ matrix obtained from A by deleting the last $n - k$ columns and $n - k$ rows
- Symmetric matrix $A \in \mathbb{R}^{n \times n}$ is pos. def if and only if all its n leading principal minors are strictly positive
- Symmetric matrix $A \in \mathbb{R}^{n \times n}$ is neg. def if and only if all its n leading principal minors alternate in sign such that kth order leading principal minor has the same sign as $(-1)^k$

Matrix inversion

- Main approaches
  - direct elimination of variables
  - row reduction (Gaussian elimination and Gauss-Jordan elimination)
  - Cramer's rule
  - more efficient methods for numerical purposes (LU decomposition, iterative methods etc.)
- Number of operations with Gaussian elimination $n^3/3 + n^2 - n/3$, i.e. $O(n^3)$
  - theoretical record (this far) $O(n^{2.376})$
- Note: for solving $Ax = b$ it is not necessary to find A^{-1}

Gaussian elimination

- Idea, let e_i denote ith coordinate vector
  - finding solutions x_i to equations $Ax = e_i$, $i = 1, \ldots, n$

  $\Rightarrow$ $A^{-1} = (x_1 \; x_2 \; \cdots \; x_n)$
- Elementary row operations
  - interchage two rows of a matrix
  - change a row by adding to it a multiple of another row
  - multiply each element in a row by the same nonzero number
- Form an augmented matrix $[A|I]$
  - when solving $Ax = b$ we use $[A|b]$ instead
  - terminology: matrix is in row echelon form if in place of A we have triangular matrix, if we have an identity matrix we have a reduced row echelon form

Gaussian elimination

- Forward elimination
  - using elementary row operations the augmented matrix is reduced to echelon form (this is Gaussian elimination)
  - if the result is degenerate (n zeros in the beginning of some row) there is no inverse (or solution to the equation)
  - note: rank of a matrix = number of non-zero rows in echelon form
- Back substitution
  - bring the echelon form matrix to reduced row echelon form by elem. row operations (if this step is included we call the algorithm as Gauss-Jordan elimination)

  $\rightarrow$ in place of A we have I and in place of I we have A^{-1}

Example

- For what values of $b \in \mathbb{R}^4$ there is a solution for $Ax = b$ where $A = \begin{pmatrix} 1 & 2 & 1 & 4 \\ 0 & 0 & 0 & 2 \\ 2 & 4 & 2 & 2 \\ 0 & 2 & 0 & 3 \end{pmatrix}$
  - the rank of A is three (transform A into echelon form by Gaussian elimination)
  - there is a solution when $b \in \mathcal{R}(A)$
- What about the number of solutions?

Cramer's rule for inversion

- The matrix whose (i, j)-entry is C_{ij} (cofactor) is called as the adjoint of A, adj(A)
- Cramer's rule:

$$A^{-1} = \frac{1}{\det(A)} \cdot \text{adj}(A)$$

- Note: in terms of numbers of operatios Cramer's rule is more demanding than Gaussian elimination

Linear algebra: Eigenvalues

SB 23.1, 23.3, 23.7–23.9

Eigenvalues: The idea

- Idea for a square matrix A
- can we make a change of variables so that instead of A we could consider some other matrix that would be "easier" to handle?
- change of variables: invertible mapping (matrix) P, new variables $P^{-1}x$
- how does A behave when we introduce the new variables: take y (new variables) and find the corresponding x, i.e., $x = Py$, map this with A, i.e., APy, return back to original variables, i.e., $P^{-1}APy$ (image of y in new variables)
- transformed mapping $D = P^{-1}AP$
- previous question: can we find P such that D is a diagonal matrix?
- let λ_i denote the diagonal element in the ith row of D and v_i the ith column vector of P $\rightarrow$ we can deduce that $Av_i = \lambda_i v_i$ (because $PD = AP$)

Eigenvalues

Definition

λ is the eigenvalue of A and $v \neq \mathbf{0}$ is the corresponding eigenvector if $Av = \lambda v$

- Note: it is assumed $v \neq \mathbf{0}$ because $\mathbf{0}$ would always be an eigenvector (and is therefore not particularly interesting)
- Other formulation: λ is an eigenvalue if and only if $A - \lambda I$ is singular $\Leftrightarrow \det(A - \lambda I) = 0$, which is called the characteristic equation (polynomial) of A

Eigenvalues: Example 1

$A = \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix}$ by subtracting $2\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ from A we get $\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$, which is singular, hence, 2 is an eigenvalue and the corresponding eigenvector $v = (v_1, v_2)$ is obtained by solving $\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$, i.e., vectors of the form $v_1 \neq 0$ and $v_2 = 0$ are all eigenvectors corresponding to eigenvalue 2

3 is also an eigenvalue because $A - 3I = \begin{pmatrix} -1 & 0 \\ 0 & 0 \end{pmatrix}$ is singular

Facts about eigenvalue and eigenvectors

- Every non-zero element of the space spanned by eigenvectors corresponding to an eigenvalue is an eigenvector (each eigenvector defines a subspace/eigenspace of eigenvectors)

$\Rightarrow$ if v is an eigenvector, so is αv for all $\alpha \neq 0$

- The eigenvectors corresponding to different eigenvalues are linearly independent
- Diagonal elements of a diagonal matrix are eigenvalues
- Eigenvector determines a direction that is left invariant under A
- Matrix is singular if and only if 0 is its eigenvalue
- If a matrix is symmetric, it has n real eigenvalues
- If $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of $A \in \mathbb{R}^{n \times n}$, then $\lambda_1 + \lambda_2 + \cdots + \lambda_n = \text{trace}(A)$ and $\lambda_1 \cdot \lambda_2 \cdots \lambda_n = \det(A)$.
  - trace(A) is the sum of diagonal elements of A

Eigenvalues: Example 2

$A = \begin{pmatrix} -1 & 3 \\ 2 & 0 \end{pmatrix}$, because this is not a diagonal matrix we need characteristic equation for finding the eigenvalues

$$\begin{aligned} \det(A - \lambda I) &= \det\begin{pmatrix} -1-\lambda & 3 \\ 2 & 0-\lambda \end{pmatrix} = \\ &= -(1+\lambda)(-\lambda) - 6 = \lambda^2 + \lambda - 6 \\ &= (\lambda+3)(\lambda-2) \end{aligned}$$

the eigenvalues are $\lambda_1 = -3$ and $\lambda_2 = 2$. The eigenvector corresponding to λ_1 can be obtained from $(A - (-3)I)v = \begin{pmatrix} 2 & 3 \\ 2 & 3 \end{pmatrix}\begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$, e.g., $v = (-3, 2)$ is an eigenvector (like all vectors αv, $\alpha \neq 0$), what is the other eigenvector?

Eigenvalues: Example 2

$A = \begin{pmatrix} 1 & 0 & 2 \\ 0 & 5 & 0 \\ 3 & 0 & 2 \end{pmatrix}$, characteristic equation

$$\begin{aligned} \det(A - \lambda I) &= \det\begin{pmatrix} 1-\lambda & 0 & 2 \\ 0 & 5-\lambda & 0 \\ 3 & 0 & 2-\lambda \end{pmatrix} = \\ &= (5-\lambda)(\lambda-4)(\lambda+1) \end{aligned}$$

there are three eigenvalues $\lambda_{1,2,3} = 5, 4, -1$, with corresponding eigenvectors $v_{1,2,3} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 2 \\ 0 \\ 3 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}$

Eigenvalues: More facts

- Characteristic equation is a polynomial of degree n

$\rightarrow$ there are n eigenvalues some of which may be complex numbers

- in general, finding the roots of nth degree polynomial is hard

Diagonalization

Definition

$A \in \mathbb{R}^{n \times n}$ is diagonalizable if there is an invertible matrix P such that $D = P^{-1}AP$ is a diagonal matrix

- P can be seen as a change of the basis

Theorem

$A \in \mathbb{R}^{n \times n}$ is diagonalizable if it has n distinct nonzero eigenvalues

- when eigenvalues are distinct and $\neq 0$, then eigenvectors are linearly independent
- eigenvectors are the columns of P and eigenvalues are the diagonal elements of D
- note: matrix can be diagonalizable even when the eigenvalues are not distinct

Diagonalization: Example

- Example 3 (cont'd)
  - transformation $P = \begin{pmatrix} 0 & 2 & 1 \\ 1 & 0 & 0 \\ 0 & 3 & -1 \end{pmatrix}$ and $D = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & -1 \end{pmatrix}$

Information of eigenvalues

- The nature of the linear function defined by A can be sketched by using eigenvalues
  - if an eigenvalue of A is positive, then A scales the vectors in the direction of the corresponding eigenvector
  - if an eigenvalue is negative, A reverses the direction of the corresponding eigenvector and scales
  - example: how does $A = \begin{pmatrix} k_1 & 0 \\ 0 & k_2 \end{pmatrix}$ behave?
  - if there are no real eigenvectors, there is no directions in which A behaves as described above

  $\rightarrow$ the matrix turns all the directions (and possibly scales them)
  - e.g., $A = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$ reverses vectors by angle θ (what are the eigenvalues?)

Information of eigenvalues

- Investigating the definiteness of a matrix by using eigenvalues
  - symmetric $A \in \mathbb{R}^{n \times n}$ is pos.sem.def (neg.sem.def) if and only if the eigenvalues of A are ≥ 0 (≤ 0)
  - respectively, pos. def. (neg. def.) if eigenvalues are > 0 (< 0)
  - matrix is indefinite if and only if it has both positive and negative eigenvalues

Solving linear difference equations

SB 23.2, 23.6

Linear difference equations

- Dynamical system given by $z^{k+1} = Az^k$, $k = 0, 1, \ldots$, where $A \in \mathbb{R}^{n\times n}$ and $z^0 \in \mathbb{R}^n$ is given
  - model for a discrete time process
  - eigenvalues can be utilized in solving this kind of systems
  - later we shall obtain this kind of systems by linearizing nonlinear difference equation
- Example 1: calculating compound interests
  - capital y_k, interest rate ρ, $y_{k+1} = (1+\rho)y_k$
  - beginning from period 0, $y_1 = (1+\rho)y_0$, $y_2 = (1+\rho)^2 y_0$, $\ldots$, we notice that $y_k = (1+\rho)^k y_0$ (solution of the difference equation)

Linear difference equations

- Example 2: model for unemployment
  - employed (on the average) x, unemployed y
  - probability of getting a job p, probability of staying in job q

$$\begin{aligned} x_{k+1} &= qx_k + py_k \\ y_{k+1} &= (1-q)x_k + (1-p)y_k \end{aligned}$$

  in matrix form $\begin{pmatrix} x_{k+1} \\ y_{k+1} \end{pmatrix} = \begin{pmatrix} q & p \\ 1-q & 1-p \end{pmatrix} \begin{pmatrix} x_k \\ y_k \end{pmatrix}$
  - first idea (brute force): let $z^{k+1} = Az^k$ denote the system, find z^N corresponding to z^0 by iterating the system, i.e., $z^N = A^N z^0$, which means that we need A^N

Linear difference equations

- Observations from the 2d model $z^{k+1} = Az^k$, $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$
  - if A was a diagonal matrix, i.e., $b = c = 0$, then the variables (components of z) would be *uncoupled* and we could solve for the components separately as in the compound interest example

  $\rightarrow$ idea: make a transformation of variables which leads to a diagonal system $\rightarrow$ use eigenvalues

Solution by diagonalization

- Let us consider a difference equation $z^{k+1} = Az^k$ with diagonalizable A
  1. transform of variables, $Z = P^{-1}z$ and $z = PZ$
     - old variables $z \in \mathbb{R}^n$ and new variables $Z \in \mathbb{R}^n$
     - find the eigenvalues of A and the corresponding eigenvectors, form P from the eigenvectors
  2. form a difference equation for Z
     $Z^{k+1} = P^{-1}z^{k+1} = P^{-1}Az^k = P^{-1}APZ^k$
     - note: $P^{-1}AP$ is a diagonal matrix with eigenvalues in the diagonal
  3. solve the resulting difference equation for Z
  4. return back to original variables

Example

- Let $A = \begin{pmatrix} 1 & 4 \\ 1/2 & 0 \end{pmatrix}$ and the aim is to solve $z^{k+1} = Az^k$
- Finding the eigenvalues and eigenvectors
  - characteristic equation
    $\det(A - \lambda I) = 0 \Leftrightarrow (1-\lambda)(0-\lambda) - 4 \cdot 1/2 = 0$

  $\rightarrow$ we get the eigenvalues $\lambda_{1,2} = 2, -1$
  - eigenvectors, (*i*) $(A - \lambda_1 I) = \begin{pmatrix} -1 & 4 \\ 1/2 & -2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$, e.g. $v^1 = (4, 1)$

    (*ii*) $(A - \lambda_2 I) = \begin{pmatrix} 2 & 4 \\ 1/2 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$, e.g. $v^2 = (-2, 1)$
  - transformation $P = \begin{pmatrix} 4 & -2 \\ 1 & 1 \end{pmatrix}$ and $P^{-1} = \begin{pmatrix} 1/6 & 1/3 \\ -1/6 & 2/3 \end{pmatrix}$

Example

- Transform of variables (change of basis)
  - let us denote $z = (x, y)$, new variables X and Y, $Z = (X, Y)$:
    $\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} 1/6 & 1/3 \\ -1/6 & 2/3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$ and $\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 4 & -2 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix}$
- Deriving the difference equation for Z
  - $Z^{k+1} = P^{-1}APZ^k$ and $P^{-1}AP$ is a diagonal matrix with eigenvalues on the diagonal; $P^{-1}AP = \begin{pmatrix} 2 & 0 \\ 0 & -1 \end{pmatrix}$
  - in this difference equation X and Y are uncoupled!

Example

- Solve for Z^k: $X_k = 2^k X_0$ and $Y_k = (-1)^k Y_0$
- Going back to original variables

$$\begin{aligned} z^k = PZ^k &= \begin{pmatrix} 4 & -2 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 2^k X_0 \\ (-1)^k Y_0 \end{pmatrix} \\ &= \begin{pmatrix} 4 \cdot 2^k X_0 - 2 \cdot (-1)^k Y_0 \\ 2^k X_0 + (-1)^k Y_0 \end{pmatrix} = (2^k X_0) \begin{pmatrix} 4 \\ 1 \end{pmatrix} + ((-1)^k Y_0) \begin{pmatrix} -2 \\ 1 \end{pmatrix} \end{aligned}$$

- Initial values X_0 and Y_0 can be obtained by $Z^0 = P^{-1}z^0$

Results for linear systems

- Assume that A is diagonalizable with real eigenvalues
  - eigenvalues $\lambda_1, \ldots, \lambda_n$ and eigenvectors $v^1, \ldots, v^2$
  - $D = \begin{pmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{pmatrix}$, in which case $D^k = \begin{pmatrix} \lambda_1^k & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n^k \end{pmatrix}$

Theorem

When the initial value is z^0, the solution of the difference equation is $z^k = PD^kP^{-1}z^0$

- The result follows by observing that $A^k = PD^kP^{-1}$ and plug this into $z^k = A^kz^0$
  - note: with the formula for A^k we can define, e.g., $\sqrt{A}$ as a series

Results for linear systems

Theorem

The general solution of the difference equation $z^{k+1} = Az^k$ ($z^k \in \mathbb{R}^n$) is $z^k = c_1\lambda_1^k v^1 + c_2\lambda_2^k v^2 + \ldots + c_n\lambda_n^k v^n$, where c_i, $i = 1, \ldots, n$, are constants

- In the general solution z^0 is unspecified
- Regardless of the initial value z^0 the solution is of the previous form
  - constants c_i, $i = 1 \ldots, n$, can be solved when z^0 is given; denote $c = (c_1, \ldots, c_n)$, then $c = P^{-1}z^0$

Complex eigenvalues

- 2d intuition
  - complex eigenvalues means that A reverses all directions, .i.,e no direction is mapped to the subspace defined by the direction
- As in the case of real eigenvalue, we obtain exactly the same solution for the difference equation but now the matrices P, D will be complex matrices
  - for $n = 2$ we get (by using the rules of addition and multiplication of complex numbers) a general solution in the form $z^k = 2r^k[(c_1 \cos k\theta - c_2 \sin k\theta)u - (c_2 \cos k\theta + c_1 \sin k\theta)v]$, where c_1 and c_2 are real numbers determined by z^0 and the eigenvalues are $\alpha \pm \beta i$ and the eigenvectors are $u \pm iv$
  - complex eigenvalues imply oscillations

Stability of linear systems

Definition

Origin is (globally asymptotically) stable equilibrium of a system $z^{k+1} = Az^k$ if $z^k \to \mathbf{0}$ for all z^0

- note: there may be other equilibria than origin (when?)

Theorem

Origin is stable when the eigenvalues are inside a unit circle in complex plane, i.e., $r = \sqrt{a^2 + b^2} < 1$ if $a \pm bi$ are eigenvalues

- in case of real eigenvalues, stability is obtained when the eigenvalues less than 1 in absolute values

Markov processes

Definition

let us assume that there is a finite number of states $I = \{1, \ldots, n\}$. Stochastic process is a rule that gives the probability that the system will be in state $i \in I$ at time $k + 1$ given the probabilities of its being in the various states in previous periods

- when (for all $i \in I$) the probability depends only on what state the system was in at time k the process is called as a Markov process

- Example
  - let us classify households as urban (1), suburban (2), or rural (3)
  - the different classes can be considered as states $1, 2, 3$

Markov processes: state transitions

- Let $x_i(k)$ be the probability that in period k the system is at state i
- State transition probabilities m_{ij} = probability that in period $k + 1$ the state is i when the state was j in period k
- collecting the probabilities in a Markov matrix
  $M = \begin{pmatrix} m_{11} & \cdots & m_{1n} \\ \vdots & \ddots & \vdots \\ m_{n1} & \cdots & m_{nn} \end{pmatrix}$
- Example (cont'd.)
  - we may think that the probability $x_i(k)$ is the portion of populations in are i in period k
  - e.g., m_{1j} = probability of staying urban ($j = 1$) or becoming urban ($j = 2$ or $j = 3$),
  - set $M = \begin{pmatrix} 0.75 & 0.02 & 0.1 \\ 0.2 & 0.9 & 0.2 \\ 0.05 & 0.08 & 0.7 \end{pmatrix}$

Markov process as a difference equation

- Probability that in period $k + 1$ the state is i: $x_i(k + 1)$= Prob(transition from state 1 to state i) $\times$ Prob(initial state is 1) $+ \ldots +$ Prob(transition from state n to state i) $\times$ Prob(initial state is n) $= \sum_j m_{ij}x_j(k)$
- using the state transition matrix we get $x(k + 1) = Mx(k)$, i.e.,

$$\begin{pmatrix} x_1(k+1) \\ \vdots \\ x_n(k+1) \end{pmatrix} = \begin{pmatrix} m_{11} & \cdots & m_{1n} \\ \vdots & \ddots & \vdots \\ m_{n1} & \cdots & m_{nn} \end{pmatrix} \begin{pmatrix} x_1(k) \\ \vdots \\ x_n(k) \end{pmatrix}$$

Example

- Markov matrix $M = \begin{pmatrix} 0.9 & 0.4 \\ 0.1 & 0.6 \end{pmatrix}$
- Eigenvalues, $\det(M - \lambda I) = (0.9 - \lambda)(0.6 - \lambda) - 0.1 \cdot 0.4 = \lambda^2 - (0.9 + 0.6)\lambda + 0.9 \cdot 0.6 - 0.04 = 0$, we get $\lambda_{1,2} = (1.5 \pm \sqrt{1.5^2 - 4 \cdot 0.5})/2 = (1.5 \pm 0.5)/2 = 1, 0.5$
- Eigenvectors $v_1 = (4, 1)$ and $v_2 = (1, -1)$
- General solution

$$\begin{aligned} x_1(k+1) &= c_1 \cdot 4 \cdot 1^k + c_2 \cdot 1 \cdot 0.5^k \\ x_2(k+1) &= c_1 \cdot 1 \cdot 1^k + c_2 \cdot (-1) \cdot 0.5^k \end{aligned}$$

Example

- Remarks
  - one eigenvalue is one and the other is less than one
  - when $k \to \infty$, the limit is $c_1(4, 1)$ (direction corresponding to the eigenvector for eigenvalue 1), because the limit should be a probability distribution $4c_1 + c_1 = 1$, i.e., $c_1 = 1/5$
  - general result: when M is a Markov matrix with all components positive (or M^k has positive components for some k), then one of the eigenvalues is one and the rest are on the interval $(-1, 1)$, and the process converges to the limit distribution determined by the eigenvector corresponding to the eigenvalue 1

  $\to$ the limit distribution is a stable equilibrium of the difference equation

Linear differential equations

SB: 25.2, 25.4

Scalar equations

- kth order differential equation is an equation of the form $F(y^{[k]}, y^{[k-1]}, \ldots, y^{[1]}, y, t) = 0$, where $y^{[j]} = d^j y(t)/dt^j$ (and $y(t) \in \mathbb{R}$)
  - model for a continuous time process or dynamical system (time t)
  - when $F : \mathbb{R}^{k+1} \mapsto \mathbb{R}$ is linear in first k arguments the differential equation is linear
- First order scalar equation $dy(t)/dt = a(t)y$
  - notation $\dot{y} = dy(t)/dt$
  - if $a(t) = a \ \forall t$ the equation is autonomous
  - general solution $y(t) = Ce^{\int^t a(s)ds}$ where C is an integration constant

Scalar equations

- Second order linear scalar equation $a\ddot{y} + b\dot{y} + cy = 0$
  - example: what is the class of utility functions for which $-u''(x)/u'(x) = a$ (the left hand side is Arrow-Pratt measure of absolute risk aversion) $\Leftrightarrow u''(x) + au'(x) = 0$
  - the solution can be found by using an educated guess (ansatz) $y = e^{\lambda t}$

Linear systems

- Linear system of differential equations is of the form $\dot{x} = A(t)x + b(t)$, where $t \in \mathbb{R}$ (time) $x(t), b(t) \in \mathbb{R}^n$, and $A(t) \in \mathbb{R}^{n \times n}$
- Homogeneous autonomous system $A(t) = A$ and $b(t) = 0$ for all t:

$$\begin{aligned} \dot{x}_1 &= a_{11}x_1 + \ldots + a_{1n}x_n \\ &\vdots \\ \dot{x}_n &= a_{n1}x_1 + \ldots + a_{nn}x_n \end{aligned}$$

  - if the question is of finding the solution for a given initial value $x(t_0) = x^0$ the problem is an initial value problem

Linear systems

- From second order scalar equation to a linear system
  - take a new variable $z = \dot{y}$, when the differential equation becomes $a\dot{z} + bz + cy = 0$, i.e., we have a pair $\dot{z} = -bz/a - cy/a$, $\dot{y} = z$
  - more generally, by introducing new variables we can transform higher order scalar equations into linear systems

Solution by diagonalization

Theorem

When A is diagonalizable and its eigenvalues λ_i, $i = 1, \ldots, n$ are distinct and non-zero, the general solution of $\dot{x} = Ax$ is $x(t) = C_1 e^{\lambda_1 t} v^1 + C_2 e^{\lambda_2 t} v^2 + \ldots + C_n e^{\lambda_n t} v^n$, where C_i are constants, and v^i are the eigenvectors corresponding to the eigenvalues ($i = 1, \ldots, n$)

- The solution of the initial value problem is $x(t) = Pe^{Dt}P^{-1}x(0)$
  - note: $Pe^{Dt}P^{-1} = e^{At}$, i.e., the solution is of the same form as for scalar equations ($x(t) \in \mathbb{R}$)

Solution by diagonalization

- Example: $\dot{x} = Ax$ with $A = \begin{pmatrix} 1 & 4 \\ 1 & 1 \end{pmatrix}$
  - finding the eigenvalues: $\det(A - \lambda I) = (1-\lambda)(1-\lambda) - 4 = 0$, i.e., $\lambda^2 - 2\lambda - 3 = 0$, we get $\lambda_{1,2} = (2 \pm \sqrt{2^2 - 4(-3)})/2 = 1 \pm 2 = 3, -1$
  - eigenvectors: $(1, 0.5)$ and $(1, -0.5)$
  - the general solution:

$$\begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} = c_1 e^{3t} \begin{pmatrix} 1 \\ 0.5 \end{pmatrix} + c_2 e^{-t} \begin{pmatrix} 1 \\ -0.5 \end{pmatrix}$$

  - what if the initial value was $x(0) = (1, 2)$?

Stability

- Origin $x = \mathbf{0}$ is the steady state of $\dot{x} = Ax$

Definition

$x = \mathbf{0}$ is globally asymptotically stable steady state of $\dot{x} = Ax$, if the solution $\lim_{t \to \infty} x(t; x^0) = \mathbf{0}$ for all x^0

  - here $x(t; x^0)$ is the solution of the system for initial state x^0
- Stability by using eigenvalues
  - if the real parts of eigenvalues are negative, then origin is as. stable
  - note: some of the eigenvalues may be complex

Multivariate calculus

Differentiation, Taylor series, linearization
SB: 14–15, 30

Motivation

- Need for nonlinear models
  - need to model nonlinear phenomena, e.g., decreasing marginal profits
  - linear models may lead to optimization problems with no (bounded) solutions, even when there are solutions, the solutions may behave against economic intuition (e.g., discontinuities)
- Nonlinear models are often hard solve analytically
  - numerical solution
  - local approach: linearize the model around equilibrium, optimum or other interesting point $\rightarrow$ differentiation

Differentiation in $\mathbb{R}$

Definition

A function $f : \mathbb{R} \mapsto \mathbb{R}$ is differentiable at $x \in \mathbb{R}$ if there is a number $f'(x)$ such that

$$\lim_{h \to 0} \frac{|f(x+h) - f(x)|}{|h|} = f'(x)$$

- rewriting we get $f(x+h) - f(x) = f'(x)h + o(h)$, where $o(h)$ is a function such that $o(h)/h \to 0$ as $h \to 0$
- around x, i.e., at $x+h$, function f behaves as a linear function $f'(x)h$ (plus the error $o(h)$)
- $f'(x)$ is the derivative of f at x and $f'(x)h$ is the differential
- note: if f is differentiable for all $x \in \mathbb{R}$ then $f'(x)$ is a function
- if $f'(x)$ is differentiable at x we obtain the second derivative $f''(x)$ (higher order derivatives respectively)

Differentiation in $\mathbb{R}^n$

Definition

A function $f : \mathbb{R}^n \mapsto \mathbb{R}^m$ is differentiable at $x \in \mathbb{R}^n$ if there is a linear function $Df(x) : \mathbb{R}^n \mapsto \mathbb{R}^m$ such that

$$\lim_{h \to \mathbf{0}} \frac{\|f(x+h) - f(x) - [Df(x)]h\|}{\|h\|} = 0$$

- $Df(x)$ is the derivative (or total derivative or differential) of f at x
- note: in the definition of the above limit, $Df(x)$ is independent on how h goes to zero, e.g., for $f(x) = |x|$ we have $f(0+h) - f(0) = h$ for all $h > 0$ but f is not differentiable at $x = 0$ because for $h < 0$ we have $f(0+h) - f(0) = -h$, i.e., there is no $Df(x)$ as required in the definition

Differentiation in $\mathbb{R}^n$

Theorem

Derivative of a differentiable function is unique

- Chain rule: suppose f maps an open set $S \subseteq \mathbb{R}^n$ into $\mathbb{R}^m$ and is differentiable at $x^0 \in S$, g maps an open set containing $f(S)$ into $\mathbb{R}^k$ and is differentiable at $f(x^0)$, then $F(x) = g(f(x))$ is differentiable at x^0 and $DF(x^0) = Dg(f(x^0))Df(x^0)$

Partial derivatives

- Partial derivatives: assume that $f = (f_1, \ldots, f_m)$, for $x \in \mathbb{R}^n$, $i = 1, \ldots, m$, $j = 1, \ldots, n$ we define (provided the limit exists)

$$D_j f_i(x) = \lim_{h \to 0} \frac{f_i(x + he_j) - f_i(x)}{h},$$

  - $D_j f_i$ is the derivative of f_i w.r.t. x_j, we call it a partial derivative of f w.r.t x_j and write

$$D_j f_i(x) = \frac{\partial f_i(x)}{\partial x_j}$$

Partial derivatives

- If f is differentiable at x, then all partial derivatives exist at x and the (i,j)-component of $Df(x)$ is $D_j f_i(x)$, we write $Df(x) = [D_j f_i(x)]$ and call this matrix as the Jacobian matrix
- If partial derivatives are continuous then function is differentiable
- When there are several argument vectors with respect to which we can compute the Jacobian we use the subscript to denote the variables, e.g., $f(x, a)$, Jacobian w.r.t. x is $D_x f(x, a)$

Continuous differentiability

- A function that has continuous partial derivatives in an open set containing x is said to be continuously differentiable at x
  - k-times continuously differentiable function has continuous kth order partial derivatives
  - if f is k-times continuously differentiable on S we denote $f \in C^k(S)$ (note: k can be infinite)

Gradient of a function

- Gradient of f at x is $\nabla f(x) = Df(x)^T$
  - geometric interpretation: $y - f(\hat{x}) = \nabla f(\hat{x}) \cdot (x - \hat{x})$ is a hyperplane in $\mathbb{R}^{n+1}$
  - $\nabla f(x)$ is perpendicular (orthogonal) to the tangent space of $\{y \in \mathbb{R}^n : f(y) = f(x)\}$ (level set)
- Directional derivative of f at x to direction v is (if the limit exists)

$$f'(x; v) = \lim_{h \to 0} \frac{f(x + hv) - f(x)}{h}$$

  - for differentiable $f : \mathbb{R}^n \mapsto \mathbb{R}$ we have $f'(x; v) = \nabla f(x) \cdot v = \nabla f(x)^T \cdot v$
  - if we let v with $\|v\| = 1$ vary we notice that maximum of $f'(x; v)$ is attained at $v = \nabla f(x)/\|\nabla f(x)\|$, i.e., $\nabla f(x)$ is the direction of steepest ascent at x

Taylor's theorem

- Motivation: sometimes it is not enough to linearize a nonlinear system but more information is needed $\rightarrow$ higher order approximation is needed
- Purpose: approximate function of a single variable around $a \in \mathbb{R}$
- Notations
  - $k!$ is the factorial of k ($0! = 1$)
  - $f^{(k)}$ is the kth derivative of f
  - R_n is the remainder (function) such that there is ξ on the interval joining x and a, such that

$$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x - a)^{n+1}$$

Taylor's theorem

Theorem

If $n \geq 0$ is an integer and f a function which is n times continuously differentiable on the closed interval $[a, x]$ and $n + 1$ times differentiable on the open interval (a, x), then

$$f(x) = \sum_{k=0}^{n} \frac{f^{(k)}(a)}{k!}(x - a)^k + R_n(x),$$

- for $n = 0$ the result is known as the mean value theorem
- note: there are also other expressions for $R_n(x)$

Taylor's theorem: multivariate case

Theorem

Let $f : \mathbb{R}^n \mapsto \mathbb{R}$ be $(m + 1)$-times continuously differentiable on $\bar{N}_r(a) \subset \mathbb{R}^n$, for any $x \in N_r(a)$

$$f(x) = \sum_{k=0}^{m} \frac{f^{(k)}(a; x - a)}{k!} + R_m(x),$$

- *where $f^{(1)}(a; t) = \sum_i D_i f(x) t_i$, $f^{(2)}(a; t) = \sum_{i,j} D_{ij} f(x) t_i t_j$, $f^{(3)}(a; t) = \sum_{i,j,k} D_{ijk} f(x) t_i t_j t_k$, ..., and there is z on the line between x and a such that*

$$R_m(x) = \frac{f^{(m+1)}(z; x - a)}{(m + 1)!}$$

- *note: D_{ij} is the second (partial) derivative of f with respect to x_i and x_j, D_{ijk} and higher orders respectively*

Linearizations

- According to Taylor's theorem linear function $\hat{f}(x) = f(a) + \nabla f(a) \cdot (x - a)$ approximates f around a with the error term $R_1(x)$
  - $\hat{f}(x)$ is the linearization of f at a, or a first order approximation
  - if we have a relation $y = f(x)$, then around a the change of y is given by the linear term, i.e., $y - f(a) \approx \nabla f(a) \cdot (x - a)$
- In many economic models log-linearization provides more attractive interpretations than standard (above) linearization
  - reason: expressions are obtained in terms of elasticities

Example: Arrow-Pratt

- Deriving the Arrow-Pratt measure of absolute risk aversion (ARA)
- Finding an approximation for a certain loss that a decision maker with utility function u is willing take instead of a lottery
  - ε is random variable zero mean and variance σ^2
  - condition for x at w: $\mathbb{E}[u(w + \varepsilon)] = u(w - x)$
  - by Taylor's theorem $u(w + \varepsilon) \approx u(w) + \varepsilon u'(w) + \varepsilon^2 u''(w)/2$ (second order approximation) and $u(w - x) \approx u(w) - x u'(w)$ (first order approximation)
  - observing that $\mathbb{E}[\varepsilon u'(w) + \varepsilon^2 u''(w)/2] = 0 + \sigma^2 u''(w)/2$ (note $\mathbb{E}[\varepsilon^2] = \sigma^2$)and manipulating gives $x \approx -(\sigma^2/2)[u''(w)/u'(w)]$
  - the second term is ARA

Log-linearizations

- The usual way: make a log-transform and linearize

1. if the original variable is X take log-transform $x = \log X$, then $X = e^x$ (note $X > 0$)
2. plug e^x in place of X (if there are several variables do the same for each of them)
3. linearize $f(e^x)$ around $X^* = e^{x^*}$
   - the linearization gives $\hat{f}(x) = f(X^*) + X^* f'(X^*)(x - x^*)$
   - the change of X is now relative, i.e., $x - x^* = \log(X) - \log(X^*) = \log(X/X^*)$ and $X/X^* = 1+\%$ change
   - the relative change of $y = f(X)$ has an approximation $[f(X) - f(X^*)]/f(X^*) \approx [X^* f'(X^*)/f(X^*)](x - x^*) = \varepsilon(x - x^*)$, where ε is the elasticity

Log-linearizations

- Multivariate case: $X \in \mathbb{R}^n_{++}$, and $f : \mathbb{R}^n_{++} \mapsto \mathbb{R}$, then $\hat{f}(x) = f(X^*) + \sum_i X_i^* [\partial f(X^*)/\partial X_i](x_i - x_i^*)$ (turn this into relative changes form!)
- Examples
  - $X \approx X^*(1 + x - x^*)$
  - resource constraint of the economy $Y = C + I$, $(y - y^*) \approx [C^*/Y^*](c - c^*) + [I^*/Y^*](i - i^*)$ (log-linearize both sides and manipulate)
  - consumption Euler equation: $1 = R_{t+1}\beta[C_{t+1}/C_t]^{-\gamma}$, log-linearization around steady state R^*, C^*, i.e., $R_t = R^*$ and $C_{t+1} = C_t = C^*$, leads to $(c_t - c^*) \approx -[1/\gamma](r_{t+1} - r^*) + (c_{t+1} - c^*)$ (linear relation)

Hessian matrix

- Hessian matrix of a twice differentiable function $f : \mathbb{R}^n \mapsto \mathbb{R}$
  - the matrix
$$D^2 f(x) = [D_{ij} f(x)] = \begin{pmatrix} \partial^2 f(a)/\partial x_1 \partial x_1 & \cdots & \partial^2 f(a)/\partial x_1 \partial x_n \\ \partial^2 f(a)/\partial x_2 \partial x_1 & \cdots & \partial^2 f(a)/\partial x_2 \partial x_n \\ \vdots & \cdots & \vdots \\ \partial^2 f(a)/\partial x_n \partial x_1 & \cdots & \partial^2 f(a)/\partial x_n \partial x_n \end{pmatrix}$$
  - Taylor's theorem
$$f(x) = f(a) + \nabla f(a) \cdot (x-a) + (1/2)(x-a) \cdot D^2 f(a)(x-a) + R_2(x)$$
  - this gives second order (quadratic) approximation of f at a
  - when $f \in C^2(N_\varepsilon(a))$ then $D^2 f(a)$ is symmetric

Implicit function theorem

- When the endogenous (y) and exogenous (x) variables cannot be separated as in $f(x, y) = c$ the following questions arise
  - does f define y as a continuous function of x (we denote $y(x)$) around a given point x^0 such that $y(x^0) = y^0$ and $f(x, y(x)) = c$?
  - how does the change in x affect the corresponding y's (comparative statics)?
  - note: there has to be as many equations as there are endogenous variables
- The case $f : \mathbb{R}^2 \mapsto \mathbb{R}$ (and $f \in C^1$)
  - assume $f(x^0, y^0) = c$, sufficient condition for the existence of an implicit function is that $\partial f(x^0, y^0)/\partial y \neq 0$
  - If $y(x)$ is the implicit function the its derivative at x^0 can be obtained by applying the chain rule
$$y'(x^0) = -\frac{\partial f(x^0, y^0)/\partial x}{\partial f(x^0, y^0)/\partial y}$$

Linear implicit function theorem

Theorem

- *Assume that $A \in \mathbb{R}^{m \times (n+m)}$, and $A = (A_x, A_y)$, with $A_x \in \mathbb{R}^{m \times n}$, $A_y \in \mathbb{R}^{m \times m}$ such that $A(x, y) = A_x x + A_y y$ (here $(x, y) = \begin{pmatrix} x \\ y \end{pmatrix}$)*
- *If A_y is invertible we obtain from $A(x, y) = \mathbf{0}$ the function $y(x) = -A_y^{-1} A_x x$ (i.e., $Dy(x) = -A_y^{-1} A_x$)*
- *If we write $A_x dx + A_y dy = 0$ we obtain $dy = -A_y^{-1} A_x dx$, when only one exogenous variable is changed, then we can use Cramer's rule to find dy*

Implicit function theorem

Theorem

$f : \mathbb{R}^{n+m} \mapsto \mathbb{R}^m$ continuously differentiable on $N_\varepsilon(x^0, y^0)$ for some $\varepsilon > 0$ and $f(x^0, y^0) = \mathbf{0}$, $\det(D_y f(x^0, y^0)) \neq 0$, then there is $\delta > 0$ and function $y(x) \in C^1(N_\delta(x^0))$ such that

1. $f(x, y(x)) = \mathbf{0}$
2. $y(x^0) = y^0$
3. $D_x y(x^0) = -(D_y f(x^0, y^0))^{-1} D_x f(x^0, y^0)$

IFT: Example 1

- $f(x, y) = \begin{pmatrix} f_1(x, y) \\ f_2(x, y) \end{pmatrix}$, $f_1(x, y) = y_1 y_2^2 - x_1 x_2 + x_2 - 7$, $f_2(x, y) = y_1 - x_1/y_2 + x_2 - 5$
- Let us take $(x^0, y^0) = (-2, 2, 1, 1)$ $(x_1^0 = -2, x_2^0 = 2, y_1^0 = 1, y_2^0 = 1)$
$$D_y f(x^0, y^0) = \begin{pmatrix} \frac{\partial f_1(x^0, y^0)}{\partial y_1} & \frac{\partial f_1(x^0, y^0)}{\partial y_2} \\ \frac{\partial f_2(x^0, y^0)}{\partial y_1} & \frac{\partial f_2(x^0, y^0)}{\partial y_2} \end{pmatrix} = \begin{pmatrix} (y_2^0)^2 & 2 y_1^0 y_2^0 \\ 1 & x_1^0/(y_2^0)^2 \end{pmatrix} = \begin{pmatrix} 1 & 2 \\ 1 & -2 \end{pmatrix}$$
$$D_x f(x^0, y^0) = \begin{pmatrix} \frac{\partial f_1(x^0, y^0)}{\partial x_1} & \frac{\partial f_1(x^0, y^0)}{\partial x_2} \\ \frac{\partial f_2(x^0, y^0)}{\partial x_1} & \frac{\partial f_2(x^0, y^0)}{\partial x_2} \end{pmatrix} = \begin{pmatrix} -(x_2^0) & 1 - x_2^0 \\ -1/y_2^0 & 1 \end{pmatrix} = \begin{pmatrix} -2 & -1 \\ -1 & 1 \end{pmatrix}$$

IFT: Example 1

- How does change in x_1 affect the endogenous variables y?
$$D_y f(x^0, y^0) \begin{pmatrix} dy_1 \\ dy_2 \end{pmatrix} + D_{x_1} f(x^0, y^0) dx_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$
  we get
$$\begin{pmatrix} 1 & 2 \\ 1 & -2 \end{pmatrix} \begin{pmatrix} dy_1 \\ dy_2 \end{pmatrix} + \begin{pmatrix} -2 \\ -1 \end{pmatrix} dx_1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$
  by Cramer's rule
$$dy_1 = \frac{\det \begin{pmatrix} 2 & 2 \\ 1 & -2 \end{pmatrix}}{\det \begin{pmatrix} 1 & 2 \\ 1 & -2 \end{pmatrix}} dx_1 = \frac{1}{2} dx_1$$
$$dy_2 = \frac{\det \begin{pmatrix} 1 & 2 \\ 1 & 1 \end{pmatrix}}{\det \begin{pmatrix} 1 & 2 \\ 1 & -2 \end{pmatrix}} dx_1 = \frac{1}{4} dx_1$$

Example 2

- Nonlinear IS-LM
$$\begin{aligned} Y - C(Y - T) - I(r) - G &= 0, \\ M(Y, r) - M^s &= 0, \end{aligned}$$
  - $C'(\cdot) \in (0, 1)$, $I'(r), M_r = \partial M/\partial r < 0$, $M_Y = \partial M/\partial Y > 0$ (why?)
  - write this as $f(G, T, M^s, Y, r) = \mathbf{0}$
  - assume that G, T, M^s are exogenous and there is an equilibrium around which
$$D_{(Y,r)} f(\cdot) = \begin{pmatrix} 1 - C'(\cdot) & -I'(r) \\ M_Y(\cdot) & M_r(\cdot) \end{pmatrix}$$
  - we get $\det(D_{(Y,r)} f(\cdot)) = [1 - C'(\cdot)] M_r(\cdot) + I'(r) M_Y(\cdot) < 0$, i.e., (Y, r) can be defined as functions of exogenous variables around equilibrium

Example 2

- What happens to equilibrium when M^s is kept fixed and G and T are raised equally much ("$dT = dG$")?
$$\begin{pmatrix} 1 - C'(\cdot) & -I'(r) \\ M_Y(\cdot) & M_r(\cdot) \end{pmatrix} \begin{pmatrix} dY \\ dr \end{pmatrix} = \begin{pmatrix} dG - C'(\cdot) dT \\ 0 \end{pmatrix}$$
  - Cramer's rule:
$$\begin{aligned} dY/dG &= \frac{\det \begin{pmatrix} 1 - C'(\cdot) & -I'(r) \\ 0 & M_r(\cdot) \end{pmatrix}}{\det(D_{(Y,r)} f(\cdot))} > 0 \\ dr/dG &= \frac{\det \begin{pmatrix} 1 - C'(\cdot) & 1 - C'(\cdot) \\ M_Y(\cdot) & 0 \end{pmatrix}}{\det(D_{(Y,r)} f(\cdot))} > 0 \end{aligned}$$

Total derivative

- Let $f : \mathbb{R}^n \mapsto \mathbb{R}$. What does the following expression mean?

$$df = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} dx_i \tag{1}$$

  - heuristically, we think about relations between infinitesimals
  - the benefit is that we can easily account for relations like $x_1^2 = x_2 x_3$, i.e., $2x_1 dx_1 = x_3 dx_2 + x_2 dx_3$
  - the expression is total derivative or differential form
- Formally (1) can be defined at $p \in \mathbb{R}^n$ by thinking of x_i's as functions on a neighborhood of p and

$$f'(p; v) = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} x_i'(p; v)$$

Total derivative

- The result obtained in example 2 using total derivatives can be obtained by assuming that $T(G)$ satisfies $T'(G) = 1$ or $T = G + c$

$$\begin{pmatrix} dY/dG \\ dr/dG \end{pmatrix} = -[D_{(Y,r)} f(G, T(G), M^s, Y, r)]^{-1} D_G f(\cdot)$$

$$= -\frac{1}{\det(D_{(Y,r)} f(G, T(G), M^s, Y, r))} \begin{pmatrix} M_r(\cdot) & I'(r) \\ -M_Y(\cdot) & 1 - C'(\cdot) \end{pmatrix} \begin{pmatrix} C'(\cdot) - 1 \\ 0 \end{pmatrix}$$

Inverse function theorem

Theorem

$f : \mathbb{R}^n \mapsto \mathbb{R}^n$ continuously differentiable on an open set S. If $\det(Df(x^0)) \neq 0$ then inverse function f^{-1} exists on an open neighborhood of $f(x^0)$ (f is a bijection around x^0) and $Df^{-1}(f(x^0)) = (Df(x^0))^{-1}$

- checking directly that a given nonlinear function is a bijection is extremely difficult
- for bijectivity we only need to show the non-singularity of the derivative which is an easy task

Local stability of steady-states

non-linear dynamic systems, stability by linearization
SB: 23.2, 25.4.

Dynamical systems

- Dynamical systems
  - time (discrete or continuous) and a state (in state space)
  - rule describing the evolution of the state: how future state follows from past states
- Nonlinear systems

  → it may be impossible to find an analytic solution
  → instead of finding solutions the main interest is in steady states and their stability

Difference equations

- A sequence $\{x_k\}_{k=0}^{\infty} \subset \mathbb{R}^n$ satisfies a difference equation of order T if $G(x_k, x_{k-1}, \ldots, x_{k-T}) = \mathbf{0}$ (for all $k \geq T$), where $G : \mathbb{R}^{Tn} \mapsto \mathbb{R}^n$
  - when a difference equation is given in a first order form we have a recursive presentation, i.e., $x^{k+1} = F(x^k)$
  - in this case the solution of a difference equation is the sequence that is obtained from the recursion when the initial value x^0 is given
  - the recursion is also called as the state transition equation, law of motion or dynamic system

Example: National income

- period k national income Y_k satisfies $Y_k = C_k + I_k + G_k$, where C_k is the consumers expenditures, I_k is investments, and G_k is public expenditures
- assume $C_k = \alpha Y_{k-1}$, $G_k = G_0$ for all k, $I_k = \beta(C_k - C_{k-1})$
- we obtain a difference equation
  $Y_k = \alpha Y_{k-1} + \beta(C_k - C_{k-1}) + G_0 =$
  $\alpha(1+\beta) Y_{k-1} - \alpha\beta Y_{k-2} + G_0$
- this is a second order equation
- by choosing $x_k = Y_k$ and $z_k = Y_{k-1}$ as variables we get the difference equation in the recursive form

$$\begin{aligned} x_k &= \alpha(1+\beta)x_{k-1} - \alpha\beta z_{k-1} + G_0 \\ z_k &= x_{k-1} \end{aligned}$$

Other examples

- Economic growth
  - production function $f(K_k, L_k)$, K_k capital, L_k labor, C_k consumption, δ decay of capital
    $K_{k+1} = f(K_k, L_k) + (1-\delta)K_k - C_k$
- Natural resources
  - resource stock s_k, harvest of the resource x_k, growth function $f(s)$, dynamics $s_{k+1} = f(s_k) - x_k$

First order difference equations

Definition

$x^* \in \mathbb{R}^n$ is steady state (equilibrium) of a difference equation $x^{k+1} = F(x^k)$, $F : \mathbb{R}^n \mapsto \mathbb{R}^n$ if $x^* = F(x^*)$

- non-linear difference equation can be linearized around steady state: $F(x) \approx F(x^*) + DF(x^*)(x - x^*)$, with the change of variables $z = x - x^*$ we obtain a linear difference equation $z^{k+1} = Az^k$ with $A = DF(x^*)$

→ it is possible to analyze local stability of an equilibrium by analyzing the resulting linear difference equation

Stability

Definition

x^* is locally asymptotically stable if there is $\varepsilon > 0$ such that $x^0 \in N_\varepsilon(x^*)$ implies that $x^k \to x^*$ as $k \to \infty$, $x^k = F(x^{k-1})$ for $k \geq 1$

- x^* is globally (as.) stable if $x^k \to x^*$ for all x^0
- for a linear difference equations $z^{k+1} = Az^k$ we are usually interested in the stability of the origin

Theorem

If $I - DF(x^*)$ is non-singular and the eigenvalues of $DF(x^*)$ are inside a unit circle then x^* is locally asymptotically stable

Nonlinear differential equations

- Consider first order autonomous nonlinear system $\dot{x} = F(x)$, where $F : \mathbb{R}^n \mapsto \mathbb{R}^n$
  - general solution and the solution to an initial value problem are defined similarly as for linear systems and difference equations

Definition

x^* is a steady state (equilibrium) of the system $\dot{x} = F(x)$ if $F(x^*) = \mathbf{0}$

- the equilibrium need not be unique

Stability

Definition

x^* is asymptotically stable equilibrium of $\dot{x} = F(x)$ on $S \subseteq \mathbb{R}^n$, if $x(t; x^0) \to x^*$, when $t \to \infty$ for all $x^0 \in S$

- Here $x(t; x^0)$ is the solution of the initial value problem for x^0
- If there is $\varepsilon > 0$ such that x^* is stable on $N_\varepsilon(x^*)$ then x^* is locally stable
- If x^* is stable for all possible initial states x^0, then x^* is globally stable
- Example: growth model $\dot{k} = sf(k) - (1+\delta)k$, where k is wealth per capita and f is a production function
  - questions: when is there an equilibrium and when is it stable?
  - the number of equilibria may also be of interest

Analyzing stability

Theorem

the equilibrium x^* of a system $\dot{x} = F(x)$ is locally asymptotically stable if the real parts of eigenvalues of $DF(x^*)$ are negative

- if there is one eigenvalue with positive real part, then the equilibrium is unstable
- the qualitative behavior of the solution is characterized by the linearized system, note:
  $F(x) \approx F(x^*) + DF(x^*)(x - x^*) = DF(x^*)(x - x^*)$, with the translation of origin $y = x - x^*$ we get a linear system $\dot{y} = DF(x^*)y$
- for $n = 1$ the condition is that $F'(x^*) < 0$

Analyzing stability

- Example: population model for two species

$$\begin{aligned} \dot{x}_1 &= x_1(4 - x_1 - x_2) \\ \dot{x}_2 &= x_2(6 - x_2 - 3x_1) \end{aligned}$$

  - finding the equilibria: (*i*) $(x_1, x_2) = (0, 0)$, (*ii*) $(x_1, x_2) = (0, 6)$, (*iii*) $(x_1, x_2) = (4, 0)$, (*iv*) $(x_1, x_2) = (1, 3)$
  - computation of the Jacobian $\begin{pmatrix} 4 - 2x_1 - x_2 & -x_1 \\ -3x_2 & 6 - 2x_2 - 3x_1 \end{pmatrix}$
  - eigenvalues: (*i*) $\lambda_{1,2} = 4, 6$, (*ii*) $\lambda_{1,2} = -2, -6$, (*iii*) $\lambda_{1,2} = -4, -6$, (*iv*) $\lambda_{1,2} = -2 \pm \sqrt{10}$

Optimization

Introduction, unconstrained optimization
SB: 17, 19.2

Basic concepts

Definition

Point $x^* \in X$ is a maximum of $f : X \mapsto \mathbb{R}$ on $S \subseteq X$ if $f(x^*) \geq f(x)$ for all $x \in S$

- Mathematical programming (or optimization) problem: find a maximum (or a minimum) of a given function $f : X \mapsto \mathbb{R}$ on a set $S \subset X$
  - function f is the objective function
  - S is a feasible set, if $x \in S$, then x is a feasible point (solution)
  - if $x \in S$ solves the maximization problem, it is an optimum
  - note: the main tool for the existence optimum is the Weierstrass theorem

Basic concepts

Definition

Assume that the domain X is a subset of a metric space. Then x^* is a local maximum of the problem if there is $\varepsilon > 0$ such that $f(x^*) \geq f(x)$ for all $x \in N_\varepsilon(x^*) \cap S$

- Local maxima and minima are called (local) extrema
- If $f(x^*) > f(x)$ for all $x \in N_\varepsilon(x^*) \cap S$ and $x \neq x^*$, then x^* is strict local maximum (strict global maximum respectively)

Notations

- The maximization problem is denoted as $\max_{x \in S} f(x)$ and the minimization problem as $\min_{x \in S} f(x)$
- If f is maximized over x while y are fixed (exogenous) we denote: $\max_x f(x, y)$
- Note: if x is a minimum of f, then it is a maximum of $-f(x)$ (wlog we may consider maximization problems)
- If the supremum of f over S is attained at a point x^*, then $f(x^*) = \max_{x \in S} f(x) = \sup_{x \in S} f(x)$

Optimization problems

- When $S = X$, i.e., the feasible set is the whole domain of f, the problem is unconstrained, otherwise the problem is constrained
- The general optimization problem (for $X \subseteq \mathbb{R}^n$) is a *nonlinear programming problem*
- When the problem involves time, it is a dynamic optimization problem
  - when there is no time involved, the problem is static

Optimization problems: NLP

- Nonlinear programming problems where $S = \{x \in \mathbb{R}^n : g_1(x) \leq 0, \ldots, g_k(x) \leq 0$ and $h_1(x) = 0, \ldots, h_m(x) = 0\}$, $g_i : \mathbb{R}^n \mapsto \mathbb{R}$, $h_j : \mathbb{R}^n \mapsto \mathbb{R}$, $i = 1, \ldots, k$, $j = 1, \ldots, m$
  - the inequalities $g_1(x) \leq 0, \ldots, g_k(x) \leq 0$ are inequality constraints (note that $g_i(x) \geq 0 \Leftrightarrow -g_i(x) \leq 0$), and the constraints $h_1(x) = 0, \ldots, h_m(x) = 0$ are equality constraints
  - if f, $g_1, \ldots, g_k$, $h_1, \ldots, h_m$ are linear (or affine) the problem is a linear programming problem

Unconstrained problems: FOC

Theorem

Let $X \subseteq \mathbb{R}^n$ be an open set, and $f : X \mapsto \mathbb{R}$ be a differentiable function on X. If x^* is a local extremum of f, then $\nabla f(x^*) = \mathbf{0}$.

- Note: the converse is not necessarily true, i.e., $\nabla f(x^*) = \mathbf{0}$ is a necessary but not sufficient optimality condition, we call $\nabla f(x^*) = \mathbf{0}$ as a first order optimality condition
- The points at which $\nabla f(x) = \mathbf{0}$ are critical points of f
- For $n = 1$ the FOC is simply $f'(x^*) = 0$

Second order optimality conditions

- f twice differentiable on X
  - 2nd order necessary condition: if x^* is a local maximum, then $D^2 f(x^*)$ is negative semidefinite
  - 2nd order sufficient condition (SOC): if x^* is a critical point and $D^2 f(x^*)$ is neg.def. then x^* is a strict local maximum
  - note: for $n = 1$, $D^2 f(x^*) = f''(x^*)$, i.e., $f''(x^*) \geq 0$ (i.e., $f'(x^*)$ decreasing at x^*)
  - note: for minimization problem replace neg. (sem.) def. with pos. sem. def.

Least squares problems

- Data, observations $(y_1, x_{11}, x_{21}, \ldots, x_{k1}), \ldots, (y_n, x_{1n}, \ldots, x_{kn})$
- Linear model $y_i = \beta_1 x_{1i} + \cdots + \beta_k x_{ki} + \varepsilon_i$
  - independent and identically distributed observation error ε_i, $i = 1, \ldots, n$
  - unknown parameter β (vector)
  - more generally we may have a (nonlinear) model $y_i = f(x^i, \beta) + \varepsilon_i$, where β is a parameter vector
  - the linear model can be written as a system $y = X\beta + \varepsilon$, where $y, \varepsilon \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times k}$ and $\beta \in \mathbb{R}^k$

Least squares problems

- Least squares problem $\min_\beta (y - X\beta)^{\mathrm{T}}(y - X\beta)$
- Solution by FOC and SOC
  - FOC: $\nabla f(\beta) = \nabla_\beta (y - X\beta)^{\mathrm{T}}(y - X\beta) = \mathbf{0}$, now $\nabla f(\beta) = -2X^{\mathrm{T}}y + 2X^{\mathrm{T}}X\beta$, solving for β gives $\beta = (X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}y$
  - SOC: $D^2 f(\beta) = 2X^{\mathrm{T}}X$ which is pos. sem. def. (why?)

Envelope theorem

- Consider a problem $\max_{x \in \mathbb{R}^n} f(x, a)$, where $a \in \mathbb{R}^m$ is exogenous
- How does the optimal f vary as a is changed?
  - what are partial derivatives of $\partial f(x(a), a)/\partial a_i$, $i = 1, \ldots, m$?
  - i.e., we are interested in comparative statics of the value function $v(a) = \max_x f(x, a)$

Envelope theorem

Theorem

Assume that the optimum of f is unique in the neighborhood of a^*, f is differentiable at $(x(a^*), a^*)$, and $x(a)$ is differentiable function at a^*. It follows that $dv(a^*)/da_i = \partial f(x(a^*), a^*)/\partial a_i$ for all $i = 1, \ldots, m$, where v is the value function

- we denote $Dv(a^*) = D_a f(x(a^*), a^*)$ (total derivative of v is the partial derivative of f w.r.t. a)
- when is $x(a)$ a differentiable function?
- it is possible to get qualitative results without actually finding $x(a)$
- partial derivatives are easier to compute than total derivatives

Example

- $f(x, a) = -x^2 + 2ax + 4a^2$
- Optimum by setting gradient to zero: $x(a) = a$
- Plug the solution into the objective function are differentiate w.r.t a
  - $g(a) = f(x(a), a) = 5a^2$, hence $dg(a)/da = df(x(a), a)/da = 10a$
- Same result by using the envelope theorem $\partial f(x(a), a)/\partial a = 2x + 8a = 10a$

Example: marginal costs

- Assume that a firm minimizes costs of labor (L) and capital (K) when producing a given amount q
  - production function $f(K, L)$, i.e., output $q = f(K, L)$
  - unit costs of capital and labor w_L and w_K
  - problem $\min_{L,K} w_L L + w_K K$ s.t. $q = f(K, L)$
  - eliminate labor from $q = f(K, L)$ to obtain (short run optimal) cost function $c(K, q)$
- In the long run capital is chosen such that the costs are minimized $\Rightarrow K(q)$, in the short run capital is fixed
  - $c(K, q)$ is the short run optimal cost
- Long run and short run marginal costs are the same
  - by the envelope theorem $dc(K(q), q)/dq = \partial c(K(q), q)/\partial q$

Example: marginal costs

- Test with production function $\sqrt{KL}$, where L is labor
  - set unit costs of labor and capital $= 1$
  - cost of input K (capital) and L (labor) is $K + L$, production function $\sqrt{KL}$, i.e., output $q = \sqrt{KL}$, from which we get $L = q^2/K$, and $c(K, q) = q^2/K + K$
  - long run optimum $K(q) = q$, at which $c(K(q), q) = 2q$ and $dc(K(q), q)/dq = 2$
- Observations:
  - the graph of a function $c(K, q)$ is above $c(K(q), q)$ except at $K = q$ (optimum)
  - at $K = q$ the tangent of the functions are the same

Convex analysis

Convex sets, concave and quasiconcave functions
SB: 21, 22.1

Convex sets

Definition

Let V be a vector space. The set $X \subset V$ is convex if $\lambda x + (1-\lambda)y \in X \ \forall x, y \in X$ and $\lambda \in [0,1]$

- vector $\lambda x + (1-\lambda)y$ is a convex combination of x and y
- X is convex if and only if $y = \sum_{i=1}^{k} \lambda_i x^i \in X$ for all $k \geq 2$, $x^i \in X$, $i = 1, \ldots, k$, $\sum_i \lambda_i = 1$ and $\lambda_i \in [0,1]$, $i = 1, \ldots, k$, y is called a convex combination of $x^1, \ldots, x^k$
- a set X is strictly convex if X is convex and $\lambda x + (1-\lambda)y \notin \partial X$ for any $x, y \in X$, $x \neq y$, $\lambda \in (0,1)$

Convexity

- Operations that preserve convexity
  - Closure and interior of a convex set are convex
  - Intersection of convex sets is convex
  - Linear map of a convex set is convex
  - Cartesian product of convex sets is convex

Theorem

(Minimum distance theorem) Let $S \subset \mathbb{R}^n$ be a closed convex set and $y \notin S$. Then there is a unique point $\bar{x} \in S$ with minimum distance from y. Necessary and sufficient condition for $\bar{x}$ to be the minimizer is $(y - \bar{x}) \cdot (x - \bar{x}) \leq 0$ for all $x \in S$

Convexity

- Examples
  - hyperplanes $p \cdot x = c$, (open and closed) half spaces $p \cdot x \leq c$, (also $p \cdot x < c$, $p \cdot x > c$), polyhedral sets = intersections of half spaces, ellipsoids $x \cdot Vx \leq c$ (V pos.def.), solutions of linear equations, simplex
  - Note: sets $\{x\}$, $\emptyset$, and $\mathbb{R}^n$ are convex
- Every closed convex set can be presented as an intersection of closed half spaces

Convex and concave functions

Definition

Let V be a vector space and $X \subset V$ a convex subset of V, function $f : X \mapsto \mathbb{R}$ is concave if

$$f(\lambda x + (1-\lambda)y) \geq \lambda f(x) + (1-\lambda) f(y) \ \forall x, y \in X, \ \lambda \in [0,1]$$

- Function f is strictly concave if

$$f(\lambda x + (1-\lambda)y) > \lambda f(x) + (1-\lambda) f(y) \ \forall x, y \in X, \ x \neq y, \ \lambda \in (0,1)$$

- If $-f$ is (strictly) concave then f is (strictly) convex

Concave functions

- Hypograph of a function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is hyp$f = \{(x, \mu) \in \mathbb{R}^{n+1} : f(x) \geq \mu\}$
  - hyp$f$ is the set of point below the graph of a function, respectively the set above the graph is the epigraph (epi$f$)
  - f is concave (convex) function if and only if hyp$f$ (epi$f$) is a convex set
- Examples of concave functions
  $\ln(x)$, $-e^x$, x^α when $\alpha \in [0,1]$, $-|x|^p$ when $p \geq 1$, $\min\{x_1, \ldots, x_n\}$, $x \cdot Vx$ when V is neg.sem.def

Properties of concave functions

- Upper level sets $U(f;\alpha) = \{x \in X : f(x) \geq \alpha\}$ are convex sets
- Concave function is locally Lipschitz continuous on the interior of its domain
- Extra: concave function is almost everywhere differentiable on the interior of its domain
- If $V = \mathbb{R}^n$ and a function is bounded, convexity is equivalent to $f(\lambda x + (1-\lambda)y) \geq \lambda f(x) + (1-\lambda) f(y)$ for some $\lambda \in (0,1)$
- f is concave if and only if $g_{x,y}(\lambda) = f(\lambda x + (1-\lambda)y)$ is concave for all x and y on $[0,1]$

Operations that preserve concavity

- Positively weighted sum of concave functions $\sum w_i f_i(x)$, $w_i \geq 0$, is concave
- Minimum $g(x) = \min_{\alpha \in A} f(x, \alpha)$ is concave when $f(x, \alpha)$ is concave for all $\alpha \in A$
- Composite function $f(x) = h(g(x))$, $g = (g_1, \ldots, g_m) : \mathbb{R}^n \mapsto \mathbb{R}^m$, $h : \mathbb{R}^m \mapsto \mathbb{R}$, f s concave in the following cases

1. when h is concave, non-decreasing and g_i is concave for all i
2. when h is concave, non-increasing and g_i is convex for all i
3. when h is concave and g is linear/affine ($g(x) = Ax + b$)

Examples

- Model of a firm
  - input vector $x \in \mathbb{R}^n$
  - production function $f : \mathbb{R}^n_+ \mapsto \mathbb{R}_+ = \{y \in \mathbb{R} : y \geq 0\}$ that is concave and in increasing in all its arguments
  - cost function $c : \mathbb{R}^n_+ \mapsto \mathbb{R}$, convex and increasing in all its arguments
  - price $p > 0$, profits $pf(x) - c(x)$ defined for $x \geq \mathbf{0}$
- Expenditure function $e(p) = \min_{x \in X} p \cdot x$ is concave
  - $p \in \mathbb{R}^n$ is a vector of prices of inputs $x \in X \subset \mathbb{R}^n_+$, and X convex

Differentiable concave functions

- Graph of a concave function is below tangent plane:

$$f(y) - f(x) \leq \nabla f(x) \cdot (y - x), \ \forall x, y \in X$$

Theorem

Differentiable function is (strictly) concave if and only if its gradient is (strictly) monotone

- for $f : \mathbb{R} \mapsto \mathbb{R}$, $f'(x)$ is decreasing
- for $f : \mathbb{R}^n \mapsto \mathbb{R}$ we have $[\nabla f(x) - \nabla f(y)] \cdot (x - y) \leq 0$ for all $x, y \in \mathbb{R}^n$
- strict monotonicity: strict inequality for $x \neq y$

Differentiable concave functions

Theorem

Twice differentiable function f is concave if and only if its Hessian matrix $D^2 f(x)$ is negative semidefinite (on the domain of f)

- note 1: this is the main tool for analyzing concavity
- note 2: negative semidefiniteness implies monotonicity of ∇f
- note 3: positive (semi)definiteness $\Leftrightarrow$ convexity
- note 4: for $f : \mathbb{R} \mapsto \mathbb{R}$ concavity $\Leftrightarrow f''(x)$ non-positive for all x
- Hessian matrix of a strictly concave function is not necessarily neg. nef. but if it is, then the function is strictly concave

Concave optimization

- An optimization problem $\max_{x \in S} f(x)$ is concave (or convex) if f is a concave function and S is a convex set
  - when $S = \{x \in \mathbb{R}^n : g(x) \leq \mathbf{0}, \ h(x) = \mathbf{0}\}$, and $g = (g_1, \ldots, g_k)$, $h(x) = Ax + b$, then S is a convex set when $g_i(x)$, $i = 1, \ldots, k$, are convex functions

Theorem

Local maximum of a concave function is a global maximum

- first order conditions give an optimum

Concave optimization

Theorem

The set of solutions of a concave problem is a convex set

- The set of maximizers over S is denoted as $\arg\max_{x \in S} f(x)$
- If f is strictly concave and the problem has a solution then it is unique
- Note: concavity does not guarantee existence

Theorem

When f is concave and differentiable, then x^* is an optimum if and only if $\nabla f(x^*) \cdot (x - x^*) \leq 0$ for all $x \in S$

- Variational inequality of concave optimization
- Necessary and sufficient optimality condition for a concave problem

Parametric optimization and concavity

- Function $v(a) = \max_{x \in X(a)} f(x, a)$ (value function) is concave when f is concave in (x, a) and $X(a)$ is graph convex correspondence
- Graph convexity: $\lambda X(a^1) + (1 - \lambda) X(a^2) \subseteq X(\lambda a^1 + (1 - \lambda) a^2)$
- e.g., $X(a) = \{x \in \mathbb{R}^n : g_i(x, a) \leq 0, \ i = 1, \ldots, k\}$ is graph convex when g_i are convex functions in (x, a)

Quasiconcave functions

Definition

Let X be a convex subset of a vector space V, function $f : X \mapsto \mathbb{R}$ is quasiconcave if $f(\lambda x + (1 - \lambda) y) \geq \min\{f(x), f(y)\}$ $\forall x, y \in X$ and $\lambda \in [0, 1]$

- Strict quasiconcavity: $f(\lambda x + (1 - \lambda) y) > \min\{f(x), f(y)\}$, $x \neq y$, $\lambda \in (0, 1)$
- (Strict) quasiconcavity is equivalent with $U(f, \alpha)$ (upper level set) being a (strictly) convex set for all $\alpha \in \mathbb{R}$
- If $-f$ is (strictly) quasiconcave, then f is (strictly) quasiconvex

Quasiconcave functions

- Properties of a quasiconcave function $f : \mathbb{R}^n \mapsto \mathbb{R}$
  1. a differentiable function f on an open set convex set X is quasiconcave if and only if $f(y) \geq f(x) \Rightarrow \nabla f(x) \cdot (y - x) \geq 0$ for all $x, y \in X$
  2. if $y \cdot \nabla f(x) = 0 \Rightarrow y \cdot D^2 f(x) y \leq 0$ for all $x, y \in X$, then f is quasiconcave (for strictly quasiconcave functions we need "< 0")

Quasiconcavity and transformations

- Monotone transformation of a quasiconcave function is quasiconcave
  - i.e., $f(x) = h(g(x))$, where $g : \mathbb{R}^n \mapsto \mathbb{R}$ is quasiconcave and $h : \mathbb{R} \mapsto \mathbb{R}$ is (strictly) increasing
  - $\Rightarrow$ if a function is a monotone transformation of a concave function then it is quasiconcave
- If a monotone transformation of a function is concave then the original function is quasiconcave
  - if $h(f(x))$ is concave with increasing h, then f is quasiconcave
  - important specific case: $\log$-concave functions, $h = \ln$

Quasiconcavity and transformations

- Other operations that preserve quasiconcavity
  - minimization: $g(x) = \min_y f(x, y)$ (f quasiconcave as a function of x)
  - composite function $f(x) = h(Ax + b)$ is quasiconcave, when h is quasiconcave (here $x \in \mathbb{R}^n$ and $Ax + b \in \mathbb{R}^m$)
- Examples
  - Cobb-Douglas utility functions $U(x) = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$, when $x \geq \mathbf{0}$ and $\alpha_i > 0$, $i = 1, \ldots, n$
  - C-D functions are $\log$-concave, they are concave when $\sum_i \alpha_i < 1$ and $\alpha_i \in (0, 1)$ for all i
  - CES utility functions $U(x) = [\sum_i (\alpha_i x_i^r)]^{1/r}$, when $x \geq \mathbf{0}$, $\alpha_i > 0$, $i = 1, \ldots, n$, $r \leq 1$

Utility functions

- Preference relations are often described by utility functions
  - decision maker prefers a to b if $u(a) \geq u(b)$
  - in the ordinal theory of utility the measurement unit of utility is not essential; increasing transformation of u describes the same preference relation
- ▲ A property of a function that can (cannot) be preserved by an increasing transformation is an ordinal (cardinal) property
  - quasiconcavity is an ordinal while concavity is a cardinal property
  - diminishing marginal rate of substitution, i.e., $MRS_{xy} = [\partial u(x, y)/\partial x]/[\partial u(x, y)/\partial y]$ (marginal rate of substitution between x and y) decreasing, is an ordinal property
  - diminishing marginal utility, i.e., $MU_x = \partial u(x, y)/\partial x$ decreasing, is a cardinal property

Quasiconcave optimization

- $\max_{x \in S} f(x)$ is a quasiconcave optimization problem when f is a quasiconcave function and S is a convex set
- Quasiconcave optimization problem may have local optima which are not global optima
  - $\arg\max_{x \in S} f(x)$ is a convex set
  - if f is strictly quasiconcave and the problem has a global optimum, then it is unique

Theorem

If $\nabla f(x) \cdot (y - x) < 0$ for all $y \in S$, $y \neq x$, then $x \in S$ is a global maximum of a quasiconcave optimization problem $\max_{x \in S} f(x)$

- $v(a) = \max_{x \in S} f(x, a)$ is quasiconcave when f is quasiconcave in (x, a) and S is a convex set

Example: consumer's problem

- Variables
  - n commodities, x_i amount of commodity i, price p_i, income I
- Objective function = utility function $u(x)$
  - u is usually assumed to be quasiconcave
- Constraints
  - budget constraint $p_1 x_1 + p_2 x_2 + \ldots + p_n x_n \leq I$ and non-negativity $x_i \geq 0$, $i = 1, \ldots, n$
- Consumer's problem: for given p and I find the utility maximizing commodity bundle
  - the solution is the consumer's demand, if it is unique then it is a demand function
- Questions: existence of demand functions, continuity, etc

Nonlinear programming

constrained optimization
SB: 18

NLP problems

- NLP problem $\max f(x)$ s.t. (subject to) $g_i(x) \leq 0$ $i = 1 \ldots, k$, $h_j(x) = 0$, $j = 1, \ldots, m$
  - we also denote $g(x) \leq \mathbf{0}$ (inequality constraint in vector form) and $h(x) = \mathbf{0}$ (equality constraints)
  - if x satisfies the constraints it is feasible
  - if $g_i(x) = 0$, then ith inequality constraint is active (binding) at x
  - when we denote $\max_x f(x, a)$ we refer to a problem in which a is exogenous

Inequality constraints into equality constraints

- Adding *slack variables*
  - way to transform an inequality constraint to an equality constraint, e.g., by adding variable s we can write a problem $\max f(x)$ s.t. $g(x) \leq 0$ to equality constrained problem $\max_{(x,s)} f(x)$ s.t. $g(x) + s^2 = 0$
  - with this trick it is actually possible to derive most of the results for inequality constrained problems from the theory of equality constrained problems, otherwise the trick is seldom of any use
- Note: replacing equality constraint with two inequality constraints does not work (the problem is that the Jacobians of the constraints become linearly dependent)
- In many inequality constrained problems, it is possible to detect active constraints from the structure of the problem

Nonlinear programming

- Example: consumer's problem
  - $\max_x u(x)$ s.t. $p_1 x_1 + \ldots + p_n x_n \leq I$ and $x_i \geq 0$, $i = 1, , \ldots, n$
  - when u is increasing in all of its arguments, then budget constraint is active at optimum (when prices are positive). Moreover, often $u(x) = 0$ when $x_i = 0$ for some i and $U(x) > 0$ when $x \gg \mathbf{0}$, which means that at optimum $x \gg \mathbf{0}$
- Feasible points of inequality constrained problems
  - if $g_i(x) = 0$ for some $i = 1, \ldots, k$, $k > 1$, we say that x is a corner point, otherwise x is an interior point
  - respectively, when x is an optimum, we say that it is a corner solution or an interior solution

First order conditions: assumptions

- Lagrange function $L(x, \lambda, \mu) = f(x) - \sum_i \lambda_i g_i(x) - \sum_j \mu_j h_j(x)$
  - $\lambda \in \mathbb{R}^k$ is a vector of Lagrange multipliers for inequality constraints and $\mu \in \mathbb{R}^m$ contains the Lagrange multipliers of equality constraints
- Assumptions: x^* is a local optimum, f, g, h are differentiable at x^*, k_0 first inequality constraints are active at x^*, the following matrix has full rank

$$\begin{pmatrix} \frac{\partial g_1(x^*)}{\partial x_1} & \ldots & \frac{\partial g_1(x^*)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_{k_0}(x^*)}{\partial x_1} & \ldots & \frac{\partial g_{k_0}(x^*)}{\partial x_n} \\ \frac{\partial h_1(x^*)}{\partial x_1} & \ldots & \frac{\partial h_1(x^*)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_m(x^*)}{\partial x_1} & \ldots & \frac{\partial h_m(x^*)}{\partial x_n} \end{pmatrix}$$

First order conditions

- Result: there are Lagrange multipliers such that

$$\begin{aligned} \frac{\partial L(x^*, \lambda, \mu)}{\partial x_i} &= 0 \; \forall i = 1, \ldots, n, && (1) \\ \lambda_i g_i(x^*) &= 0 \; \forall i = 1, \ldots, k, && (2) \\ g_i(x^*) &\leq 0 \; \forall i = 1, \ldots, k, && (3) \\ h_i(x^*) &= 0 \; \forall i = 1, \ldots, m, && (4) \\ \lambda_i &\geq 0 \; \forall i = 1, \ldots, k && (5) \end{aligned}$$

- In vector notation:

$$\begin{aligned} \nabla L(x^*, \lambda, \mu) &= \mathbf{0}, \\ \lambda \cdot g(x^*) &= 0, \\ g(x^*) &\leq \mathbf{0}, \\ h(x^*) &= \mathbf{0}, \\ \lambda &\geq \mathbf{0} \end{aligned}$$

Remarks on FOCs

- Necessary optimality condition
  - known as Karush-Kuhn-Tucker conditions
  - for minimization problems the condition is otherwise the same but the sign of Lagrange multiplier of inequality constraints ("$\leq$"-constraints) is changed
  - not all points satisfying FOCs are (local) optima
- Works also when there are no constraints at all, or either only equality or inequality constraints

FOCs: nomenclature

- The rank condition is known as non-degeneracy constraint qualification (NDCQ)
  - assuming k_0 first inequality constraints active was done to simplify notation, more generally the matrix can be formed by collecting the active constraints

(1) Lagrangian optimality (x^* is a critical point of L)
(2) complementary slackness
(3)–(4) primal feasibility
(5) dual feasibility

Geometric interpretation

- Problem $\max f(x)$ s.t. $g(x) \leq 0$
- At local maximum, x^*, there is no feasible ascent direction
  - $d \neq \mathbf{0}$ is a feasible direction at x^* if there is $\bar{\lambda}$ such that $g(x^* + \lambda d) \leq 0$ for $\lambda \in (0, \bar{\lambda})$
  - $d \neq \mathbf{0}$ is an ascent direction at x^* if there is $\bar{\lambda}$ such that $f(x^* + \lambda d) > f(x^*)$ for all $\lambda \in (0, \bar{\lambda})$

  $\Rightarrow$ at local maximum there is no $d \neq \mathbf{0}$ such that $\nabla g(x^*) \cdot d < 0$ and $\nabla f(x^*) \cdot d > 0$
  - when $\nabla g(x^*) \neq \mathbf{0}$ this is equivalent to FOCs

Cookbook: solving FOCs

- Method 1 (brute force): find all the points that satisfy the FOCs
  - for equality constrained problems simply find all solutions of the system of equations determined by FOCs
  - when there are inequality constraints, go through all possible combinations of active constraints, if some combination is not possible it will imply wrong sign for one of the Lagrange multipliers
- Method 2: make an educated guess of active constraints (or even guess the solution)
  - check your guess by plugging it into FOCs

Example 1

- Problem $\max x_1 - x_2^2$ s.t. $x_1^2 + x_2^2 = 4$, $x_1, x_2 \geq 0$
- Lagrangian $L(x, \lambda, \mu) = x_1 - x_2^2 + \lambda_1 x_1 + \lambda_2 x_2 - \mu(x_1^2 + x_2^2 - 4)$
- FOCs:

$$\begin{aligned}
\partial L/\partial x_1 = 1 + \lambda_1 - 2\mu x_1 &= 0 \\
\partial L/\partial x_2 = -2x_2 + \lambda_2 - 2\mu x_2 &= 0 \\
x_1^2 + x_2^2 &= 4 \\
x_i &\geq 0, \ i = 1, 2 \\
\lambda_i x_i &= 0, \ i = 1, 2 \\
\lambda_i &\geq 0, \ i = 1, 2
\end{aligned}$$

Example 1

- Deducing the active constraints
  - both inequality constraints cannot be active at the same time
  - if $\lambda_1 = \lambda_2 = 0$ (e.g., none of the constraints is active) then from $\partial L/\partial x_2 = 0$ we get $\mu = -1$ ($\Rightarrow x_1 = -1/2$, infeasible!) or $x_2 = 0$ ($\Rightarrow x_1 = 2, \ \mu = 1/4$)
  - Are there other solutions than $(x_1, x_2, \lambda_1, \lambda_2, \mu) = (2, 0, 0, 0, 1/4)$? If there were then $x_2 > 0$ and $\lambda_2 = 0$, by the constraints $\mu = -1$ which leads to infeasible x_1, hence there are no other solutions

Example 2

- $f(x) = -(x_1 - 1)^2 - (x_2 - 1)^2$ and $h(x) = x_1^2 + x_2^2 - 8$
- Geometric interpretation: minimize the distance from $(1, 1)$ to circle $h(x) = 0$
- By geometric intuition the solution seems to be $(2, 2)$
- FOCs: equality constraint and $-2(x_1 - 1) + 2\mu x_1 = 0$, $-2(x_2 - 1) - 2\mu x_2 = 0$, by choosing $\mu = 1/2$ the FOCs are satisfied $(2, 2)$

Example: Cobb-Douglas consumer

- Two commodities, Cobb-Douglas utility function $U(x_1, x_2) = x_1^a x_2^{1-a}$ and budget constraint $h(x) = 0$, where $h(x_1, x_2) = p_1 x_1 + p_2 x_2 - I$
- $L(x, \mu) = x_1^a x_2^{1-a} - \mu(p_1 x_1 + p_2 x_2 - I)$
  - NDCQ holds for $p \neq \mathbf{0}$
- FOC: budget constraint and $a x_1^{a-1} x_2^{1-a} - \mu p_1 = 0$, $(1 - a) x_1^a x_2^{-a} - \mu p_2 = 0$
- By eliminating μ we get $(1 - a)(x_1/x_2)^a - a(x_1/x_2)^{a-1}(p_2/p_1) = 0$, i.e., $p_1(1 - a) - p_2 a (x_1/x_2)^{-1} = 0$ and eventually $p_1 x_1 (1 - a) - p_2 x_2 a = 0$
  - note: we assume $x_1, x_2 > 0$ at optimum

Example: Cobb-Douglas consumer

- Solving the pair of equations: budget equation and the above condition
  - from budget equation we get $p_1 x_1 = I - p_2 x_2$, and by plugging this into the other we obtain $x_2 = (1 - a)I/p_2$ and $x_1 = aI/p_1$
- What can you say about the optimality of the solution?

More general FOCs

- FOCs can be formulated for problems where the optimized variables are infinite dimensional
  - formally this works only when the underlying space of optimized variables has particular structure
- Example: consumer's life cycle consumption problem $\max \sum_{k=0}^{\infty} \delta^k u(c_k)$, $\delta \in (0, 1)$
  - period k consumption c_k
  - initial wealth (exogenous) w_0, interest rate r, dynamics $w_{k+1} = (1 + r)(w_k - c_k)$, where w_k is period k wealth
  - difference equation for wealth is considered as a constraint

Example (cont'd)

- FOCs
  - Lagrangian

$$L(c, w, \mu) = \sum_{k \geq 0} \left[\delta^k u(c_k) - \mu_k \left(w_{k+1} - (1+r)(w_k - c_k) \right) \right]$$

  - note: optimized variables are $c_k,\ w_{k+1},\ k \geq 0$, i.e., there are infinitely many variables to be optimized
  - differentiate L w.r.t all variables (c_k's and w_k's) and set the differential to zero

$$\begin{aligned} \delta^k u'(c_k) + \mu_k (1+r)(-1) &= 0 \text{ (diff. } L \text{ w.r.t } c_k) \\ (1+r)\mu_k - \mu_{k-1} &= 0 \text{ (diff. } L \text{ w.r.t } w_k,\ k \geq 1) \\ w_{k+1} &= (1+r)(w_k - c_k),\ k \geq 0 \end{aligned}$$

  - FOCs are a difference equation for $c_k,\ a_k$ and $\mu_k,\ k = 0, 1, \ldots$
  - note: we can eliminate μ_k's

Comparative statics

Constrained optimization,
interpretation of Lagrange multipliers,
generalized envelope theorem
SB: 19.1, 19.2

Introduction

- How does the solution of an optimization problem change when exogenous variables change?
  - example: how does consumer's optimum change when the income is increased?
- Optimization problem $\max f(x, a)$ s.t. $h(x, a) = \mathbf{0}$, where a is exogenous
  - the solution x (or $\arg\max$) will depend on exogenous variables, i.e., it is a correspondence (or function) $x(a)$
  - how does $x(a)$ change as a function of a, what about $v(a) = f(x(a), a)$?

Interpretation of Lagrange multipliers

Theorem

Assume that $x(a)$ solves $\max f(x)$ s.t. $h(x) = a$, where $h : \mathbb{R}^n \mapsto \mathbb{R}^m$, and $x(a)$ and $\mu(a)$ (Lagrange multipliers corresponding to $x(a)$) are differentiable functions on an open set of exogenous variables a. Assume also that the first order conditions with NDCQ are satisfied at $x(a)$, $\mu(a)$. Then $\nabla v(a) = \mu(a)$, i.e., $df(x(a))/da_i = \mu_i(a)$.

- Note: the result tells the change of optimal f for small changes of exogenous variables
- Lagrange multipliers can be interpreted as shadow prices!
- Why these assumptions? When can we be certain that $x(a)$ is differentiable?

Interpretation of Lagrange multipliers

- When we think the right hand side of $h(x) = a$ as an amount of resource, then $df(x(a))/da_i$ tells how the optimum changes as the amount of resource i changes
- Sketch for a proof:
  - let v denote the value function, by chain rule $\nabla_a v(a) = [D_a x(a)]^T \nabla_x f(x(a))$
  - Lagrangian optimality $\nabla_x f(x(a)) - [D_x h(x(a))]^T \mu(a) = \mathbf{0}$, i.e., $\nabla_x f(x(a)) = [D_x h(x(a))]^T \mu(a)$
  - by differentiating $h(x) = a$, we get $D_a[h(x(a))] = I \Leftrightarrow D_x h(x(a)) D_a x(a) = I$ (by chain rule)
  - combining the results $\nabla_a v(a) = [D_a x(a)]^T [D_x h(x(a))]^T \mu(a) = I^T \mu(a) = \mu(a)$

Interpretation of Lagrange multipliers

Theorem

$\max_x f(x)$ s.t. $g(x) \leq a$ and assumptions otherwise as before and a has an open neighborhood such that the active constraints remain the same on that neighborhood. Then $df(x(a))/da_i = \lambda_i(a)$, where $\lambda_i(a)$ is the Lagrange multiplier corresponding to ith constraint.

- example: $\max x_1 + x_2$ s.t. $x_1^2 + x_2^2 \leq a$ and $x_2 \leq 1$, $a > 0$. What happens at $a = 2$?

Generalized envelope theorem

Theorem

$\max_x f(x, a)$ s.t. $h(x, a) = \mathbf{0}$, $a \in \mathbb{R}$. Assume that that the solution of the problem $x(a)$, $\mu(a)$ are differentiable (on an open neighborhood of a) and the FOCs are satisfied and NDCQ holds at optimum. It follows that $df(x(a), a)/da = \partial L(x(a), \mu(a), a)/\partial a$

- Combination of envelope theorem and the result on the interpretation of Lagrange multipliers

Example 1

- $f(x_1, x_2) = x_1^2 x_2$ and $h(x_1, x_2) = 2x_1^2 + x_2^2 - a$
- The problem can be solved by eliminating the other variable from the constraint $h(x) = 0$, we get an unconstrained problem
  - this trick works more generally, but usually elimination of variables is difficult, note also that if there are inequality constraints they have to be taken into account after the elimination
  - by eliminating x_1^2, i.e., $x_1^2 = (a - x_2^2)/2$ and plugging this into the objective function we get $\max x_2(a - x_2^2)/2$
  - differentiating and setting the derivative to zero we get $a - 3x_2^2 = 0$, i.e., $x_2 = \pm\sqrt{a/3}$ and $x_1 = \pm\sqrt{a/3}$, the optimum is $x = (\sqrt{a/3}, \sqrt{a/3})$

Example 1

- The drawback of elimination approach is that it does not produce Lagrange multiplier for the constraint $h(x) = 0$, to get the Lagrange multiplier we have to formulate FOCs $\partial f(x)/\partial x_1 = 2x_1 x_2$ and $\partial h(x)/\partial x_1 = 4x_1$, hence, $\mu(a) = x_2(a)/2 = \sqrt{a/12}$
- Try e.g., $a_1 = 3$ and $a_2 = 3.3$, according to theory $[f(x(a_2)) - f(x(a_1))]/[a_2 - a_1] \approx \mu(a_1)$, now $f(x(a_2)) - f(x(a_1)) \approx 0.1537$, $\mu(a_1) = 0.5$, which gives $\mu(a_1)(a_2 - a_1) = 0.3 \cdot 0.5 = 0.15$, the difference is in the third decimal

Example 2

- Output is produced of two raw materials x_1, x_2 (amounts)
  - production function $x_1^\alpha x_2^\beta$, unit costs c_1 and c_2, price p
  - objective function (profit) $px_1^\alpha x_2^\beta - c_1x_1 - c_2x_2$
- Regulator sets a constraint $x_1 = x_2$
- Assume that the firm wants to add the usage of input 1
  - this assumption imposes some restrictions on exogenous variables (see below)
  - the firm would rather face a constraint $x_1 = x_2 + a$, $a > 0$ than the original constraint

Example 2

- We estimate how much the firm should invest in lobbying (or bribing) for the change of the constraint to the form $x_1 = x_2 + a$
- We solve the problem without making the tempting elimination $x_1 = x_2$
  - we do not eliminate variables at this stage because we want the Lagrange multiplier of the constraint
- FOCs:

$$\begin{aligned} \alpha p x_1^{\alpha-1} x_2^\beta - c_1 - \mu &= 0 \\ \beta p x_1^\alpha x_2^{\beta-1} - c_2 + \mu &= 0 \\ x_1 &= x_2 \end{aligned}$$

Example 2

- Solving the FOCs by plugging $x_2 = x_1$ into two other conditions and eliminating μ
  - we get $\alpha p x_1^{\alpha+\beta-1} + \beta p x_1^{\alpha+\beta-1} - c_1 - c_2 = 0$
  - the solution is $x_1 = x_2 = [(c_1 + c_2)/(p(\alpha+\beta))]^{1/(\alpha+\beta-1)}$
  - Lagrange multiplier is $\mu = (\alpha c_2 - \beta c_1)/(\alpha+\beta)$
- When exogenous variables satisfy $\alpha c_2 > \beta c_1$, the firm is willing to add input 1
  - for a small a the firm should invest at most $a\mu$ for lobbying, μ is the price of the constraint

Differentiability of solutions

- When is $x(a)$ differentiable? And what is its derivative?
- Unconstrained problem
  - FOC: $\partial f(x,a)/\partial x_i = 0$, $i = 1, \dots, n$, denote this set of equations as $F(x,a) = \mathbf{0}$
  - by implicit function theorem, the solution $x(a)$ is differentiable if the matrix $D_xF = D_x^2 f$ is non-singular
  - the assumptions of IFT are satisfied when $f \in C^2(N_\varepsilon(x(a)))$ and $D_x^2 f(x,a)$ is non-singular on $N_\varepsilon(x(a))$

Differentiability of solutions

- Equality constrained problem
  - FOCs are a system of equations that determines $x(a)$ and $\mu(a)$
  - FOCs can be written as

$$\begin{aligned} \nabla_x L(x,\mu,a) &= \mathbf{0} \\ h(x,a) &= 0 \end{aligned}$$

  - IFT requires that the Jacobian of this system w.r.t (x,μ) is non-singular
  - note: non-singularity also guarantees NDCQ
  - denote $F(x,\mu,a) = \mathbf{0}$, IFT:

$$D(x(a),\mu(a)) = -[D_{(x,\mu)}F(x,\mu,a)]^{-1} D_a F(x,\mu,a)$$

Example: consumer's problem

- $\max_x u(x)$ s.t. $p \cdot x \le I$, $x \ge \mathbf{0}$, $x, p \in \mathbb{R}^n$, $p \gg \mathbf{0}$ and I are exogenous
  - we assume that at optimum budget constraint is active and $x \gg \mathbf{0}$
- FOC:

$$\begin{aligned} \nabla u(x) - \mu p &= \mathbf{0} \\ I - p \cdot x &= 0 \end{aligned}$$

- The solution (when unique) is Marshallian demand function $x(p, I)$

Example: consumer's problem

- Differentiability of the demand, the below matrix should be non-singular

$$H(x,p) = \begin{pmatrix} D^2u(x) & -p \\ -p^T & 0 \end{pmatrix}$$

  - H is the Jacobian of FOCs w.r.t. (x,μ)
  - H is non-singular when $D^2u(x)$ is neg. def. (and $\mu > 0$), in that case also H is neg. def.
- Example $u(x_1,x_2) = \ln x_1 + \ln x_2$, FOCs:

$$\begin{aligned} 1/x_1 - \mu p_1 &= 0 \\ 1/x_2 - \mu p_2 &= 0 \\ I - p_1x_1 - p_2x_2 &= 0 \end{aligned}$$

and $H(x,p) = \begin{pmatrix} -1/x_1^2 & 0 & -p_1 \\ 0 & -1/x_2^2 & -p_2 \\ -p_1 & -p_2 & 0 \end{pmatrix}$

Example: consumer's problem

- Using IFT we get the Jacobian

$$D_{(p,I)}\left(x(p,I),\mu(p,I)\right) = -[H(x,p)]^{-1} D_{(p,I)}F(x,\mu,p,I),$$

where

$$D_{(p,I)}F(x,\mu,p,I) = \begin{pmatrix} -\mu & 0 & 0 \\ 0 & -\mu & 0 \\ -x_1 & -x_2 & 1 \end{pmatrix}$$

- e.g., for $p = (2,2)$ and $I = 1$, we get $x_1 = x_2 = 1/4$ and $\mu = 2$,

$$D_{(p,I)}\left(x(p,I),\mu(p,I)\right) = -\begin{pmatrix} -16 & 0 & -2 \\ 0 & -16 & -2 \\ -2 & -2 & 0 \end{pmatrix}^{-1} \begin{pmatrix} -2 & 0 & 0 \\ 0 & -2 & 0 \\ -1/4 & -1/4 & 1 \end{pmatrix}$$

we get $D_{(p,I)}\left(x(p,I),\mu(p,I)\right) = \begin{pmatrix} -1/8 & 0 & 1/4 \\ 0 & -1/8 & 1/4 \\ 0 & 0 & -2 \end{pmatrix}$

Sufficient optimality conditions

SB: 19.3, 21.5

Sufficient conditions for global optimum

- NLP problem $\max f(x)$ s.t. $g(x) \leq \mathbf{0}$ and $h(x) = \mathbf{0}$

Theorem

Solution of the FOCs is a global optimum if f is concave and g_i, $i = 1, \ldots, k$ ($g : \mathbb{R}^n \mapsto \mathbb{R}^k$) are quasiconvex and $h(x) = Ax + b$

- when f is strictly concave the solution is unique
- concavity of f can be replaced by the assumption that f is quasiconcave and $\nabla f(x) \neq \mathbf{0}$ for all feasible x

Second order conditions

Theorem

Let f, g, h be twice continuously differentiable around x^* which satisfies the FOCs is a local maximum, if $D_x^2 L(x^*, \lambda, \mu)$ is negatively definite on the set

$$\begin{aligned} C = \{ & v \neq \mathbf{0} : \nabla g_i(x^*) \cdot v = 0, i = 1, \ldots k_0, \\ & \nabla g_i(x^*) \cdot v \leq 0, \ i = k_0 + 1, \ldots, k_1, \\ & Dh(x) v = \mathbf{0} \}, \end{aligned}$$

where k_1 first inequality constraints are active and first k_0 of these have strictly positive Lagrange multipliers

- neg. def. mathematically: $v^T D_x^2 L(x, \lambda, \mu) v < 0 \ \forall v \in C$
- note: Lagrange multiplier vectors λ and μ corresponding to x^*, i.e., for the variables FOCs hold
- the neg.definiteness requirement is known as the second order sufficient condition

Second order conditions

- Remarks
  - when all Lagrange multipliers corresponding to active inequality constraints are positive, then C is a tangent set defined by the active inequality constraints and the equality constraints
  - the 2nd order condition is not satisfied when $C = \emptyset$
  - a point can be a maximum even though neg. def. requirement is not satisfied, however we have ...

Theorem

If x^* is a local maximum, then $D_x^2 L$ is neg.sem.def $\forall v \in C$ (when $C \neq \emptyset$)

- negative**semi**definiteness does not imply optimality

Second order conditions

- Variations of second order conditions
  - if $D_x^2 L(x^*, \lambda, \mu)$ is neg. def. (for all v), then x^* is a local maximum
  - if λ and μ are Lagrange multipliers corresponding to x^* in FOCs and $D_x^2 L(x, \lambda, \mu)$ is neg. def. (for all v) and for all feasible x, then x^* is a global maximum

Investigating definiteness

- Determining the definiteness of $D_x^2 L$ on linear constraints on set $C = \{ v \in \mathbb{R}^n : Dh(x^*) v = \mathbf{0} \}$
  - determine the definiteness of the bordered Hessian

$$H = \begin{pmatrix} \mathbf{0} & Dh(x) \\ Dh(x)^T & D_x^2 L(x, \mu) \end{pmatrix}$$

  - if $h : \mathbb{R}^n \mapsto \mathbb{R}^m$, then we need to check the last leading $(n - m)$ principal minors: if they alternate in sign and $\det(H)$ has the same sign as $(-1)^n$, then $D_x^2 L(x^*, \mu)$
  - example: $n = 2$ and $m = 1$, it is enough to compute the determinant of H, if it positive, then $D_x^2 L(x, \mu)$ is neg. def.

Recipe for checking the sufficient conditions

- Let us assume that we have a candidate for an optimum from FOCs

1. Does the problem seem (quasi)concave?
   - if yes, show it, if no proceed to 2.
2. Does it seem possible that $D^2 L$ is neg. def. for all v?
   - if yes, show it, otherwise proceed to step 3.
   - note: $D^2 f$ is part of $D^2 L$, thus, it makes sense to begin by checking whether $D^2 f$ is neg. def.
   - if $D^2 f$ is not neg. def. then it is unlikely that $D^2 L$ is neg. def. for all v
   - if $D^2 f$ is neg. def. then check the Hessians of constraints $D^2 g_i$, $i = 1, \ldots, k$ (and equality constraints)
   - if all Hessians are pos. def. (neg. def.) for constraints with positive (negative) Lagrange multiplier, then show that $D^2 L$ is neg. def. for all v.

Recipe for checking the sufficient conditions

3. Determine the definiteness of the bordered Hessian
   - alternatively you may check the definiteness of $D^2 L$ on C directly by using the definition; eliminate some of the variables (components of v) from the equation of the tangent set, plug the variables in the expression of $D^2 L$ and hope that you can now say something on the definiteness

Example

- $\max x_1^2 x_2$ s.t. $2x_1^2 + x_2^2 = 3$
- From FOCs we get six candidates for an optimum
$$(x_1, x_2, \mu) = \begin{cases} (0, \pm\sqrt{3}, 0) \\ (\pm 1, 1, 1/2) \\ (\pm 1, -1, -1/2) \end{cases}$$
- Hessian matrix of the objective function $\begin{pmatrix} 2x_2 & 2x_1 \\ 2x_1 & 0 \end{pmatrix}$ the determinant is $-4x_1^2 \leq 0$, i.e., the leading principal minors are $2x_2$ (1st lead. princ. minor) and $-4x_1^2$ (2nd lead. princ. minor), it seems that the matrix indefinite

Example

- Hessian of the constraint $\begin{pmatrix} 4 & 0 \\ 0 & 2 \end{pmatrix}$ matrix is pos. def.
  - at points $(x, \mu) = (\pm 1, 1, 1/2)$ we get $D_x^2 L = \begin{pmatrix} 0 & \pm 2 \\ \pm 2 & -1 \end{pmatrix}$ which is indefinite, however on $v \in C = \{v \neq \mathbf{0} : (4 \times (\pm 1), 2 \times 1) \cdot v = 0\}$ the quadratic form defined by the Hessian is $v \cdot D^2 L v = -4v_1^2 \pm v_1^2 < 0$, i.e., the points are local maxima

Example

- What about the rest of the solution candidates?
  - $x = (0, \pm\sqrt{3})$: when $x_2 \geq 0$ ($x_2 \leq 0$) the objective function is at least (at most) zero. Hence, $x = (0, \sqrt{3})$ is local minimum and $x = (0, -\sqrt{3})$ is local maximum
  - at $x = (\pm 1, -1)$ we get $D_x^2 L = \begin{pmatrix} 0 & \pm 2 \\ \pm 2 & 1 \end{pmatrix}$, which is indefinite, the bordered Hessian is $\begin{pmatrix} 0 & \pm 4 & -2 \\ \pm 4 & 0 & \pm 2 \\ -2 & \pm 2 & 1 \end{pmatrix}$
  - in this case $n = 2$ and $m = 1$, i.e., we need check the determinant of the matrix which is -48 implying that $D_x^2 L$ is pos. def. on C which means that these points are local minima

Introduction to Programming and Matlab

Matlab tutorial

Introduction

- Elements of programming
  - core: syntax (language) + logic + math
  - creativity + rigidity
  - learning by doing; work + making mistakes
- Motivation for economists
  - most problems are too complicated to be solved with paper and pencil
  - way to gain deeper understanding of economic models, e.g. finding "new phenomena"
  - trend: computers become more efficient, computation costs decrease

Programming as a part of academic work

1. Analyzing problem; creating a model
2. Designing an algorithmic solution
3. Programming the solution in a particular programming language
4. Testing the correctness; going back to previous steps
5. Revising and extending the model and the software

Programming languages

- Lay out an algorithm in a form that computer "understands" it
  - a programming language needs to be precise enough and close to human languages
- Elements of a programming language
  - commands ("vocabulary")
  - rules how to combine commands
- Learning to program is comparable to learning a language
  - note: programming languages are not forgiving to mistakes

Overview on programming languages

- High-level programming languages
  - languages close to English (note: high vs. low level is relative)
  - need to be translated (compiled) to instructions that the computer hardware understands
- Compiled languages
  - a separate compilation step is needed before program execution
  - examples: C, C++. Java, Fortran
- Interpreted languages
  - program is compiled (translated) during the execution
  - examples: Matlab, Python, Lisp

Some examples and curiosities

- R
- GAUSS
- Stata
- GAMS
- Mathematica
- APL: a functional language
- Octave: free alternative to Matlab

Matlab

- Matlab is interactive system and programming language for general scientific and technical computation and visualization
- History
  - developed in 1079's by Cleve Moler
  - the purpose was to develop an interactive program to access LINPACK and EISPACK (efficient Fortran linear algebra packages)
  - the name comes from "Matrix Laboratory"
  - rewritten in C in 1980's
  - Mathworks was founded in 1984

Features of Matlab

- Designed for scientific computing
- Relatively easy to learn
- Quick when preforming Matrix operations
- Can be used to build graphical user interfaces that function as stand-alone programs
- Has a huge number of Toolboxes
- Has good built in graphics for plotting data and displaying images
- Matlab as a programming language
  - is an interpreted language; writing programs is easy, errors are easy to fix, slower than compiled languages
  - has object-oriented elements
  - is NOT general purpose programming language

Basics of Using Matlab

variables and elementary operations

Basic principles

- In Matlab all numeric data is stored in various size of arrays
  - the elementary (algebraic) operators apply to arrays (or matrices = array with two dimensions)
  - state of mind: everything is arrays (crucial for writing efficient code and understanding Matlab programming)
- Note: Matlab can be used as a calculator for scalars
  - the usual scalar operators: $+$, $-$, $*$, $/$, ^, parentheses
  - relational operators: $<$, $>$, $<=$, $>=$, $==$, ~=
  - relational operators: & (and), | (or), ~(complement), xor
  - standard algebraic rules apply

Numerical variables

- Entering a variable: $=$
  - name of variable is a sequence of letters and numbers (no spaces, no numbers in the beginning), name cannot be a reserved keyword
- Entering scalars, matrices, and vectors
  - examples 1, entering a scalar:
    `a=1`
  - example 2, entering a 2×2 matrix:
    `A= [1 0; 2 3];`
  - note: the role of semicolon
  - example 3, entering a row vector:
    `A= [1 0 2 3];`
  - example 4, entering a column vector:
    `A= [1; 0; 2; 3];`
- Accessing the i,j element of an array:
  `A(i,j)`

Other data types

- Character arrays
  - `A = 'string'`
- Structures
  - used for grouping data, example:
    ```
    persons(1).name='John';
    persons(1).age=35;
    persons(2).name='Diana';
    persons(2).age=60
    ```
- Cell arrays
  - elements can contain any data, example:
    ```
    A{1}=ones(3);
    A{2}='John';
    A{3}=persons
    ```

Generating arrays

- Colon operator
  - syntax: variable=startvalue:stepsize:endvalue
  - example:
    `a=0:2:6;`
- Commands for creating arrays
  - arrays of zeros; zeros, array of ones; ones
  - diagonal matrices; diag, identity matrix; eye
  - random arrays: rand, randn, randi
- Initializing variables
  - Matlab may execute faster if the zeros (or ones) function is used to set aside storage for an array before its elements are generated

Handling arrays

- Arrays can be concatenated
  - example: A=[B C];
- A(r_1:r_2,k_1:k_2) returns the block consisting of the elements k_1 up to and including k_2 from rows r_1 up to and including r_2 from A
  - respectively if there are more dimensions
  - note: A(r_1:r_2,:) picks the matrix consisting of rows from r_1 to r_2
- A(v_1,v_2) returns the elements in the rows specified in vector v_1 and the columns specified in vector v_2
- "end" refers to the size of the relevant dimension
- Examples:
  ```
  A=[magic(6) ones(6,1)];
  B=A([1,3], [2,4])
  a=A(2,:)
  B=A(2:4,3:end)
  ```

Information about variables

- To show a list of variables stored in memory use who and whos
- To clear variables use clear
  - during programming, when testing your scripts, it may be worth clearing the variables every now and then
  - other clear commands: clf and clc
- Array functions
  - information on arrays: size and length (for vectors)
  - reshape, repmat, squeeze
- Saving data
  - save command (and load) and diary
- Importing data
  - loading .txt files: importdata, dlmread, csvread, fopen, ...
  - reading .xls files: xlsread

Basic maths

- $+$ and - work for arrays of the same size
- * is matrix product
  - example:
    `a=[1 2; 3 4]*[3;6]`
- A^n is used to raise matrix A to a power n
- ' is transpose
- To make an operation ($+$,-,*,/,^) element-wise use . before the operator
  - example:
    ```
    a=[1 2; 3 4].*[5 6; 7 8];
    b=[1 2; 3 4]*[5 6; 7 8];
    ```
  - note: many built-in functions operate element-wise

Logical expressions

- Relational operators
  - smaller than (<), smaller or equal than (<=) respectively, larger than, equal ==, not equal ~=
  - outcome of logical expression is either true (1) or false (0)
  - other commands: any, all
- Logical conditions can be combined into logical expressions using logical operators
  - and (&), or (|), not ~
- Relational and logical operators can be used over arrays
  - for scalars it is recommended to use && (short-circuit and) instead of & and || instead of |
  - the outcome of logical condition is a logical array
- Example:

```
x=rand(10,1);
a= (x>.4 & x<.6)
```

Graphs and Plots in Matlab

2d and 3d graphics and image manipulation

Basic 2D plotting

- plot command
  - plot(y) plots vector y against its indices
  - plot(x,y) plots vector y against x
  - fplot can be used for function plots (no need for arrays)
  - example:

    `x=1:100; y=x.^2-5; plot(x,y)`
- Style commands, see help plot
  - line styles: dashed, dotted, dash dotted
  - plot markers: $+$, $*$, o, x
  - line colors, line width etc
- New figure window: figure

Plot commands

- Labels, title and legends
  - xlabel, ylabel, title, legend, text
  - note: Matlab understands some latex
- Multiple plots in same figure
  - hold on (and hold off)
  - use plot(x1,y1,x2,y2)
  - combine multiple y values in a matrix: plot(x,[y1 y2])
- Miscellaneous
  - grid on (or off), axis box, axis square, box on (off)
  - axis range: axis([xmin xmax ymin ymax])
- Example:

```
figure(2); x=1:100; y=x.^2-5; plot(x,y,'k-.');
grid on; axis([10 50 0 10000]), axis square,
y2=x.^(1.5)-10; hold on; plot(x,y2,'b--');
xlabel('x'); title('example')
```

3D plots

- Use plot3 to draw 3d lines
  - example:

```
z=[0:pi/50:10*pi];x=exp(-.2.*z).*cos(z);
y=exp(-.2.*z).*sin(z); plot3(x,y,z);
```
- Surfaces and contours
  - to evaluate a function at given x, y-grid-points use meshgrid to create the grid
  - use surf to plot the surface
  - use contour to plot contours
  - example:

```
[x,y]=meshgrid(-5:.5:5,-10:.5:10); z=x.^2+2*sin(y);
figure(1); surf(x,y,z);
figure(2); contour(x,y,z)
```

Useful graphics commands

- Several plots in the same figure: subplot(m,n,p)
  - splits the figure into $m \times n$ matrix, p identifies the part where the next plot will be drawn
- Bars, pies, histograms
  - example:

```
subplot(2, 2, 1); bar([2 3 1 4; 1:4]), title('Bar graph { multiple series')
subplot(2, 2, 2); bar3([2 3 1 4; 1:4]),
title('3D bar graph')
subplot(2, 2, 3); pie([1 2 3 4]), title('Pie chart')
subplot(2, 2, 4); hist(randn(1000,1))
```
- Exporting figures
  - use print command, example:

```
print -deps test.eps
```

Programming with Matlab

scripts, loops and conditional expressions, functions

Introduction

- It is possible to write Matlab commands (or scripts) in m-files and execute these files
  - m-files can be called/run in Matlab by their name
  - anything in the Matlab workspace can be called from a script
  - note: m-files are text
- *Functions* are computer codes that accept inputs from the user, carry out computations, and produce outputs
  - functions are created in m-files
  - functions differ from scripts in having a function declaration (more later) and you cannot (unless you necessarily want) call variables in Matlab workspace inside a function
  - note: many Matlab functions are implemented as m-files, to view the code use edit command
- Note: comments are added in m-files by starting the comment with %
  - in general you cannot comment your codes too much

For loops

- Syntax to repeat execution of a block of code
  - basic syntax:
    ```
    for counter=startvalue:stepsize:endvalue
     % some commands
    end
    ```
  - the following syntax is also possible
    ```
    for counter=[vector]
     % some commands
    end
    ```
  - example:
    ```
    n=100; x=1:n;
    for i=2:5:n
     x(i)=x(i-1)+i;
    end
    plot(x)
    ```

Conditional execution with if/else structure

- Syntax
  ```
  if logical_expression1
   % commands executed if expression1 is true
  elseif logical_expression2
   % commands executed if logical_expression1 is false
   % but logical_expression_2 is true
  else
   % executed in all other cases
  end
  ```
  - note: else and elseif are not necessary
  - logical expression is composed of relational and logical operators
  - example:
    ```
    if x >= 0
    y = sqrt(x);
    else
    disp('Number cannot be negative');
    end
    ```

Conditional execution with Switch structure

- Alternative for if structure
- Syntax:
  ```
  switch variable
  case option1
   statements ...
  case option n
   statements
  otherwise
   statements
  end
  ```
- Example:
  ```
  state = input('Enter state: ')
  switch state
  case {'TX', 'FL', 'AR'}
   disp('State taxes: NO');
  otherwise
   disp('State taxes: YES')
  end
  ```

While loop

- Syntax:
  ```
  while logical_expression
   commands
  end
  ```
  - the loop executes the commands as long as logical expression is true
  - example:
    ```
    n = input('Enter nr > 0: '); f = 1;
    while n > 0
     f = f .* n
     n = n - 1
    end
    ```
- Breaking loops
  - break command inside loop (for or while) terminates the loop
  - note: any Matlab script can be stopped from running by pressing control + c

Interaction with the user

- Receiving input from the user
  - as function arguments (to come)
  - input command, e.g., before "input('Enter state: ')"
- Displaying output
  - disp command, example:
    ```
    A = [1 4];disp('The entries in A are'); disp(A);
    ```
  - printf (allows specifying the output type), example:
    ```
    x = 2;
    printf('The value of x as integer is %d \n
    and as real is %f', x, x);
    ```
- Number display can be controlled by format command

Writing a function

- Each function starts with a function-definition line that contains
  - the word function
  - a variable that defines the function output
  - a function name
  - input argument(s)
  - example:
    ```
    function avg = Average3(x, y, z)
    avg = (x + y + z) / 3;
    ```
- Function name and name of M-file should be the same
  - function name has to be a valid Matlab variable name
  - recall that Matlab is case sensitive
  - m-file without function declaration is a "script" file
- Using help
  - comment lines immediately following the first function line are returned when help is queried

More advanced function programming

- Inputs and outputs
  - nargin and nargout, varargin
  - note: it is not necessary to give all arguments (unless needed) or to take all outputs
  - note: it is possible to write a function without arguments
- Subfunctions
  - complex functions can be implemented by grouping smaller functions (called subfunctions) together in a single file
  - any function declaration after the initial one is a subfunction to the main function, a subfunction is visible to all functions declared before it
  - example:

```
function [output1,output2]=myfunction(input1)
 % some commands
c=sub_myfunction(a,b);
 % some commands
function output1=sub_myfunction(input1,input2)
```

Local and global variables

- Variables inside a function in an M-file are local variables
  - local variables are not defined outside the function
  - subfunctions have their own part of memory and cannot access variables of the main function (unless they are shared via global)
- Global variables are available to all functions in a program
  - warning: it might become unclear what a global variable contains at a certain point as any intermediate function could have accessed it and could have changed it (hence avoid them)
  - global variables are typically reserved for constants
  - global variables are declared in the workspace using the keyword global
  - when used inside a function, the function must explicitly request the global variable by name

Miscellaneous: directories, diagnostics

- How does Matlab search for function files?
  - first from the current working directory, then from other directories specified in the path
  - useful path command: path, pathtool
- Debugging and enhancing your code
  - Matlab editor has a debugger tool
  - profile and tic (toc) commands can be used to analyze computation times
- Function Handles
  - useful for functions of functions, example: f = @fun creates a function handle "f" that references function "fun"

Tips for efficiency

- Avoid loops and vectorize instead
- Use functions instead of scripts
- Avoid overwriting variables
- Do not output too much information to the screen
- Use short circuit && and || instead of & and |
- Use the sparse format if your matrices are very big but contain many zeros
- Finally: there is a trade-off between speed, precision, and programming time
  - don't spend an hour figuring out how to save yourself micro seconds of computing time

Dynare

basics on writing a Dynare model,
material: Dynare user guide at www.dynare.org

Introduction

"DYNARE is a pre-processor and a collection of MATLAB routines that has the great advantages of reading DSGE model equations written almost as in an academic paper." (DYNARE User Guide)
DYNARE can be used to:

- compute the steady states of DSGE models
- compute the solutions of deterministic models
- compute first and second order approximations to solutions of stochastic models
- estimate parameters of DSGE models
- compute optimal policies in linear-quadratic models

See www.dynare.org

The purpose

- Finding a solution to a dynamic system of the form $\mathbb{E}_\varepsilon[f(y_{t+1}, y_t, y_{t-1}, \varepsilon_t; \theta)] = \mathbf{0}$
  - such systems arise from stochastic macro models
  - y_t are endogenous variables, ε_t are exogenous shocks (expected value zero), θ are exogenous parameters
  - a solution is a function (e.g., policy) $y_t = g(y_{t-1}, \varepsilon_t)$, note that f is known
  - in the deterministic case ε_t are known (no expectation) and solution is simply a path of y's
- Dynare gives linearization of g around the steady state
  - at the steady state $\bar{y}$ it holds that $\mathbb{E}_\varepsilon[f(\bar{y}, \bar{y}, \bar{y}, \varepsilon; \theta)] = 0$
  - the first order Dynare solution is of the form $y_t = \bar{y} + A(y_{t-1} - \bar{y}) + B\varepsilon_t$, second order approximation respectively

Example

- Optimal growth model

$$\max_{k_t, c_t} \mathbb{E}\left[\sum_{t=0}^{\infty} \beta^t (c_t^{1-\nu} - 1)/(1-\nu)\right] \tag{1}$$

$$c_t + k_t = z_t k_{t-1}^{\alpha} + (1-\delta)k_{t-1} \tag{2}$$

$$z_t = (1-\rho) + \rho z_{t-1} + \varepsilon_t \tag{3}$$

- The FOC's lead to a system to be solved: equations (2), (3), and $c_t^{-\nu} = \mathbb{E}\left[\beta c_{t+1}^{-\nu}(\alpha z_{t+1} k_t^{\alpha-1} + 1 - \delta)\right]$
  - here $y = [c, k, z]$

Dynare files

- Dynare models are written in .mod files
  - use the Matlab editor as if writing an m-file
  - when saving, it is important to append the extension '.mod' instead of '.m'.
- There are four major blocks in a mod-file
  - The preamble, listing the variables and the parameters
  - The model, containing the equations
  - Initial values for linearization around the steady state
  - The variance-covariance matrix for the shocks

Dynare file blocks

- The preamble
  - variables, example:
    `var c, k, z;`
  - exogenous variables, e.g., shocks:
    `varexo e;`
  - parameters and their values:
    ```
    parameters alpha, beta, delta, nu, rho;
    alpha=.36; beta=.99; delta=.025; nu=2; rho=.95;
    ```
- The model (x($\pm$n) denotes $x_{t\pm n}$)

```
model;
c^(-nu)=beta*c(+1)^(-nu)*(alpha*z(+1)*k^(alpha-1)+1-delta);
c+k=z*k(-1)^alpha+(1-delta)*k(-1);
z=(1-rho)+rho*z(-1)+e;
end;
```

Dynare file blocks

- The model block (cont'd)
  - all the relevant equations need to be listed, also those for the shock process
  - there should be as many equations as there are endogenous variables
  - the timing reflects when a variable is decided, note contemporaneous are without parenthesis
- Initialization block
  - initial values for finding the steady state
  - it can be tricky to figure out a good initial guess!
    ```
    initval;
    c = 1; k = .8; z = 11;
    end;
    ```

Dynare file blocks

- Random shock block
  - indicate the standard deviation for the exogenous innovation
    ```
    shocks;
    var e; stderr 0.007;
    end;
    ```
  - covariances are given as var e, u = ...; (cov. of e and u)
- Solution and properties block
  - the solution is found by command stoch_simul; which (1) computes a Taylor approximation around the steady state of order one or two (2) simulates data from the model (3) computes moments, autocorrelations and impulse responses
  - inputs of stoch_simul: *periods* is the number of periods to use in simulations (default=0), *irf* is the number of periods on which to compute the impulse response functions (default=0), *nofunctions* suppresses the printing of the approximated solution. *order* is the order of the Taylor approximation (1 or 2)

Dynare outputs

- Dynare is run in Matlab by "dynare modelname"
- Outputs are saved in a global variable oo_ which contains the following fields
  - steady_state contains the steady states of the variables (in the order used in the preamble)
  - mean, var (variances), autocorr (autocorrelations) etc
- Saving variables
  - use "dynasave(FILENAME) [variable names separated by commas]" at the end of .mod file to save variables
  - variables can be retrieved by load -mat FILENAME
  - note: Dynare clears all variables in the memory unless called by "dynare program.mod noclearall"

Dynare outputs

- The main output is the linearized (or second order approx) policy function "g" around the steady state
- Example:

```
POLICY AND TRANSITION FUNCTIONS
          k          z         c
constant 37.989254 1.000000 2.754327
k(-1)     0.976540 -0.000000 0.033561
z(-1)     2.597386  0.950000 0.921470
e         2.734091  1.000000 0.969968
```

  - interpretation, e.g., $k_t = \bar{k} + 0.97(k_{t-1} - \bar{k}) + 2.59(z_{t-1} - \bar{z}) + 2.73\varepsilon_t$, where $\bar{k}$ and $\bar{z}$ are steady-state values (the constants on the first line)

Loops

- Running the same Dynare program for several parameter values
- Write a Matlab script (or function) in which you loop over the relevant parameter values
  - for each value of *parameter* use "save parameterfile parameter"
  - in the Dynare file call "load parameterfile" and replace the part in which the parameter value is set with "set_param_value('·',·)"; the first argument is the name in the .mod file and the second is the numerical value

Scope and limitations

- Dynare has tools for LQ models, estimation, and regression
- Finding steady-states
  - use other Matlab routines
- Limitations
  - difficult to modify the Dynare code for "non-standard" calculations
  - difficult to detect errors
  - documentation?
  - limited to approximations, e.g., in dynamic programming the are other methods (implemented e.g., in CompEcon toolbox)