

Aplikasi Algoritma Genetika

SANJOYO

JUNI 2006

Daftar Isi

1	Pendahuluan	1
1.1	Latar Belakang	1
1.2	Perumusan Masalah	2
1.3	Tujuan Eksperimen	2
1.4	Metode Eksperimen	2
2	Teori Pendukung	4
2.1	Skema Pengkodean	5
2.2	Nilai Fitness	6
2.3	Seleksi Orang Tua	6
2.4	Pindah Silang (Cross-over)	7
2.5	Mutasi	9
2.6	Elitisme	9
3	Prosedur Eksperimen	11
4	Hasil Eksperimen dan Analisis	14
4.1	Fungsi Produksi Cobb- Dauglas	14
4.1.1	Least Square Error	15
4.1.2	Maksimum Likelihood	16
4.2	Fungsi Produksi CES	17
4.2.1	Least Square Error	17
4.2.2	Maksimum Likelihood	18
4.3	Model Terbaik	19
5	Kesimpulan	21
	Lampiran	22

Bab 1

Pendahuluan

1.1 Latar Belakang

Penaksiran model Regresi Non-Linier dengan menggunakan metode iterasi dengan Algoritma Gause-Newton dalam mengestimasi parameter fungsi produksi Cobb-Dauglas maupun fungsi produksi CES, masih menyisakan pertanyaan tentang jaminan tentang terjadinya konvergensi, apakah nilai optimum yang diperoleh benar-benar sebagai global optimum. Selain itu, metode iterasi tersebut sangat sulit untuk menentukan nilai awal yang mencapai konvergensi dan diperlukan *trial error*.

Algoritma Genetika adalah salah satu pendekatan untuk menentukan global optimum yang didasari oleh Teori Darwin. Secara garis besar langkah dalam prosedur ini dimulai dengan menetapkan suatu set solusi potensial dan melakukan perubahan dengan beberapa iterasi dengan algoritma genetika untuk mencapai solusi terbaik. Set solusi potensial ini ditetapkan diawal dan disebut dengan kromosom. Kromosom ini dibentuk secara random berupa susunan angka binary yang di-generate dan dipilih. Keseluruhan set dari kromosom yang diobservasi mewakili suatu populasi.

Kemudian, kromosom-kromosom tersebut akan berevolusi dalam beberapa tahap iterasi yang disebut dengan generasi. Generasi baru (offsprings) di-generate dengan teknik kawin silang (crossover) dan mutasi (mutation). Cross over meliputi pemecahan (splitting) dua kromosom dan kemudian mengkombinasikan setengah bagian dari masing-masing kromosom dengan pasangan-pasangan lainnya. Sedangkan mutasi meliputi penggantian (flipping) satu bit (bagian) dari kromosom dengan satu bagian lain dari kromo-

som lain yang menjadi pasangannya. Kromosom-kromosom ini selanjutnya berevolusi dengan suatu kriteria kesesuaian (fitness) yang ditetapkan dan hasil terbaik akan dipilih sementara yang lainnya diabaikan.

Selanjutnya, proses dilakukan berulang-ulang sampai dengan suatu kromosom yang mempunyai kesesuaian terbaik (best fitness) akan diambil sebagai solusi terbaik dari permasalahan. Keunggulan dari algoritma genetika adalah berproses sangat baik untuk global optimization khususnya bilamana fungsi objektif adalah diskontinu atau mempunyai beberapa local minima.

1.2 Perumusan Masalah

Permasalahan yang dihadapi dalam eksperimen ini adalah sebagai berikut:

1. Bagaimana penaksiran model fungsi produksi Cobb-Douglas dan fungsi produksi CES dengan metode Algoritma Genetika?
2. Bagaimana simulasi dilakukan agar dapat diperoleh global optimum?
3. Bagaimana penaksiran pemilihan model terbaik dengan metode Algoritma Genetika?

1.3 Tujuan Eksperimen

Tujuan eksperimen ini adalah untuk mengaplikasikan Algoritma Genetika pada proses penaksiran model fungsi Cobb-Douglas dan CES. Hasil dari Algoritma Genetika diharapkan berupa nilai yang dianggap paling optimum dan memenuhi fungsi tujuan. Adapun fungsi tujuannya adalah meminimumkan least square dan memaksimumkan likelihood function untuk setiap model CD dan CES.

1.4 Metode Eksperimen

Metode yang akan digunakan dalam eksperimen ini, pertama dengan memahami konsep Algoritma Genetika dari literatur, referensi dan catatan kuliah. Kedua, mempelajari hasil praktek program aplikasi algoritma genetika di kelas dan memodifikasi program sesuai dengan tujuan dari eksperimen ini. Ketiga, menjalankan eksperimen ini dengan aplikasi yang telah dimodifikasi

berdasarkan data fungsi produksi sesuai dengan tugas ke dua yang telah diberikan. Program aplikasi Algoritma Genetika ditulis dalam bahasa program Matlab 7.0.1. Hasil eksperimen yang diperoleh kemudian akan dikaji dan dianalisa.

Bab 2

Teori Pendukung

Sejak algoritma genetika (AG) pertama kali dirintis oleh John Holland dari Universitas Michigan pada tahun 1960-an, AG telah diaplikasikan secara luas pada berbagai bidang. AG banyak digunakan untuk memecahkan masalah optimasi, walaupun pada kenyataannya juga memiliki kemampuan yang baik untuk masalah- masalah selain optimasi. John Holland menyatakan bahwa setiap masalah yang berbentuk adaptasi (alami maupun buatan) dapat diformulasikan dalam terminologi genetika. Algoritma genetika adalah simulasi dari proses evolusi Darwin dan operasi genetika atas kromosom.

Pada algoritma genetika, teknik pencarian dilakukan sekaligus atas sejumlah solusi yang mungkin dikenal dengan istilah populasi. Individu yang terdapat dalam satu populasi disebut dengan istilah kromosom. Kromosom ini merupakan suatu solusi yang masih berbentuk simbol. Populasi awal dibangun secara acak, sedangkan populasi berikutnya merupakan hasil evolusi kromosom-kromosom melalui iterasi yang disebut dengan generasi. Pada setiap generasi, kromosom akan melalui proses evaluasi dengan menggunakan alat ukur yang disebut dengan fungsi fitness. Nilai fitness dari suatu kromosom akan menunjukkan kualitas dari kromosom dalam populasi tersebut. Generasi berikutnya dikenal dengan istilah anak (offspring) terbentuk dari gabungan dua kromosom generasi sekarang yang bertindak sebagai induk (parent) dengan menggunakan operator penyilangan (crossover). Selain operator penyilangan, suatu kromosom dapat juga dimodifikasi dengan menggunakan operator mutasi. Populasi generasi yang baru dibentuk dengan cara menyeleksi nilai fitness dari kromosom induk (parent) dan nilai fitness

dari kromosom anak (offspring), serta menolak kromosom-kromosom yang lainnya sehingga ukuran populasi (jumlah kromosom dalam suatu populasi) konstan. Setelah melalui beberapa generasi, maka algoritma ini akan konvergen ke kromosom terbaik.

Ada tiga keunggulan dari aplikasi Algoritma Genetika dalam proses optimasi, yaitu: (a) Algoritma Genetika tidak terlalu banyak memerlukan persyaratan matematika dalam penyelesaian proses optimasi. Algoritma Genetika dapat diaplikasikan pada beberapa jenis fungsi obyektif dengan beberapa fungsi pembatas baik berbentuk linier maupun non-linier; (b) Operasi evolusi dari Algoritma Genetika sangat efektif untuk mengobservasi posisi global secara acak; dan (c) Algoritma Genetika mempunyai fleksibilitas untuk diimplementasikan secara efisien pada problematika tertentu.

Berikut ini penjelasan sistim operasi algoritma genetika yang sumber utamanya berasal dari Suyanto, Yingsong Zheng dan Sumio Kiyooka serta catatan kuliah ekonometrik 3, sebagai berikut:

2.1 Skema Pengkodean

Misalkan kita ingin memecahkan masalah optimasi fungsi produksi Cobb-Dauglas yaitu $y = \beta_1 L^{\beta_2} K^{\beta_3}$ dengan sample yang ada untuk L dan K berapa nilai $\beta_1, \beta_2, \beta_3$ dengan fungsi tujuan meminimumkan least square atau memaksimumkan fungsi likelihood. Dengkian pula untuk persoalan yang sama pada fungsi produksi CES. Persoalan tersebut dapat diselesaikan dengan AG, yaitu: ketiga parameter $\beta_1, \beta_2, \beta_3$ dikodekan dalam kromosom. Masing-masing kromosom berisi sejumlah gen, yang mengkodekan informasi yang disimpan di dalam kromosom. Misalkan untuk memudahkan digunakan binary encoding dengan panjang kromosom 12 gen (12 bits), masing-masing parameter $\beta_1, \beta_2, \beta_3$ dikodekan dengan 4 gen, sehingga dapat diilustrasikan skema pengkodean pada Gambar 1 dibawah ini:

Gambar 1: Skema Binary Encoding

Parameter	β_1				β_2				β_3			
Binary number	1	0	1	1	1	1	1	0	1	0	1	0
	g1		g4		g5		g8		g9		g12	
Decimal number	11				14				3			

Bilamana nilai parameter yang akan kita cari mempunyai konstraint

yaitu $a < \beta < b$ maka berdasarkan binary encoding, nilai parameter dapat diperoleh dengan formula $\beta = a + \beta_{dec} * (\frac{b-a}{2^n-1})$ dan misalkan n adalah banyaknya gen (bits) yaitu 4 untuk setiap parameter dan konstraint $0 < \beta < 1$, sehingga:

$$\beta_1 = 0 + 11 * (\frac{1-0}{2^4-1}) = 0,7333$$

$$\beta_2 = 0 + 14 * (\frac{1-0}{2^4-1}) = 0,9333$$

$$\beta_3 = 0 + 3 * (\frac{1-0}{2^4-1}) = 0,2$$

Setelah skema pengkodean ditentukan, AG diinisialisasi untuk sebuah populasi dengan N kromosom. Gen-gen yang mengisi masing-masing kromosom dibangkitkan secara random. Masing-masing kromosom akan dikodekan menjadi individu dengan nilai *fitness* tertentu. Kemudian sebuah populasi baru akan dibentuk dengan menggunakan mekanisme seleksi alamiah, yaitu memilih individu-individu secara proporsional terhadap nilai *fitness*-nya, dan genetika alamiah, yakni pindah silang (crossover) dan mutasi.

Pada algoritma genetika yang akan digunakan adalah dengan skema pergantian populasi yang disebut *generational replacement*, artinya, N kromosom dari suatu generasi digantikan sekaligus oleh N kromosom baru hasil pindah silang dan mutasi.

2.2 Nilai Fitness

Suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran performansinya. Di dalam evolusi alam, individu yang bernilai fitness tinggi yang akan bertahan hidup. Sedangkan individu yang bernilai fitness rendah akan mati. Pada masalah optimasi, solusi yang akan dicari adalah memaksimalkan sebuah fungsi likelihood dan meminimumkan least square baik untuk fungsi produksi Cobb-Dauglas maupun fungsi produksi CES.

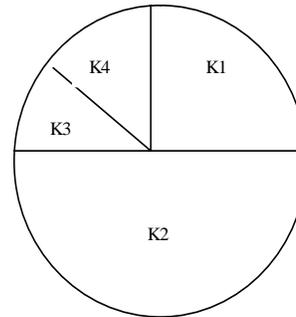
2.3 Seleksi Orang Tua

Pemilihan dua buah kromosom sebagai orang tua, yang akan dipindahsilangkan, biasanya dilakukan secara proporsional sesuai dengan nilai fitness-nya. Suatu metoda seleksi yang umumnya digunakan adalah *roulette wheel* (roda roulette). Sesuai dengan namanya, metoda ini menirukan permainan *roulette wheel* di mana masing-masing kromosom menempati potongan lingkaran pada roda roulette secara proporsional sesuai dengan nilai

fitnessnya. Kromosom yang memiliki nilai fitness lebih besar menempati potongan lingkaran yang lebih besar dibandingkan dengan kromosom bernilai fitness rendah.

Gambar 2: Contoh penggunaan metoda roulette wheel selection.

Komosome	Nilai Fitness
K1	1
K2	2
K3	0,5
K4	0,5
Jumlah	4



Metoda roulette-wheel selection sangat mudah diimplementasikan dalam pemrograman. Pertama, dibuat interval nilai kumulatif dari nilai fitness masing-masing kromosom. Sebuah kromosom akan terpilih jika bilangan random yang dibangkitkan berada dalam interval kumulatifnya. Pada Gambar 2 di atas, K1 menempati interval kumulatif $[0;0,25]$, K2 berada dalam interval $(0,25;0,74]$, K3 dalam interval $(0,75;0,875]$ dan K4 berada dalam interval $(0,875;1]$. Misalkan, jika bilangan random yang dibangkitkan adalah 0,6 maka kromosom K2 terpilih sebagai orang tua. Tetapi jika bilangan random yang dibangkitkan adalah 0,9 maka kromosom K4 yang terpilih.

2.4 Pindah Silang (Cross-over)

Salah satu komponen yang paling penting dalam algoritma genetik adalah crossover atau pindah silang. Sebuah kromosom yang mengarah pada solusi yang baik dapat diperoleh dari proses memindah-silangkan dua buah kromosom.

Gambar 3: Contoh Proses Pindah Silang

	β_1				β_2				β_3			
Orang tua 1	0	0	1	1	1	1	1	1	1	1	1	1
Orang tua 2	1	1	0	0	0	0	0	0	0	0	0	0
	g1		g4		g5		g8		g9		g12	
Anak 1	0	0	0	0	0	0	0	0	0	0	0	0
Anak 2	1	1	1	1	1	1	1	1	1	1	1	1

Pindah silang juga dapat berakibat buruk jika ukuran populasinya sangat kecil. Dalam suatu populasi yang sangat kecil, suatu kromosom dengan gen-gen yang mengarah ke solusi akan sangat cepat menyebar ke kromosom-kromosom lainnya. Untuk mengatasi masalah ini digunakan suatu aturan bahwa pindah silang hanya bisa dilakukan dengan suatu probabilitas tertentu, artinya pindah silang bisa dilakukan hanya jika suatu bilangan random yang dibangkitkan kurang dari probabilitas yang ditentukan tersebut. Pada umumnya probabilitas tersebut diset mendekati 1. Pindah silang yang paling sederhana adalah pindah silang satu titik potong (one-point crossover). Suatu titik potong dipilih secara random, kemudian bagian pertama dari orang tua 1 digabungkan dengan bagian kedua dari orang tua 2 (terlihat pada gambar 3).

Crossover adalah operator Algoritma Genetika yang utama karena beroperasi pada dua kromosom pada suatu waktu dan membentuk offspring dengan mengkombinasikan dua bentuk kromosom. Cara sederhana untuk memperoleh *crossover* adalah dengan memilih suatu titik yang dipisahkan secara random dan kemudian membentuk *offspring* dengan cara mengkombinasikan segmen dari satu induk ke sebelah kiri dari titik yang dipisahkan dengan segmen dari induk yang lain ke sebelah kanan dari titik yang dipisahkan. Metode ini akan berjalan normal dengan representasi bit string. Performa dari Algoritma Genetika bergantung pada performa dari operator *crossover* yang digunakan.

Crossover rate merupakan rasio antara jumlah offspring yang dihasilkan pada setiap generasi terhadap luas populasinya. Semakin tinggi crossover rate akan memungkinkan eksplorasi ruang solusi yang lebih luas dan mereduksi kemungkinan jatuh pada kondisi optimum yang salah. Namun memberikan rate yang memberikan konsekuensi makin lamanya waktu perhitungan yang diperlukan sebagai akibat eksplorasi pada luas populasi yang ada.

2.5 Mutasi

Mutasi dapat dilakukan dari semua gen yang ada dengan probabilitas mutasi tertentu. Jika bilangan random yang dibangkitkan kurang dari probabilitas mutasi yang ditentukan maka ubah gen tersebut menjadi nilai kebalikan yang dalam hal ini, binary encoding, 0 diubah 1, dan 1 diubah 0. Bila mana probabilitas mutasi adalah ($\frac{1}{12}$) maka sebanyak 1 gen akan dimutasi dari kromosom yang terdiri dari 12 gen (bits). Pada algoritma genetika yang sederhana, nilai probabilitas mutasi adalah tetap selama evolusi. Gambar 4 menunjukkan proses mutasi yang terjadi pada gen_5 .

Gambar 4: Contoh Proses Mutasi

	β_1				β_2				β_3			
Kromosom asal	0	0	0	1	1	1	1	1	1	1	1	1
	g1			g4	g5	g8		g9	g12			
Hasil mutasi	0	0	0	1	0	1	1	1	1	1	1	1

Mutasi dapat dikatakan sebagai operasi pendukung yang menghasilkan perubahan secara acak dan seketika pada berbagai jenis kromosom. Cara mudah untuk mendapatkan mutasi dengan mengubah satu atau lebih *genes*. Pada Algoritma Genetika, mutasi memainkan peran penting, yaitu pertama, menggantikan *genes* yang hilang dari populasi selama proses seleksi, sehingga dapat diujikan pada suatu kondisi yang baru. Kedua, menyediakan *genes* yang tidak ditampilkan pada populasi awal.

Mutation rate menyatakan presentase dari total jumlah *genes* dalam populasi. Mutation rate ini melakukan kontrol dimana *genes* baru dalam populasi dapat diuji seleksi. Jika rate terlalu kecil akan banyak *genes* yang sebenarnya bermanfaat tetapi tidak pernah diuji seleksi. Namun jika rate terlalu tinggi akan terjadi *random perturbation*, yang berakibat offspring mulai kehilangan kemiripan dengan induknya dan Algoritma Genetika akan kehilangan kemampuan untuk melihat urutan langkah observasinya.

2.6 Elitisme

Proses seleksi dilakukan secara random sehingga tidak ada jaminan bahwa suatu individu yang bernilai fitness tertinggi akan selalu terpilih. Walaupun individu bernilai fitness tertinggi terpilih, mungkin saja individu tersebut

akan rusak (nilai fitnessnya menurun) karena proses pindah silang. Oleh karena itu, untuk menjaga agar individu bernilai fitness tertinggi tersebut tidak hilang selama evolusi, maka perlu dibuat satu atau beberapa kopinya. Prosedure ini dikenal sebagai elitisme.

Bab 3

Prosedur Eksperimen

Pada eksperimen dengan Algoritma Genetika dilakukan langkah- langkah sebagai berikut:

1. Menentukan suatu initial populasi n yang dibentuk secara acak dengan l -bit kromosom sebagai kandidat solusi masalah.
2. Mengevaluasi fitness dengan menghitung fitness $f(x)$ dari setiap kromosom x dalam populasi. Proses evaluasi adalah dalam tiga tahap yaitu:
 - Konversikan binary encoding menjadi real value
 - Evaluasi objective function baik dengan meminimumkan least square dan maksimum likelihood untuk fungsi produksi Cobb-Dauglas dan CES.
 - Menghitung nilai $fitness\ eval(v_k)$ untuk setiap kromosom v_k .Bila- mana objective function adalah minimum least square maka yang terpilih adalah nilai minimum dari $fitness\ eval$, namun bila ob- jective function adalah maksimum likelihood maka yang dipilih adalah nilai maksimum $fitness\ eval$
3. Menciptakan populasi baru (generasi 2) dari yang berasal dari populasi awal (generasi 1) dengan tiga operator yaitu:
 - Mereproduksi 2 kromosom yaitu yang terbaik (best fitness) per- tama dan terbaik kedua dari generasi 1 untuk melanjutkan pada generasi 2.

- Menghitung total *fitness* dari populasi, yaitu:

$$F = \sum_{k=1}^{pop_size} eval(v_k)$$

- Menghitung peluang seleksi, p_k untuk setiap kromosom v_k :

$$p_k = \frac{eval(v_k)}{F}, k = 1, 2, \dots, pop_size$$

- Menghitung peluang kumulatif q_k untuk setiap kromosom v_k :

$$q_k = \sum_{j=1}^k p_j, k = 1, 2, \dots, pop_size$$

- Proses seleksi yang digunakan adalah one-cut-point yang dipilih secara random dan kemudian pada generasi 1 (parent) 2 kromosom yang berpasangan ditukar sebagian gennya (proses crossover) untuk memperoleh generasi 2 (offspring). Proses ini berulang untuk memperoleh 18 pasangan offspring dan 2 pasangan terbaik dari parent (bilamana populasi adalah 20).
- Melakukan proses mutasi dari 18 pasangan offspring (2 pasangan terbaik tidak dilakukan mutasi) dengan merubah gen 0 menjadi satu atau 1 menjadi 0 dan banyaknya gen yang dirubah bergantung pada mutation rate.

4. Tercipta populasi generasi ke dua.
5. Proses tersebut diatas berulang-ulang sampai dengan 10 ribu generasi atau 15 ribu generasi.

Data yang digunakan adalah sebagaimana yang digunakan pada Tugas 2 yaitu data produksi suatu komoditi (y) dengan input yang digunakan adalah kapital (K) dan tenaga kerja (L) (data terlampir bersama program). Model statistik non-linier berbentuk: $y = f(x, \beta) + e$. Dengan data tersebut akan diestimasi parameter fungsi produksi Cobb Douglas (CD) dan Constant Elasticity of Substitution (CES) dengan algoritma genetika. Baik untuk fungsi Cobb-Dauglas maupun CES, dimana $e \sim N(0, \sigma^2 I_T)$.

Fungsi produksi CD adalah sebagai berikut:

$$y = \beta_1 L^{\beta_2} K^{\beta_3}$$

sedangkan untuk fungsi CES dalam bentuk berikut:

$$y = \beta_1[\beta_2 L^{\beta_3} + (1 - \beta_2)K^{\beta_3}]^{\beta_4/\beta_3}$$

Sedangkan fungsi objektif untuk minimalisasi least square adalah:

$$S = [y - f(X, \beta)]'[y - f(X, \beta)]$$

dan fungsi objektif untuk memaksimalkan fungsi like lihood adalah:

$$L = -\frac{T}{2} \log 2\pi - \frac{T}{2} \log (y - x\beta)'(y - x\beta) / T - \frac{T}{2}$$

Untuk pemilihan model terbaik digunakan persamaan *Akaike Information Criteria* (AIC) dan *Schwart Criteria* (SC) untuk menentukan model yang paling sesuai atau efisien untuk masing-masing pendekatan. Perhitungan AIC dan SC untuk *maksimum likelihood* akan menggunakan rumus berikut ini:

$$AIC = -2 * \log \text{maximum likelihood} + 2(\#parameters)$$

$$SC = -2 * \log \text{maximum likelihood} + (\log(T))(\#parameters)$$

Sedangkan Perhitungan AIC dan SC untuk *Least Square* akan menggunakan rumus berikut ini:

$$AIC = \log(e'e/T) + 2(\#parameters)/T \quad \text{dimana } e's = S$$

$$SC = \log(e'e/T) + (T * \log(\#parameters))/T \quad \text{dimana } e's = S$$

Bab 4

Hasil Eksperimen dan Analisis

4.1 Fungsi Produksi Cobb- Dauglas

Data yang digunakan adalah sesuai dengan Tugas 2 yaitu data produksi suatu komoditi (y) dengan input yang digunakan adalah kapital (K) dan tenaga kerja (L) (data terlampir bersama program). Model statistik non-linier berbentuk: $y = f(x, \beta) + e$. Dengan data tersebut akan diestimasi parameter fungsi produksi Cobb Douglas (CD) dengan algoritma genetika. Fungsi produksi Cobb-Dauglas adalah sebagai berikut:

$$y = \beta_1 L^{\beta_2} K^{\beta_3}$$

Sedangkan fungsi objektif untuk minimalisasi least squre error adalah:

$$S = [y - f(X, \beta)]' [y - f(X, \beta)]$$

dan fungsi objektif untuk memaksimumkan fungsi like lihood adalah:

$$L = -\frac{T}{2} \log 2\pi - \frac{T}{2} \log (y - x\beta)' (y - x\beta) / T - \frac{T}{2}$$

Dalam ekperimen ini akan dilakukan 2 simulasi, yaitu, simulasi pertama adalah dengan menggunakan 10 ribu generasi dan dengan mutasi rate sebesar 0,03, sedangkan untuk simulasi kedua adalah dengan menggunakan 15 ribu generasi dan mutation rate sebesar 0,02.

4.1.1 Least Square Error

Hasil ekperimen untuk 2 simulasi dalam melakukan estimasi parameter β dengan fungsi objektif adalah meminimumkan *sum of square error* (*least square error*) dapat disajikan dalam tabel-tabel berikut ini:

Tabel 1. Hasil Estimasi Cobb-Dauglas dg Least Square

Simulasi 1:		Simulasi 2:	
generation_n = 10000;		generation_n = 15000;	
popuSize = 50;		popuSize = 50;	
xover_rate = 1.0;		xover_rate = 1.0;	
mutate_rate = 0.03;		mutate_rate = 0.02;	
bit_n = 40;		bit_n = 40;	
range = [0 2; 0 1; 0 1];		range = [0 2; 0 1; 0 1];	

	Simulasi 1	Simulasi 2
β_1	1.470948	1.468719
β_2	0.374022	0.375000
β_3	0.575195	0.575060
$S(\beta)$	0.561453	0.561468
AIC	3.7784	3.7784
SC	73.0560	73.0560

Simulasi pertama mulai konvergen pada generasi ke 9740 (output terlampir beberapa halaman saja), sedangkan simulasi ke dua konvergen mulai pada generasi yang ke 8149. Berdasarkan kriteria AIC dan SC maka hasil kedua simulasi tersebut menunjukkan nilai yang sama (mungkin digit ke lima dan seterusnya dibelakang koma akan menunjukkan nilai yang berbeda). Namun demikian, simulasi pertama mempunyai nilai sum of square $[S(\beta)]$ lebih rendah dari simulasi ke dua, maka dalam hal ini model terbaik adalah untuk simulasi pertama yaitu:

$$y = 1.470948 L^{0.374022} K^{0.575195} \quad (4.1)$$

4.1.2 Maksimum Likelihood

Sedangkan, untuk melakukan estimasi parameter β dengan fungsi objektifnya adalah *maksimum Likelihood*, hasil eksperimen-eksperimen disajikan dalam Tabel 2 sebagai berikut :

Tabel 2. Hasil Estimasi Cobb-Dauglas dg Maksimum Likelihood

Simulasi 1:		Simulasi 2:	
generation_n = 10000;		generation_n = 15000;	
popuSize = 50;		popuSize = 50;	
xover_rate = 1.0;		xover_rate = 1.0;	
mutate_rate = 0.03;		mutate_rate = 0.02;	
bit_n = 40;		bit_n = 40;	
range = [0 2; 0 1; 0 1];		range = [0 2; 0 1; 0 1];	

	Simulasi 1	Simulasi 2
β_1	1.500000	1.475439
β_2	0.368439	0.375000
β_3	0.569923	0.572569
L(β)	-13.915419	-13.914277
AIC	33.8308	33.8286
SC	38.0344	38.0321

Simulasi pertama mulai konvergen pada generasi ke 3611 (output terlampir beberapa halaman saja), sedangkan simulasi ke dua konvergen mulai pada generasi yang ke 4895. Berdasarkan kriteria AIC dan SC maka hasil kedua simulasi tersebut menunjukkan nilai yang sama (mungkin digit ke lima dan seterusnya dibelakang koma akan menunjukkan nilai yang berbeda). Namun demikian, simulasi kedua lebih kecil dari simulasi pertama, maka dalam hal ini model terbaik adalah untuk simulasi kedua yaitu:

$$y = 1.475439 L^{0.375} K^{0.572569} \quad (4.2)$$

4.2 Fungsi Produksi CES

Demikian pula, data yang digunakan adalah sesuai dengan Tugas 2 yaitu data produksi suatu komoditi (y) dengan input yang digunakan adalah kapital (K) dan tenaga kerja (L) (data terlampir bersama program). Model statistik non-linier berbentuk: $y = f(x, \beta) + e$. Dengan data tersebut akan diestimasi parameter fungsi produksi CES dengan algoritma genetika. Fungsi produksi CES adalah sebagai berikut:

$$y = \beta_1 [\beta_2 L^{\beta_3} + (1 - \beta_2) K^{\beta_3}]^{\beta_4 / \beta_3}$$

Sedangkan fungsi objektif untuk minimalisasi least square error adalah:

$$S = [y - f(X, \beta)]' [y - f(X, \beta)]$$

dan fungsi objektif untuk memaksimalkan fungsi likelihood adalah:

$$L = -\frac{T}{2} \log 2\pi - \frac{T}{2} \log (y - x\beta)' (y - x\beta) / T - \frac{T}{2}$$

Demikian pula, dalam eksperimen ini juga akan dilakukan 2 simulasi, yaitu, simulasi pertama adalah dengan menggunakan 10 ribu generasi dan dengan mutasi rate sebesar 0,03, sedangkan untuk simulasi kedua adalah dengan menggunakan 15 ribu generasi dan mutation rate sebesar 0,02.

4.2.1 Least Square Error

Hasil estimasi parameter β dengan fungsi objektifnya adalah meminimumkan *sum of square error (least square error)* diperoleh hasil yang disajikan dalam Tabel 3 yaitu sebagai berikut ini:

Tabel 3. Hasil Estimasi CES dg Least Square

Simulasi 1:	Simulasi 2:
generation_n = 10000;	generation_n = 15000;
popuSize = 50;	popuSize = 50;
xover_rate = 1.0;	xover_rate = 1.0;
mutate_rate = 0.03;	mutate_rate = 0.02;
bit_n = 40;	bit_n = 40;
range = [0 2; 0 1; 0 1];	range = [0 2; 0 1; 0 1];

	Simulasi 1	Simulasi 2
β_1	1.359375	1.375000
β_2	0.388781	0.388184
β_3	0.320313	0.299759
β_4	0.992099	0.985805
$S(\beta)$	0.524111	0.524182
AIC	3.7806	3.7804
SC	64.4943	64.4942

Simulasi pertama mulai konvergen pada generasi ke 3005 (output terlampir beberapa halaman saja), sedangkan simulasi ke dua konvergen mulai pada generasi yang ke 13625. Berdasarkan kriteria AIC dan SC maka hasil simulasi kedua lebih kecil dari simulasi pertama, maka dalam hal ini model terbaik adalah untuk simulasi kedua yaitu:

$$y = 1.375 [0.388184 L^{0.299759} + (1 - 0.388184) K^{0.299759}]^{0.985805 / 0.299759} \quad (4.3)$$

dari simulasi tersebut menunjukkan bahwa untuk menjangapai global optimum diperlukan generasi yang lebih panjang.

4.2.2 Maksimum Likelihood

Sedangkan untuk melakukan estimasi parameter β dengan proses optimisasi fungsi objektifnya yaitu *maksimum Likelihood*, maka hasil eksperimen disajikan dalam Tabel 4 dibawah ini:

Tabel 4. Hasil Estimasi CES dg Maksimum Likelihood

Simulasi 1:	Simulasi 2:
generation_n = 10000;	generation_n = 15000;
popuSize = 50;	popuSize = 50;
xover_rate = 1.0;	xover_rate = 1.0;
mutate_rate = 0.03;	mutate_rate = 0.02;
bit_n = 40;	bit_n = 40;
range = [0 2; 0 1; 0 1];	range = [0 2; 0 1; 0 1];

	Simulasi 1	Simulasi 2
β_1	1.375000	1.378906
β_2	0.389336	0.389162
β_3	0.306641	0.301819
β_4	0.985832	0.984269
$L(\beta)$	-13.879005	-13.879062
AIC	35.7580	35.7581
SC	41.3629	41.3629

Simulasi pertama mulai konvergen pada generasi ke 5182 (output terlampir beberapa halaman saja), sedangkan simulasi ke dua konvergen mulai pada generasi yang ke 13564. Berdasarkan kriteria AIC dan SC maka hasil simulasi pertama lebih kecil dari simulasi kedua, maka dalam hal ini model terbaik adalah untuk simulasi pertama yaitu:

$$y = 1.375 [0.389336 L^{0.306641} + (1 - 0.389336) K^{0.306641}]^{0.985832 / 0.306641} \quad (4.4)$$

4.3 Model Terbaik

Untuk memilih model terbaik pada umumnya digunakan kriteria AIC dan SC. Namun dalam hal ini perhitungan nilai AIC dan SC berdasarkan nilai Fitness dari operasi algoritma genetik yang mana mempunyai objective function yang berbeda antara least square error dengan maksimum likelihood. Oleh karena itu, menurut hemat kami walaupun dengan kriteria AIC dan SC, tidak dapat membandingkan antar metoda least square error dan maksimum likelihood. Sehingga yang dapat dilakukan adalah membandingkan fungsi produksi CD dan CES dengan metoda yang sama.

Dengan nilai fitness algoritma genetica berdasarkan fungsi tujuan least square error, maka bila kita perlu membandingkan persamaan [4.1] dengan persamaan [4.3] yaitu:

- Model CD pada persamaan [4.1] dengan AIC=3.7784 dan SC=73.0560 (simulasi 1 terbaik);

- Model CES pada persamaan [4.3] dengan AIC=3.7804 dan BC=64.4942 (simulasi 2 terbaik);
- Nilai AIC dan BC untuk persamaan [4.3] dan pada persamaan [4.1] tidak konsisten; untuk nilai AIC persamaan [4.1] lebih kecil, namun untuk nilai BC persamaan [4.3] yang lebih kecil.
- Berdasarkan perbandingan diatas maka dengan menggunakan least square error sebagai fungsi tujuan tidak dapat memutuskan model mana yang terbaik

Dengan nilai fitness algoritma genetika berdasarkan fungsi tujuan maksimum likelihood, maka bila kita perlu membandingkan persamaan [4.2] dengan persamaan [4.4] yaitu:

- Model CD pada persamaan [4.2] dengan AIC= 33.8286 dan SC=38.0321 (simulasi 2 terbaik);
- Model CES pada persamaan [4.4] dengan AIC=35.7580 dan BC=41.3629 (simulasi 1 terbaik);
- Nilai AIC dan BC untuk Model CD persamaan [4.2] lebih kecil dari pada Model CES persamaan [4.4];
- Oleh Karena model terbaik adalah Model CD persamaan [4.2], yaitu:

$$y = 1.475439 L^{0.375} K^{0.572569}$$

Berdasarkan nilai-nilai tersebut, terlihat bahwa fungsi CD memiliki nilai AIC dan SC yang lebih kecil, sehingga dapat disimpulkan bahwa, input data yang telah diberikan lebih sesuai dengan fungsi CD. Atau dengan perkataan lain, berdasarkan data yang ada, fungsi CD lebih efisien dengan data yang diberikan dibandingkan dengan fungsi CES. Bila dibandingkan antar metoda penaksiran algoritma genetika bahwa fitness function dengan menggunakan fungsi tujuan maksimum likelihood lebih baik dari pada menggunakan least square error.

Bab 5

Kesimpulan

1. Beberapa kesimpulan dapat diperoleh berdasarkan hasil perhitungan dan analisa untuk dua fungsi produksi sebagai berikut;
2. Dengan menggunakan algoritma genetika dan data tersedia untuk penaksiran parameter fungsi produksi CD dan CES baik yang dilakukan melalui fungsi tujuan least square dan maximum likelihood diperoleh empat persamaan terbaik:
 - Model CD dengan least square method : $y = 1.470948 L^{0.374022} K^{0.575195}$
 - Model CD dengan Maksimum Likelihood: $y = 1.475439 L^{0.375} K^{0.572569}$
 - Model CES dengan least square method : $y = 1.375 [0.388184 L^{0.299759} + (1 - 0.388184) K^{0.299759}]^{0.985805 / 0.299759}$
 - Model CES dengan Maksimum Likelihood: $y = 1.375 [0.389336 L^{0.306641} + (1 - 0.389336) K^{0.306641}]^{0.985832 / 0.306641}$
3. Berdasarkan kriteria AIC dan SC dari keempat model yang terbaik adalah : Model Cobb-Daouglas $y = 1.475439 L^{0.375} K^{0.572569}$
4. Metoda penaksiran algoritma genetika dengan fitness yang menggunakan fungsi tujuan maksimum likelihood lebih baik, dari pada fitness yang menggunakan fungsi tujuan least square error.
5. Pada umumnya diperlukan proses penciptaan generasi baru yang lebih banyak untuk mencapai global optimum.

Lampiran

Daftar Lampiran

1. Program dan Output MATLAB untuk Penaksiran Model CD dengan Least Square Method.
2. Program dan Output MATLAB untuk Penaksiran Model CD dengan Maximum Likelihood Method.
3. Program dan Output MATLAB untuk Penaksiran Model CES dengan Least Square Method.
4. Program dan Output MATLAB untuk Penaksiran Model CES dengan Maximum Likelihood Method.

Daftar Pustaka

- [1] Syamsuddin, M., (2006), Catatan Kuliah Ekonometrika 3
- [2] Zheng, Yingsong., Kiyooka, Sumio.(1999), "Genetic Algorithm Applications:Assignment #2 for Dr.Z.Dong".
- [3] Suyanto, (2005), Algoritma Genetika dalam MATLAB, Penerbit ANDI Yogyakarta
- [4] Edi P. Pambudi.,(2006), Catatan Asistensi Ekonometrik 3