PRACE NAUKOWE Akademii im. Jana Długosza w Częstochowie
Technika, Informatyka, Inżynieria Bezpieczeństwa 2013, t. I

Artur Gola
Akademia im. Jana Długosza
Al. Armii Krajowej 13/15, 42-200 Częstochowa, e-mail: a.gola@ajd.czest.pl

EA-MOSGWA – NARZĘDZIE DO WYZNACZANIA PRZYCZYNOWYCH GENÓW W BADANIACH GWAS

Streszczenie. Praca przedstawia aktualny stan rozwoju programu EA-MOSGWA. Jest to narzędzie służące do wyznaczania przyczynowych genów w badaniach asocjacyjnych całego genomu (ang. *Genome Wide Association Studies*, GWAS). Badania GWAS mają na celu określenie genów, które mogą być odpowiedzialne za różnego rodzaju choroby genetyczne (np. rak, cukrzyca), a także genów, które wpływają na daną cechę, np. wzrost lub wagę. Sprowadzają się one do przebadania wielu tysięcy polimorfizmów pojedynczego nukleotydu (ang. S*ingle Nucleotide Polymorphism*, SNP) i powiązaniu ich (pojedynczych lub grupy SNP-ów) z przypadkami klinicznymi oraz możliwymi do zmierzenia cechami. Bardzo ważne w tego typu badaniach jest określenie jak największej liczby przyczynowych SNPów (ang. *True Positive*) przy jednoczesnej minimalizacji liczby fałszywych SNP-ów (ang. *False Positive*), czyli takich, które w rzeczywistości nie są przyczynowymi, a program zaklasyfikował je jako przyczynowe. W pracy przedstawiono wyniki symulacji, które pokazują, że zaproponowany algorytm ma dobre właściwości dotyczące dwóch badanych parametrów statystycznych.

Słowa kluczowe: algorytm ewolucyjny, badanie asocjacyjne całego genomu, regresja liniowa.

Wprowadzenie

Praca przedstawia aktualny stan rozwoju programu EA-MOSGWA – narzędzia przeznaczonego do wyznaczania przyczynowych genów w badaniach asocjacyjnych całego genomu. Jest ono opracowywane w Zakładzie Informatyki Stosowanej Akademii im. Jana Długosza w Częstochowie, we współpracy z Politechniką Wrocławską i Uniwersytetem Medycznym w Wiedniu. Planowane jest również dołączenie tego narzędzia w formie modułu do większego, bardziej rozbudowanego programu MOSGWA, który od kilku lat jest rozwijany na Uniwersytecie Medycznym w Wiedniu i Politechnice Wrocławskiej. Program MOSGWA (skrót od Model Selection for Genome Wide Associations) jest za-

awansowanym narzędziem, które służy do wyznaczania przyczynowych genów w badaniach całego genomu. Posiada kilka zaimplementowanych metod wyboru modelu oraz daje możliwość dołączania nowych metod. Prezentowany w artykule program rozszerza możliwości programu MOSGWA o nową metodę wyboru modelu, która korzysta ze zmodyfikowanego algorytmu ewolucyjnego (ang. *Evolutionary Algorithm*) [4, 7], stąd nazwa programu EA-MOSGWA.

Program MOSGWA ma również zaimplementowanych wiele funkcji, które są niezbędne przy implementacji nowych metod służących do identyfikacji przyczynowych genów. Między innymi należą do nich: odczyt plików z danymi wejściowymi, wykonanie testów na pojedynczych markerach (ang. *Single Marker Test*) i wyznaczenie p-wartości, ocena modelu według jednego z dostępnych kryteriów, obliczenie korelacji pomiędzy markerami, zapis wyników w formacie Matlaba lub R. Wszystkie te funkcje zostały wykorzystane w programie EA-MOSGWA.

Przegląd dostępnych metod

Na obecnym stanie rozwoju EA-MOSGWA umożliwia wyznaczenie grupy SNP-ów przyczynowych, korzystając z regresji liniowej oraz kryterium Bayesa (ang. *Bayesian Information Criterion*, w skrócie *BIC*) i jego modyfikacji mBIC2 [2, 3] do oceny otrzymywanych rozwiązań.

Obecnie można wyróżnić kilka głównych nurtów w badaniach asocjacyjnych całego genomu: z użyciem metod opartych na kryterium Bayesa [2, 3, 6], z użyciem metody Monte Carlo z wykorzystaniem łańcuchów Markowa (ang. *Markov Chain Monte Carlo*, MCMC) [5] oraz z metody lasso połączonej z walidacją krzyżową [8]. Przegląd najnowszych metod można znaleźć w pracy [1].

Podstawy teoretyczne

Program EA-MOSGWA jest przeznaczony do identyfikacji SNP-ów sprzężonych z zadaną cechą ilościową. Oznaczając miarę cechy osobnika i jako y_i, gdzie $i \in \{1, \ldots, n\}$, oraz przyjmując kodowanie wartości genotypu dla p SNP-ów, jako $x_{ij} \in \{-1, 0, 1\}$, $j \in \{1, \ldots, p\}$, możemy opisać model regresji liniowej równaniem:

$$y_i = \mu + \sum_{j \in M} \beta_j x_{ij} + \epsilon_i \tag{1}$$

gdzie μ oznacza miejsce przecięcia prostej z osią Y, M oznacza model – podzbiór markerów (SNP-ów), które mają wpływ na badaną cechę, natomiast składnik ε_i oznacza błąd o rozkładzie normalnym $N(0, \sigma^2)$.

Wybór najlepszego modelu regresji liniowej i w efekcie lokalizacja istotnych genów wymaga zastosowania kryterium, na podstawie którego modele są oceniane. W programie EA-MOSGWA zastosowano zmodyfikowane Bayesowskie kryterium informacyjne (ang. *modified Bayesian Information Criterion,* mBIC2) [3]:

$$mBIC2 = n \log RSS + k \log n + 2k \log(p/4) - 2 \log(k! \quad (2)$$

gdzie k oznacza rozmiar ocenianego modelu, n liczbę osobników, p liczbę SNPów.

Z informatycznego punktu widzenia znalezienie rozwiązania sprowadza się do sprawdzenia wszystkich podzbiorów zbioru p-elementowego, co oznacza złożoność wykładniczą. Ponieważ w rzeczywistych przypadkach liczba SNPów może osiągać wartość rzędu 1 000 000, metoda brutalna nie zdaje egzaminu.

W celu wytypowania przyczynowych SNP-ów program EA-MOSGWA wykorzystuje losowy algorytm przeszukiwania przestrzeni rozwiązań, który jest oparty na algorytmie ewolucyjnym oraz korzysta z mBIC2 do oceny rozwiązań pośrednich i ukierunkowania przeszukiwań w rejony przestrzeni rozwiązań, które są najbardziej obiecujące.

Implementacja

Idea działania algorytmów ewolucyjnych jest zaczerpnięta z obserwacji rozwoju gatunków w środowisku naturalnym, gdzie następne pokolenie jest lepiej przystosowane do warunków środowiska, w jakich przebywa. W przypadku programu komputerowego zasada ta sprowadza się do tworzenia lepszych rozwiązań, przy wykorzystaniu w tym celu rozwiązań już istniejących [4, 7]. Zmodyfikowany algorytm ewolucyjny zastosowany w EA-MOSGWA pracuje na populacji osobników, której rozmiar jest stały i wynosi n. Osobnik jest odpowiednikiem modelu i zarazem stanowi potencjalne rozwiązanie zadania.

Algorytm użyty w programie EA-MOSGWA bazuje na algorytmie memetycznym (ang. *Memetic Algorithms*, MA) przedstawionym w pracy [2]. Został on zaimplementowany w programie Matlab i przeznaczony jest do pracy na populacjach eksperymentalnych, gdzie bada się dużo mniej markerów, do 300. Najważniejsze różnice pomiędzy tymi algorytmami zostały opisane w paragrafach: Tworzenie populacji początkowej; Mutacja; Lokalne ulepszanie modelu; Rekombinacja.

Zasada działania algorytmu jest przedstawiona na rysunku 1. Po uruchomieniu algorytm tworzy populację początkową. Następnie w każdej iteracji tworzone są nowe osobniki. Wybór osobników do procesu reprodukcji następu-

je w metodzie selekcji. Z dużym prawdopodobieństwem p_c w procesie rekombinacji jest generowany nowy osobnik. Jeżeli nie dochodzi do rekombinacji, to do dalszego etapu przechodzi lepiej oceniany osobnik. Z małym prawdopodobieństwem p_m nowy osobnik ulega mutacji. W końcowym etapie potomek ten poddawany jest procesowi lokalnego ulepszenia, a następnie jest oceniany i porównywany z najgorzej ocenianym osobnikiem w populacji. Jeżeli nowy osobnik jest lepiej oceniany niż najgorszy osobnik w populacji, to go zastępuje. Algorytm kończy pracę, jeżeli w ustalonej liczbie *maxNoProgressIterations* iteracji nie nastąpi utworzenie lepszego modelu niż *B* najlepszych modeli w populacji.

W przeciwieństwie do tradycyjnych algorytmów ewolucyjnych prezentowany algorytm nie tworzy odrębnej populacji zawierającej osobniki powstałe w wyniku operacji genetycznych, ale na bieżąco zastępuje najgorsze osobniki nowo powstałymi, lepszymi osobnikami.

Reprezentacja rozwiązań

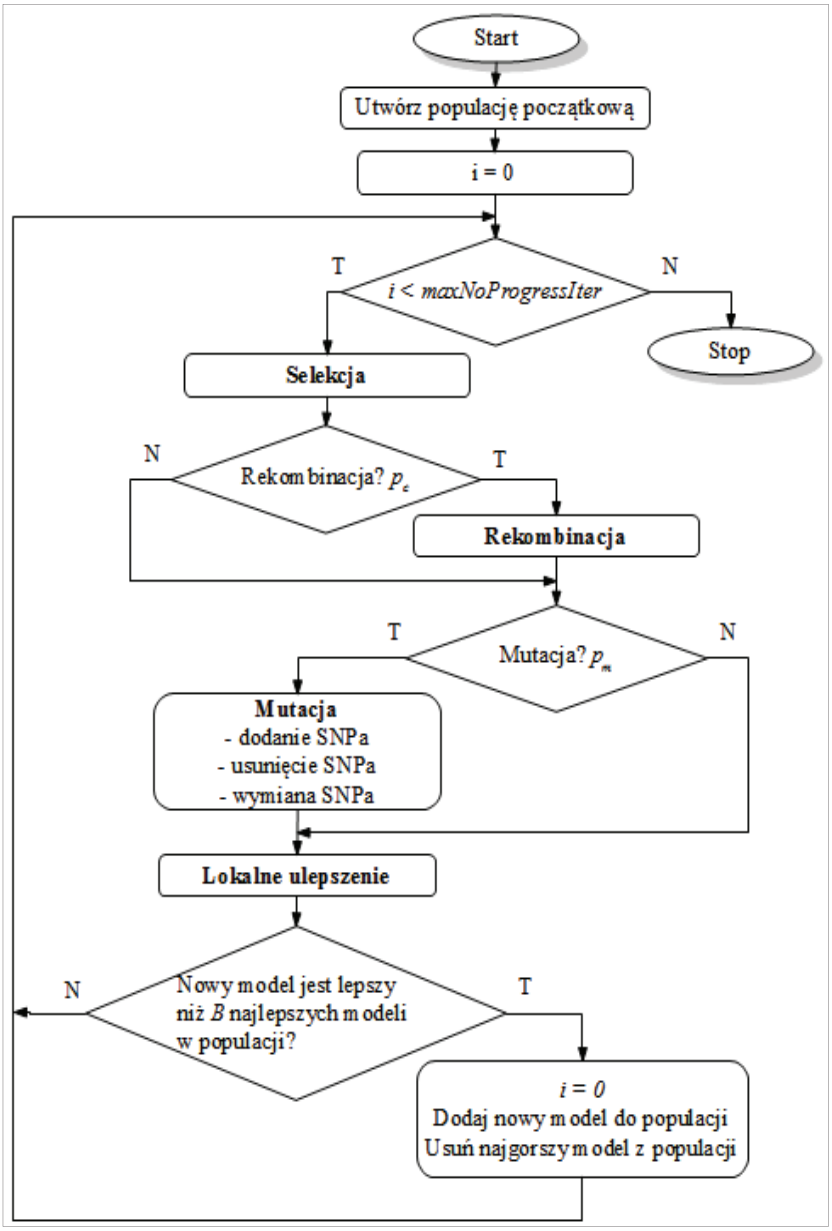
W algorytmach ewolucyjnych stosuje się kodowanie rozwiązań, które jest najlepiej dostosowanie do danego problemu. W naszym przypadku rozwiązaniem jest zbiór numerów SNP-ów, które program zaklasyfikuje jako przyczynowe.

Tworzenie populacji początkowej

Pierwszą operacją wykonywaną przez algorytm ewolucyjny jest utworzenie populacji początkowej. Najprostszą metodą jest utworzenie rozwiązań w sposób losowy. W kolejnych krokach potencjalne rozwiązania z populacji początkowej są ulepszanie, aż algorytm zbiegnie się do rozwiązania końcowego. Pracę algorytmu można przyspieszyć, stosując metody, które utworzą populację początkową z osobników lepszych niż te wygenerowane w sposób losowy.

W proponowanym narzędziu do utworzenia populacji początkowej stosowana jest zachłanna metoda *ForwardSelection*, która jest zaimplementowana w programie MOSGWA. Wymaga ona wcześniejszego obliczenia p-wartości z klasycznych testów Studenta (obliczenie te są wykonywane przez funkcje *SingleMarkerTest* zaimplementowaną w programie MOSGWA) i posortowania wszystkich SNP-ów rosnąco względem p-wartości. Metoda *ForwardSelection* dodaje kolejno do modelu takie markery, które minimalizują funkcję BIC:

$$BIC = n \log RSS + k \log n \tag{3}$$



Rys. 1. Schemat blokowy algorytmu zastosowanego w programie EA-MOSGWA

Zaczyna od SNP-ów, które mają najmniejsze p-wartości. Rozmiar modeli tworzonych w populacji początkowej został ograniczony do 150 SNP-ów. Dodatkowo, jeżeli jakiś marker został dodany do jednego modelu, to nie jest dodawany do innego. Warunek ten gwarantuje duża różnorodność markerów w populacji początkowej.

Metoda ta całkowicie różni się od tej zastosowanej w algorytmie memetycznym [2]. W przypadku MA osobniki są wybierane losowo, ale w oparciu o różne prawdopodobieństwa markerów dla poszczególnych chromosomów. W tym celu tworzony jest model, na postawie którego obliczane są te prawdopodobieństwa.

Metoda selekcji

Jako metoda selekcji została zastosowana klasyczna metoda selekcji turniejowej z liczbą uczestników równą 2. Oznacza to, że w sposób losowy, z takim samym prawdopodobieństwem, są wybierane dwa osobniki z populacji. Zwycięzcą turnieju jest model o lepszej ocenie, czyli z mniejszą wartością mBIC2. Przy pomocy metody turniejowej wybierane są dwa różne osobniki – rodzice. Z nich w procesie rekombinacji powstaje nowe rozwiązanie.

Rekombinacja

Proces rekombinacji przebiega w dwóch etapach. Oznaczmy przez S_1 i S_2 zbiory markerów należących do obu rodziców, przez $S_I = S_1 \cap S_2$ przecięcie zbiorów, przez $S_U = S_1 \cup S_2$ sumę zbiorów, natomiast przez $S_D = S_U / S_I$ różnicę symetryczną zbiorów. Na pierwszym etapie tworzony jest model wejściowy ze SNPów ze zbioru S_I. Następnie tworzone są i oceniane wszystkie możliwe modele powstałe w wyniku dodania do modelu wejściowego jednego SNP-a ze zbioru S_D, za każdym razem innego. Na przykład niech $S_I = \{S_1, S_2\}$, $S_D = \{S_3, S_4, S_5\}$. W kroku pierwszym zostaną utworzone następujące modele: $\{S_1, S_2, S_3\}$, $\{S_1, S_2, S_4\}$, $\{S_1, S_2, S_5\}$. Jeżeli najlepszy z tych modeli jest lepiej oceniany od modelu wejściowego, to zastępuje model wejściowy, a ze zbioru S_D usuwany jest SNP, który spowodował ulepszenie modelu wejściowego. Operacja ta jest powtarzana dopóki następuje ulepszenie modelu wejściowego i zbiór S_D nie jest pusty. W ten sposób powstaje pierwszy model.

Drugi model wejściowy jest utworzony ze SNPów ze zbioru S_U, z którego pojedynczo usuwane są SNPy należące do zbioru S_D. W ten sposób sprawdzane są wszystkie modele o rozmiarze o jeden mniejszym od modelu wejściowego. Na przykład niech $S_I = \{S_1, S_2\}$, $S_D = \{S_3, S_4, S_5\}$, $S_U = \{S_1, S_2, S_3, S_4, S_5\}$. W kroku pierwszym zostaną utworzone następujące modele: $\{S_1, S_2, S_4, S_5\}$, $\{S_1, S_2, S_3, S_5\}$, $\{S_1, S_2, S_3, S_4\}$. Jeżeli najlepszy z tak utworzonych modeli jest lepszy od modelu wejściowego, to go zastępuje, a ze zbioru S_D usuwany jest SNP, który spowodował ulepszenie modelu wejściowego. Cała operacja jest

powtarzana dopóki następuje poprawa lub sprawdzono wszystkie SNP-y i w modelu wejściowym pozostały tylko SNP-y ze zbioru S_I.

W procesie rekombinacji lepszy z dwóch modeli, które powstały podczas pierwszego i drugiego etapu, jest zwracany jako wynik rekombinacji.

Algorytm memetyczny [2] wyznacza model wynikowy wyłącznie na podstawie etapu pierwszego.

Mutacja

Mutacja może przebiegać na dwa sposoby, które zachodzą z takim samym prawdopodobieństwem 0,5: nowy marker może być dodany do modelu lub jeden z markerów może być usunięty z modelu. W przypadku operacji dodania, nowy marker jest wybierany w sposób losowy. Jeżeli jednak jest on skorelowany powyżej wartości 0,5 z jednym ze SNP-ów, które są w modelu, to operacja wyboru nowego markera jest powtarzana. Jeżeli w 1000 próbach nie uda się znaleźć markera skorelowanego poniżej wartości 0,5 z każdym ze SNP-ów z modelu, to metoda mutacji kończy działanie.

W przypadku usuwania jeden losowo wybrany marker jest usuwany z modelu. Jeżeli w modelu jest tylko jeden marker, to jest on zastępowany innym, losowo wybranym markerem.

W stosunku do algorytmu MA[2] klasyczna odległość między genami została zastąpiona współczynnikiem korelacji. Jest to spowodowane faktem, że w populacjach ludzkich nie ma bezpośredniej zależności między odległością a korelacją i często dobry substytut dla aktualnie badanego genu znajduje się w znacznej od niego odległości.

Lokalne ulepszanie modelu

Metoda lokalnego ulepszenia robi użytek ze skorelowanej struktury markerów. SNP-y skorelowane ze sobą są związane z tą samą cechą. Jeżeli w modelu znalazł się SNP przyczynowy, to wśród SNP-ów, które są z nim skorelowane, można szukać głównego reprezentanta grupy skorelowanych SNP-ów, dla którego kryterium mBIC2 przyjmuje wartość minimalną. W metodzie tej staramy się ulepszyć model poprzez zastąpienie danego markera kolejno markerami z nim skorelowanymi, podczas gdy pozostałe markery zostają bez zmian. Po znalezieniu lokalnego minimum dla pierwszego SNP-a z ulepszanego modelu, powtarzamy operację szukania dla kolejnego SNP-a. Prezentowane narzędzie rozpatruje SNP-y, które są w najbliższym sąsiedztwie rozpatrywanego SNP-a (w oknie o rozmiarze 50 najbliższych SNP-ów) i są z nim skorelowane powyżej 0,5.

Główna różnica pomiędzy MA i EA polega na tym, że EA-MOSGWA bierze pod uwagę tylko markery skorelowane powyżej 0,5 z ulepszanym marke-

rem i które znajdują się ograniczonej odległości – oknie, natomiast MA nie uwzględnia korelacji i bierze pod uwagę kolejne markery dopóki ich użycie powoduje poprawę ulepszanego modelu.

Wyniki eksperymentalne

Symulacje komputerowe z użyciem programu EA-MOSGWA zostały wykonane na zbiorze danych liczącym 23171 SNP-ów pochodzących z 4077 osobników.

Model przyczynowy

Zostało wybranych 15 SNP-ów ze zbioru danych wejściowych. Odległość pomiędzy dwoma kolejnymi SNP-ami w chromosomie wynosiła około 1600 markerów, co gwarantowało brak korelacji pomiędzy SNPami przyczynowymi. Wartość β dla pierwszego SNP-a została ustalona na poziomie 0,05, dla kolejnych była zwiększana o 0,01. Model przyczynowy przedstawiony jest w tabeli 1. Dla tak przygotowanego modelu zostały wyliczone wartości y_i. Przygotowałem 100 zestawów Y, w których wartości y_i zostały zaburzone losowymi wartościami wprowadzonymi przez składnik ε_i.

Tab. 1: Model przyczynowy

Nr	SNPId	Chr	Pos	beta	p-Value Single Marker Test
11	SNP_A-2131632	6	178750	0,05	0
1598	SNP_A-1878303	6	7933820	0,06	0
3193	SNP_A-2011614	6	18208311	0,07	0
4785	SNP_A-2202251	6	29364399	0,08	0
6382	SNP_A-2274313	6	38877662	0,09	0
7979	SNP_A-1922555	6	50397222	0,1	0
9575	SNP_A-2036796	6	67343887	0,11	0
11169	SNP_A-2255660	6	80036419	0,12	0
12769	SNP_A-4299403	6	91988235	0,13	0
14363	SNP_A-1987116	6	106144553	0,14	0
15962	SNP_A-2198701	6	119295855	0,15	0
17557	SNP_A-1819906	6	132321533	0,16	1.2586e-21
19146	SNP_A-4212041	6	145770626	0,17	0
20748	SNP_A-1841537	6	156216697	0,18	0
22343	SNP_A-4212125	6	165741341	0,19	0

Parametry algorytmu ewolucyjnego

Algorytm został uruchomiony z następującymi parametrami:
modelsNo = 10 – rozmiar populacji;
maxNoProgressIter = 1000 – kryterium stopu. Jeżeli nie nastąpi poprawa wśród *B* najlepszych osobników w populacji w ciągu 1000 iteracji;
B = 10 liczba określająca najlepsze osobniki, które są brane pod uwagę w kryterium stopu;
pCross = 0,95 – prawdopodobieństwo rekombinacji
pMutation = 0,25 – prawdopodobieństwo mutacji
tournamentSize = 2 – rozmiar turnieju

Scenariusze symulacji

Symulacje zostały przeprowadzone według dwóch scenariuszy. Pierwszy scenariusz miał za zadanie sprawdzenie, czy zastosowany algorytm zbiega się do najlepszego rozwiązania oraz dla jakich wartości β potrafi znaleźć SNP-y przyczynowe. SNP-y o mniejszych wartościach β są trudniejsze do zlokalizowania. Zostało wykonanych 100 symulacje na tym samym zestawie fenotypów (y_i).

Drugi scenariusz miał za zadanie sprawdzić, czy program będzie działał skutecznie w sytuacji, gdzie dane wejściowe są zaburzone. Zostało wykonanych 100 symulacji, każda na innym zestawie fenotypów, które nieznacznie różniły się między sobą wartością ε_i.

Wyniki

Najważniejszymi wynikami algorytmu są: liczba poprawnie wykrytych SNP-ów przyczynowych nazywana mocą (ang. *Power*), współczynnik fałszywych odkryć (ang. *False Discovery Rate*, w skrócie FDR) oraz liczba fałszywych odkryć (FD). Moc wyraża się wzorem:

$$POWER = TP/M_C \tag{4}$$

gdzie TP – liczba wykrytych przyczynowych SNP-ów, M_C – liczba SNP-ów w modelu przyczynowym.

Współczynnik fałszywych odkryć oblicza się ze wzoru:

$$FDR = FP/(TP + FP) \tag{5}$$

gdzie FP – liczba fałszywych odkryć. Liczba fałszywych odkryć jest równa FP.

SNP jest traktowany jako True Positive jeśli program zaklasyfikował dokładnie SNP-a przyczynowego lub SNP-a, który jest z nim skorelowany powy-

żej 0,5. Jeżeli ze SNPem przyczynowym jest skorelowanych więcej niż jeden SNP, to jeden z nich jest traktowany jako True Positive, natomiast pozostałe są traktowane jako False Positive.

Tab. 2: Wyniki EA-MOSGWA dla pierwszego i drugiego scenariusza

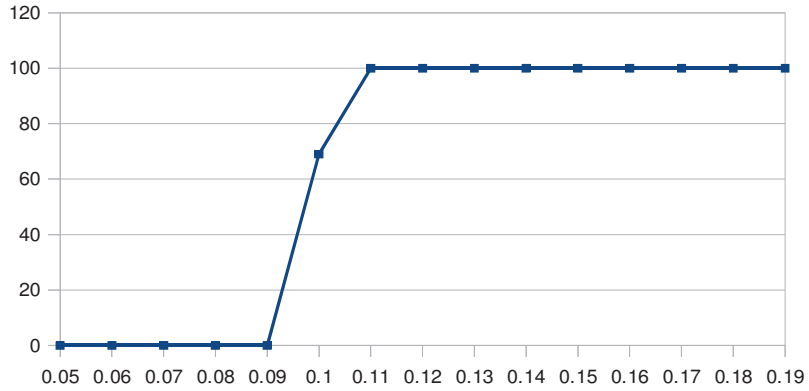
	Scenariusz pierwszy			**Scenariusz drugi**		
	POWER	**FDR**	**Liczba fałszywych SNP-ów**	**POWER**	**FDR**	**Liczba fałszywych SNP-ów**
Średnia	0,6667	0,0891	0,9800	0,6246	0,0434	0,4600
Maksimum	0,6667	0,0909	1,0000	0,8000	0,2222	3,0000
Minimum	0,6667	0,000	0,0000	0,4666	0,0000	0,0000
Odch. stand.	0,000	0,0128	0,1407	0,0818	0,0657	0,7166

Tab. 3: Częstość wykrywania przyczynowych SNP-ów w obu scenariuszach

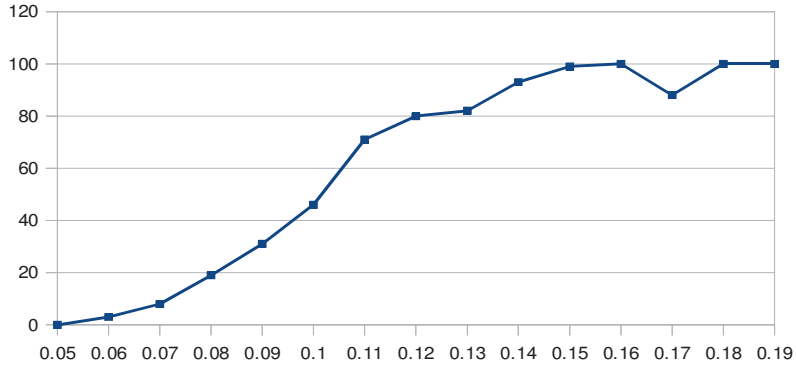
β	**scenariusz I**	**scenariusz II**
0.05	0	0
0.06	0	3
0.07	0	8
0.08	0	19
0.09	0	31
0.1	69	46
0.11	100	71
0.12	100	80
0.13	100	82
0.14	100	93
0.15	100	99
0.16	100	100
0.17	100	88
0.18	100	100
0.19	100	100

Tabela 2 przedstawia wyniki dla obu scenariuszy. Zawiera średnią, wartość maksymalną, minimalną oraz odchylenie standardowe. Natomiast tabela 3

przedstawia liczbę prawdziwych odkryć dla obu scenariuszy. Wykres mocy dla scenariusz pierwszego został przedstawiony na rysunku 2, a dla scenariusza drugiego na rysunku 3.



Rys. 2. Wykres mocy (POWER) dla pierwszego scenariusza



Rys. 3. Wykres mocy (POWER) dla drugiego scenariusza

Wyniki pierwszego scenariusza potwierdziły zdolność algorytmu do zbiegania się do optymalnego rozwiązania. EA-MOSGWA potrafi poprawnie wykrywać SNP-y przyczynowe, które mają współczynniki β na poziomie 0,1. Program generuje stabilne wyniki. Dla niewielkiej liczby symulacji algorytm

nie zaklasyfikował żadnego fałszywego SNP, o czym świadczy małe odchylenie standardowe. Natomiast w większości przypadków liczba zaklasyfikowanych fałszywych SNP-ów wynosiła 1.

W przypadku drugiej symulacji wyniki są nieznacznie gorsze, moc zmniejszyła się o ok. 4%. Zwiększyła się natomiast liczba fałszywych odkryć – do 3 SNP-ów. Przyczyną są wprowadzone zróżnicowane zaburzenia wartości y_i. Jednak najlepszy wynik jest na poziomie 80% poprawnych odkryć. W obu przypadkach modele wyznaczone przez EA-MOSGWA były lepiej oceniane (mniejsze wartości mBIC2) według zmodyfikowanego kryterium Bayesa niż przyczynowy model. Dla modelu przyczynowego mBIC2 wynosił 34109,37, natomiast proponowany algorytm w pierwszym scenariuszu osiągnął średnią wartość mBIC2: 34090,91. Tabela 4 przedstawia wartości mBIC2 dla drugiego scenariusza. Oznacza to znaczny wpływ składnika ε_i występującego w równaniu (1), który wprowadza dodatkowe zaburzenia do danych wejściowych na wielkość mBIC2.

Tab. 4: Porównanie mBIC2 modelu przyczynowego i wyznaczonego przez EA-MOSGWA

	EA-MOSGWA	**Model przyczynowy**
Średnia	34119,695	34170,054
Maksimum	34323,700	34383,200
Minimum	33924,000	33986,300
Odchylenie standardowe	83,150	85,450

Podsumowanie

Otrzymane wyniki potwierdzają prawidłowość zastosowanych rozwiązań oraz użyteczność prezentowanej metody w badaniach asocjacyjnych całego genomu. O ile EA-MOSGWA potrafi wykrywać markery przyczynowe o niewielkich wartościach β, to na ich podstawie nie można oszacować stopnia niepewności, który charakteryzuje każdą decyzję statystyczną. Jest to bardzo ważne, ponieważ pracując z danymi rzeczywistymi, nie znamy SNPów przyczynowych a powinniśmy określić, w jakim stopniu ufamy otrzymanym wynikom. Jest to jednak tematem dalszych prac nad algorytmem.

Podziękowania

Pragnę gorąco podziękować dr hab. Małgorzacie Bogdan oraz dr. hab. Florianowi Fromletowi za przekazaną wiedzę, bez której ten program by nie powstał, za pomoc przy rozwiązywaniu problemów pojawiających się w trakcie

opracowywania nowej metody oraz wszelkie wskazówki i uwagi, które wpłynęły na ulepszenie metod zaimplementowanych w programie EA-MOSGWA. Chciałbym również podziękować dr. Bernhardowi Bodenstorfer i mgr. Erichowi Dolejsi za pomoc związaną z wykorzystaniem funkcji programu MOSGA.

Symulacje komputerowe zostały wykonane w pracowni naukowej IMI, utworzonej w ramach projektu badawczego DS/IMI/106/2012.

Literatura

[1] Begum F., Ghosh D., Tseng G.C., and Feingold E., Comprehensive literature review and statistical considerations for GWAS meta-analysis, Nucleic Acids Res., 2012 May; 40(9): 3777–3784.

[2] Frommlet, F., Ljubic, I., Arnardottir, H. and Bogdan, M., QTL Mapping Using a Memetic Algorithm with modications of BIC as fitness function, Statistical Applications in Genetics and Molecular Biology 11 (4) Article 2, 2012.

[3] Frommlet F., Ruhaltinger F., Twarog P. and Bogdan M., Modied versions of Bayesian Information Criterion for genome-wide association studies. CSDA, 56 1038–1051, 2012.

[4] Goldberg D., Algorytmy genetyczne i ich zastosowania, WNT, Warszawa, 2003.

[5] Guan, Y. and Stephens, M., Bayesian Variable Selection Regression for Genome-wide Association Studies, and other Large-Scale Problems, Ann. Appl. Stat. 5 1780–1815, 2011.

[6] He Q. and Lin, D., A variable selection method for genome-wide association studies, Bioinformatics 27 1–8, 2011.

[7] Michalewicz Zb. Algorytmy Genetyczne + Struktury Danych = Programy Ewolucyjne, WNT, Warszawa, 2004.

[8] Wu, T.T., Chen, Y.F., Hastie, T., Sobel, E. and Lange, K. (2011). Genome-wide association analysis by lasso penalized logistic regression. Bioinformatics 25 714–721.

Artur Gola
Akademia im. Jana Długosza w Częstochowie

EA-MOSGWA – A TOOL FOR IDENTIFYING CAUSAL GENES IN GENOME WIDE ASSOCIATION STUDIES

Abstract

This paper presents the current stage of the development of EA-MOSGWA – a tool for identifying causal genes in Genome Wide Association Studies (GWAS). The main goal of GWAS is to identify genes which are causal for a particular disease and also genes which may be responsible for a given trait, e.g eyes color. The studies conduct to examine hundred of thousand Single Nucleotide Polymorphisms (SNP) and assign them to clinical cases or the measurable traits.
Very important in this kind of research is to identify as many causal SNP as possible while minimizing the number of false SNPs. A false positive SNP is a SNP which in fact is not causal and the program has classified him as a causal. I present the results of the simulation study, which show that the proposed algorithm has good properties with respect to these two statistical parameters. I present the results of the simulation study, which show that the proposed algorithm has good properties with respect to these two statistical parameters.

Keywords: Evolutionary Algorithm, Genome Wide Association, linear regression.