Politechnika Poznańska
Wydział Informatyki i Zarządzania
Instytut Informatyki

Praca dyplomowa inżynierska

SYSTEM WSPOMAGANIA DECYZJI DIAGNOSTYCZNYCH W OCENIE BADANIA IMMUNOFENOTYPOWEGO ROZROSTÓW NOWOTWOROWYCH WYWODZĄCYCH SIĘ Z KOMÓREK UKŁADU KRWIOTWÓRCZEGO

Maria Marchwicka, 66282
Marek Lankauf, 71329
Michał Nowak, 71348

Promotor
dr inż. Szymon Wilk

Poznań, 2008 r.

Dziękujemy wszystkim Pracownikom Katedry i Zakładu Immunologii Klinicznej Uniwersytetu Medycznego w Poznaniu, a w szczególności Panom Prof. Dr. hab. n. med. Janowi Żeromskiemu oraz Dr. hab. n. med. Grzegorzowi Dworackiemu za umożliwienie realizacji projektu. Dziękujemy również Panu Adamowi Ambrożemu z firmy Kamsoft za pomoc w integracji systemu z działającą w Katedrze bazą danych.

Spis treści

Rozdział 1

Wstęp

1.1 Informatyka w medycynie

Informatyka znajduje coraz większe zastosowanie w dzisiejszej medycynie. Związane jest to z funkcjonowaniem jednostek medycznych jako podmiotów gospodarczych, wspomaganiem procesów diagnostyki i leczenia pacjentów, komunikacją, obsługą nowoczesnej aparatury a także prowadzeniem badań naukowych, pracą z dużą ilością danych, ustalaniem i weryfikacją standardów postępowania.

Postęp w wielu działach medycyny jest bezpośrednio związany z możliwościami współczesnej techniki. W szczególny sposób dotyczy to takich gałęzi rozwoju jak immunologia, genetyka czy nowoczesna diagnostyka obrazowa i molekularna. Coraz szersze stosowanie w procesie diagnostyki i leczenia skomputeryzowanych urządzeń pomiarowych wiąże się z koniecznością pracy z dużymi ilościami danych w formie cyfrowej. Analiza wyników niektórych badań jest trudna lub niemożliwa bez wsparcia ze strony odpowiednich narzędzi wspomagania decyzji.

1.2 Cel pracy

Celem pracy jest zaprojektowanie i wykonanie systemu wspomagania decyzji diagnostycznych.Program powinien ułatwiać ocenę rozrostów nowotworowych na podstawie wyników badania immunofenotypowego komórek pochodzących z krwi obwodowej lub szpiku metodą cytometrii przepływowej. Metoda ta pozwala ocenić obecność poszczególnych antygenów (markerów) na powierzchni leukocytów (immunofenotyp) a także określić ich wielkości i ilość ziarnistości w cytoplazmie. Parametry te charakteryzują populacje komórek i pozwalają na dostrzeżenie potencjalnych nieprawidłowości. Odchylenie od normy w zakresie badanych parametrów może świadczyć o procesie nowotworowym. Charakter i zakres zmian stwierdzonych w badaniu jest podstawą do postawienia konkretnego rozpoznania (podjęcia decyzji diagnostycznej). Aplikacja ma zostać wdrożona w Katedrze i Zakładzie Immunologii Klinicznej Uniwersytetu Medycznego w Poznaniu.

Oprogramowanie powinno umożliwić odczyt danych z plików w formacie *FCS* (ang. *Flow Cytometry Standard*), który jest standardowym formatem zapisu wyników badania immunofenotypowego, ich szybką analizę oraz wydruk uzyskanych automatycznie lub wprowadzonych przez lekarza wartości i wyników analizy. Aplikacja powinna wykorzystywać metody analizy skupień do wyróżnienia w badanej populacji komórek poszczególnych subpopulacji: neutrofili, limfocytów i monocytów. Program powinien dla znalezionych grup komórek określić obecność wybranych antygenów i zapisać je do formularza wyników. Użytkownik powinien mieć możliwość edycji danych obliczonych przez program. System ma także ułatwić dostęp do lokalnej bazy danych w celu zapisu oraz odczytu wyników przeprowadzonych badań.

Program powinien mieć budowę modułową, a poszczególne jego części powinny działać niezależnie. Zapewni to możliwość łatwego przystosowania całej aplikacji do potrzeb innych laboratoriów o podobnym profilu badań. Pozwoli także na wykorzystanie modułów realizujących poszczególne funkcje (odczyt danych z bazy, odczyt plików *FCS* i ich eksport do formatu *CSV*, wizualizacja danych, generowanie raportów z wynikami badań) do realizacji zadań nie związanych bezpośrednio z diagnostyką chorób nowotworowych metodą cytometrii przepływowej.

1.3 Zakres pracy

W rozdziale drugim omówiono medyczne podstawy analizowanego problemu (Maria Marchwicka, Michał Nowak). W rozdziale trzecim przedstawiono metody analizy skupień (Marek Lankauf). Rozdział czwarty zawiera przegląd wykorzystywanych technologii (Maria Marchwicka, Marek Lankauf, Michał Nowak). Piąty rozdział to przedstawienie architektury systemu oraz zaprojektowanych klas (Maria Marchwicka, Marek Lankauf, Michał Nowak). W rozdziale szóstym omówiono działanie aplikacji (Marek Lankauf, Michał Nowak). Ostatni rozdział to podsumowanie pracy, wnioski oraz omówienie perspektyw rozwoju aplikacji (Maria Marchwicka).

1.4 Podział zadań

Maria Marchwicka Kierowanie projektem. Specyfikacja wymagań. Konsultacja z klientem. Konsultacja zagadnień medycznych związanych z projektem. Komunikacja z bazą danych. Testy porównawcze.

Marek Lankauf Przygotowanie danych do analizy. Moduł do analizy skupień. Rozpoznawanie subpopulacji. Moduł do generowania dokumentów *RTF*. Interfejs użytkownika.

Michał Nowak Odczyt danych z pliku *FCS*. Odfiltrowywanie artefaktów z populacji badanych zdarzeń. Przydział komórek do subpopulacji. Określanie charakterystyki immunofenotypowej subpopulacji komórkowych. Wizualizacja danych. Aplikacja pomocnicza do usuwania danych osobowych z plików *FCS*.

Rozdział 2

Cytometria przepływowa w diagnostyce nowotworów

2.1 Ocena immunofenotypu komórki

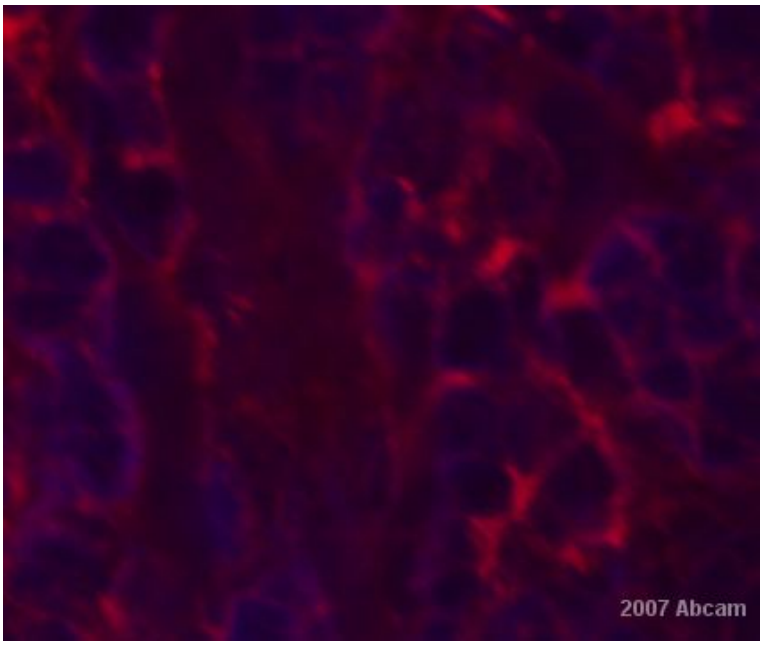
Różne komórki układu krwiotwórczego posiadają na swojej powierzchni charakterystyczny dla swojego rodzaju układ znaczników (markerów).[1] Są to antygeny różnicowania określane według klasyfikacji *CD* (ang. *Cluster Determinants*). *CD* to antygen, który w sposób swoisty wykrywany jest przy użyciu odpowiednich przeciwciał monoklonalnych. Odbywa się to na zasadzie reakcji antygen - przeciwciało, w której przeciwciało przeciw danemu antygenowi dopasowuje się do niego jak klucz do zamka. Monoklonalność oznacza, że dane przeciwciała wytwarzane są przez ten sam klon limfocytów B; wykazują one taką samą swoistość względem danego antygenu.

Jedną z metod oceny tej reakcji jest immunofluorescencja bezpośrednia, w której stosowane są przeciwciała znakowane specjalnym barwnikiem fluorescencyjnym (fluorochromem). Powstające w wyniku połączenia przeciwciał z antygenami kompleksy można obserwować wykorzystując zjawisko fluorescencji. Wiąże się ono z przejściem cząsteczki do wyższego stanu elektronowego na skutek oddziaływania na nią fali świetlnej o odpowiedniej długości. Podczas powrotu cząsteczki do stanu podstawowego zachodzi emisja światła. Długość fali światła absorbowanego jest mniejsza niż światła emitowanego. Wynika to z degradacji części energii podczas przejść termicznych i bezpromienistych. Zjawisko to nosi nazwę *przesunięcia stokesowskiego*. Zjawisko immunofluorescencji można oceniać w świetle ultrafioletowym w mikroskopie lub za pomocą cytometru przepływowego. Przykładowy obraz komórek znakowanych przeciwciałami ukazuje ilustracja 2.1.

Wykorzystanie cytometrii przepływowej do badania fluorescencji to najlepsza obecnie metoda oceny immunofenotypu (cech antygenowych) poszczególnych populacji komórkowych. Przed rozpoczęciem badania cytometrycznego do probówek zawierających izolowane komórki krwi (leukocyty) dodawane są przeciwciała monoklonalne znakowane fluorochromem. W czasie inkubacji część z nich zostaje związana na powierzchni komórek. Po przepłukaniu i odwirowaniu materiału w danej próbce zostają jedynie przeciwciała związane ze specyficznymi dla siebie antygenami. Dzięki fluorescencji znakowanych przeciwciał możliwa jest ocena ilościowa i jakościowa leukocytów wykazujących na swojej powierzchni ekspresję (obecność) danego antygenu.

Cytometria przepływowa pozwala także określić cechy fizyczne komórek (wielkość, stopień ziarnistości). Za pomocą odpowiedniego oprogramowania można na podstawie wyznaczonych parametrów (wielkość, ziarnistość, fluorescencja) wydzielić z badanej próby subpopulacje charakteryzujące

[1] Przy pisaniu rozdziału Ćytometria przepływowa w diagnostyce nowotworów" korzystano z następujących źródeł: [vLvdVW+04], [pdhJe97], [KMLO02].



Rysunek 2.1. Limfocyty T znakowane przeciwciałem anty-CD45.
Źródło: `http://www.abcam.co.jp/index.html?pageconfig=reviews&intAbID=8476&strFilterApplication=ICC/IF`

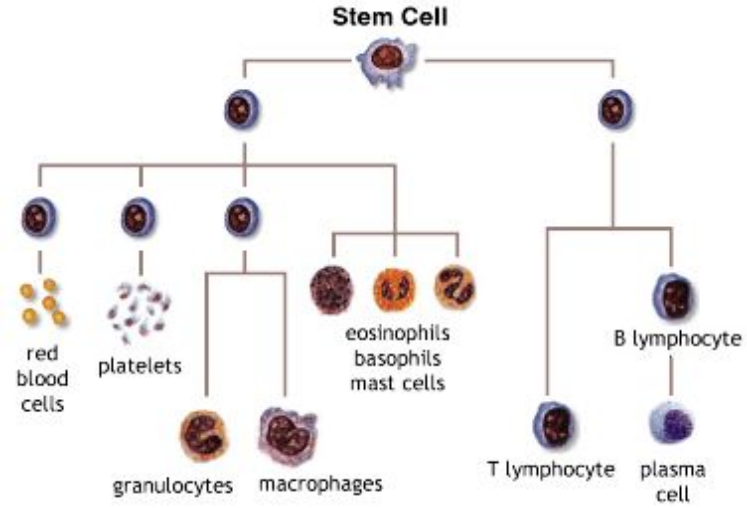
się określonymi właściwościami. Metoda cytofluorometrii przepływowej pozwala określić np. liczbę limfocytów (małe komórki, o małym stopniu ziarnistości i silnej ekspresji antygenu powierzchniowego CD45), w tym limfocytów T (ekspresja antygenów powierzchniowych CD2, CD3, CD7), a w tej subpopulacji stosunek limfocytów T pomocniczych (obecność antygenu powierzchniowego CD4) do limfocytów T supresorowych (obecność antygenu powierzchniowego CD8). Uzyskanie takich danych umożliwia postawienie lub wykluczanie rozpoznania wielu schorzeń dotyczących układu immunologicznego, w tym rozrostów nowotworowych komórek układu krwiotwórczego.

2.2 Diagnostyka rozrostów nowotworowych

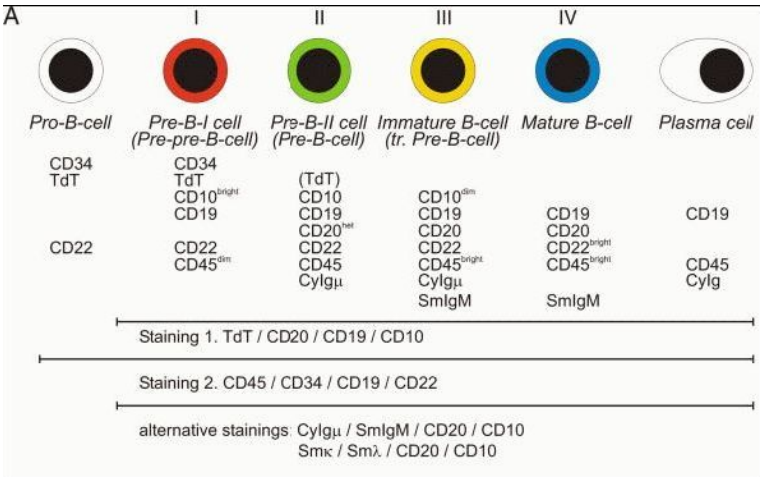
Wszelkie rodzaje komórek krwi wywodzą się z komórek macierzystych (ang. *stem cell*). W czasie rozwoju nabierają charakterystycznych dla siebie cech i zaczynają spełniać określone funkcje - proces ten jest przedstawiony na rysunku 2.2.

Na podstawie immunofenotypu komórek oraz cech morfologicznych można określić ich dojrzałość i zróżnicowanie. Na rysunku 2.3 przedstawiono etapy rozwojowe oraz charakteryzujące je markery dla limfocytów B.

Odchylenia jakościowe lub ilościowe fenotypu populacji leukocytów mogą być oznaką choroby. Procentowy udział poszczególnych subpopulacji w badanej próbie powinien mieścić się w określonych przedziałach (normach zależnych m.in. od wieku pacjenta i badanego materiału). Zdominowanie całej populacji przez jeden rodzaj komórek, pojawienie się komórek nietypowych, o parametrach istotnie różnych od obserwowanych w zdrowej populacji komórek lub niskozróżnicowanych, może świadczyć o procesie nowotworowym. Dla lekarza klinicysty niezmiernie istotne jest nie tylko samo stwierdzenie rozplemu nowotworowego ale również określenie jego przynależności do linii i etapu różnicowania. Właściwe rozpoznanie ma praktyczne znaczenie w doborze sposobu i ocenie leczenia oraz przewidywaniu prognozy.



Rysunek 2.2. Różnicowanie komórek układu krwiotwórczego. Źródło: `http://cmbi.bjmu.edu.cn/cmbidata/stem/transplatation/trans00.htm`



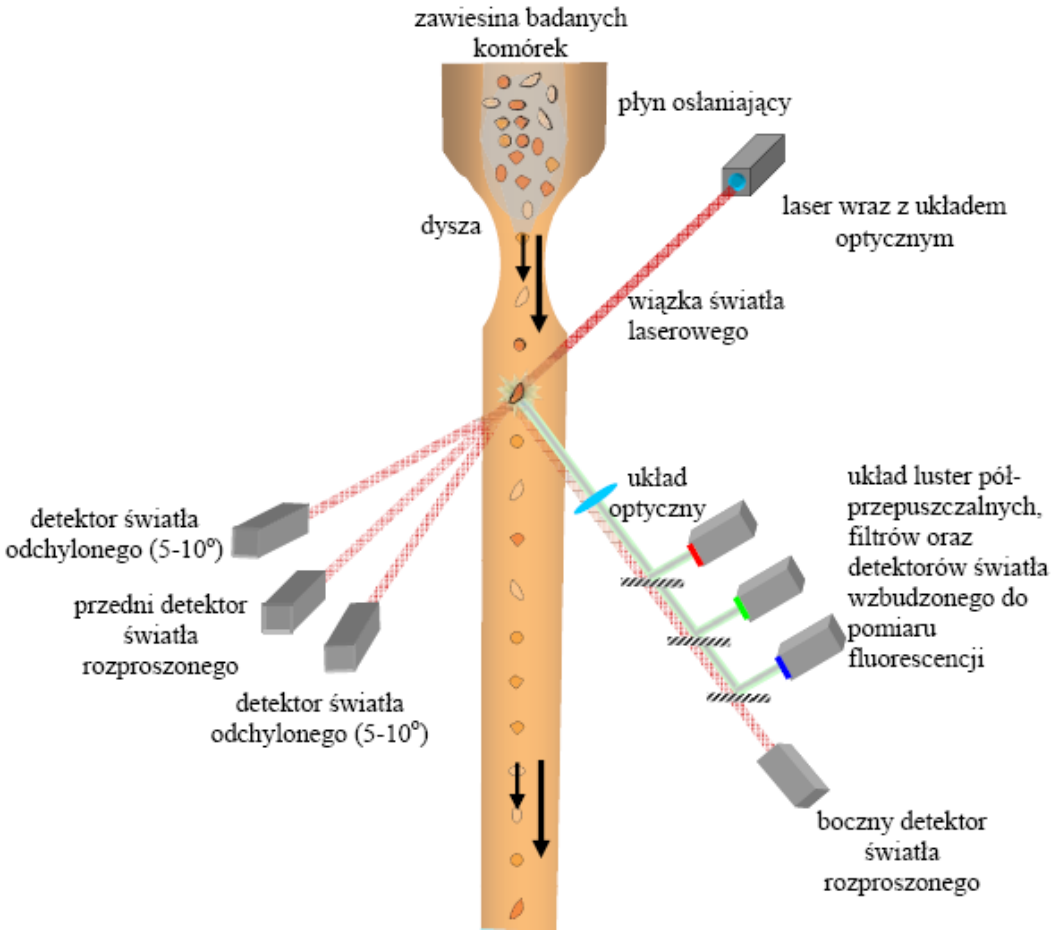
Rysunek 2.3. Etapy rozwojowe i typowe dla nich antygeny na przykładzie limfocytów B.
Źródło: [vLvdVW+04]

2.3 Cytometria przepływowa

Urządzeniem służącym do analizy komórek metodą cytofluorymetrii przepływowej jest cytofluorymetr. Schemat działania aparatu znajduje się na rysunku 2.4. W cytofluorymetrze można wyróżnić kilka układów:

- system przepływowy, który zajmuje się wymuszaniem laminarnego przepływu pojedynczych komórek,
- układ laserów wytwarzających wiązki świetlne umożliwiające pomiar rozpraszania i fluorescencji,
- układ optyczny, który skupia, odbija i filtruje światło,
- układ detektorów mierzących parametry światła wzbudzonego i rozproszonego.

Do przeprowadzenia badania niezbędny jest komputer podłączony do cytofluorymetru, który zbiera odczytywane dane i zapisuje je na dysku w postaci pliku formatu *FCS*.

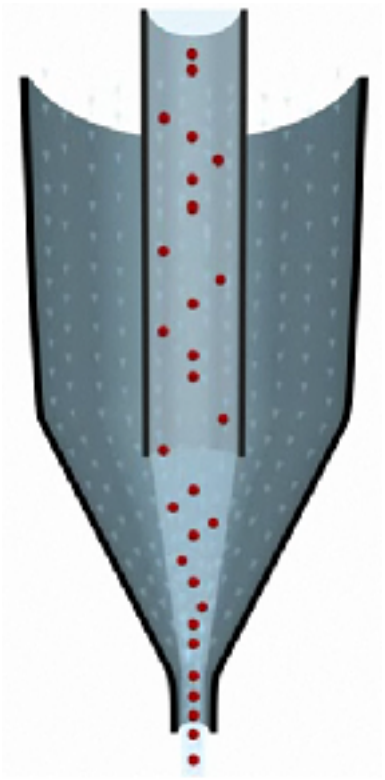


Rysunek 2.4. Schemat działania cytometru przepływowego.
Źródło: `http://pl.wikipedia.org/wiki/Cytometria_przep%C5%82ywowa`

W celu przeprowadzenie badania należy umieścić w cytofluorymetrze probówkę z odpowiednio przygotowaną zawiesiną wyizolowanych komórek. Dla poprawności otrzymywanych wyników niezwykle ważne jest to, żeby przez wiązkę lasera przechodziła w danej chwili tylko jedna komórka. W tym celu stosuje się technikę ogniskowania hydrodynamicznego przedstawioną na rysunku 2.5. Polega ona na wstrzykiwaniu komórek w dużo szybciej płynący strumień roztworu soli lub płynu osłonowego. Otrzymywana zawiesina jest wtłaczana w cienką dyszę, z której wypływa strumień o średnicy około $30\mu m$.

Kiedy komórka przechodzi przez wiązkę lasera światło jest załamywane i rozpraszane we wszystkich kierunkach. Światło padające na detektory powoduje impuls napięcia, który jest podstawą do obliczenia wartości numerycznej.

Przedni detektor światła rozproszonego (ang. *forward scatter*) mierzy ilość światła rozproszonego wokół osi wiązki lasera. Wartość odczytana na detektorze jest proporcjonalna do rozmiaru komórki. Jak już wspomniano wcześniej, komórka przechodząc przez wiązkę światła laserowego rozprasza je we wszystkie strony. Odbijanie światła pod dużym kątem powodowane jest przez ziarnistość wnętrza komórki (rysunek 2.6). Światło to jest skupiane przez soczewkę i mierzone przez boczny detektor światła rozproszonego (ang. *side scatter*). Soczewka zbiera światło odbite o około 90° od wiązki lasera. Wartość odczytana na detektorze światła rozproszonego jest proporcjonalna do stopnia ziarnistości komórki. Do pomiaru fluorescencji w cytometrze używane są z reguły dwa lasery emitujące światło i kilka do kilkunastu detektorów reagujących na różne długości



Rysunek 2.5. Ogniskowanie hydrodynamiczne.
Źródło: `http://probes.invitrogen.com/resources/education/`

fal.[2]. Mierzenie parametrów fluorescencji komórki odbywa się podobnie jak mierzenie rozproszenia bocznego - światło rozproszone skupiane jest w soczewce i skierowane w układ filtrów oraz luster półprzepuszczalnych, których zadaniem jest skierowanie fal świetlnych o odpowiedniej długości na właściwe detektory, jak przedstawiono na rysunku 2.7

Długość fali światła emitowanego przez wzbudzoną komórkę zawarta jest w pewnym przedziale zwanym spektrum emisji. Zależy ono od rodzaju fluorochromu użytego do znakowania przeciwciał. Przykładowe spektrum emisji przedstawione jest na rysunku 2.8.

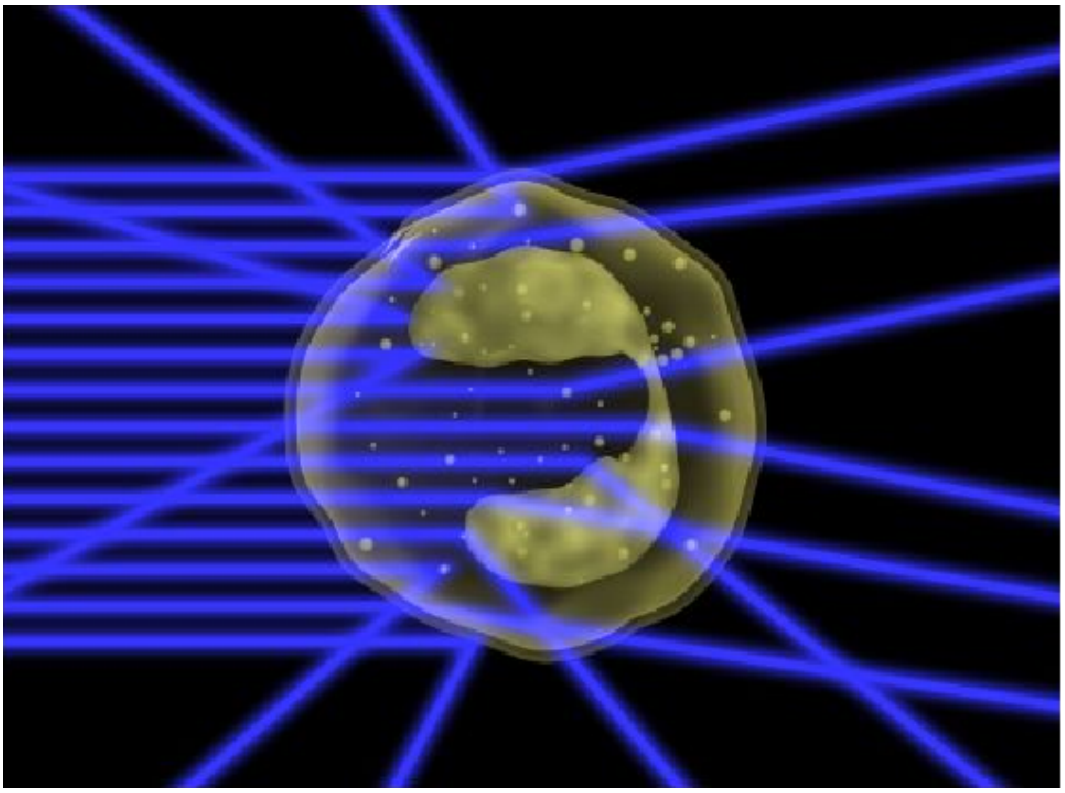
Jak widać na ilustracji, spektra emisji są stosunkowo szerokie i częściowo się pokrywają. W konsekwencji rejestrowane wartości światła emitowanego przez jeden z rodzajów fluorochromów mogą być fałszywie zwiększone przez emisję innego. Aby uniknąć tego zjawiska, podczas przygotowywania materiału do badania do tej samej probówki dodawane są fluorochromy o możliwie najmniejszym wspólnym zakresie emisji światła. Dodatkowo detektory światła emitowanego posiadają wbudowane filtry pasmowoprzepustowe, które ograniczają ich czułość do przedziału wokół wartości maksymalnej spektrum emisji. Niestety, nie rozwiązuje to do końca problemu nakładania się przedziałów, co widać na rysunku 2.9. W celu całkowitego wyeliminowania wpływu świecenia innych fluorochromów na wartość odczytywanego wykorzystuje się proces kompensacji.

Przyjmijmy następujące oznaczenia:

n - liczba fluorochromów odczytywanych przez cytofluorymetr,

P - zbiór wartości odczytanych na detektorze, P_i oznacza wartość zmierzonego świecenia fluorochromu i, gdzie $i = 1 \ldots n$,

[2]Aparat wykorzystywany w Zakładzie może rejestrować światło w 6 różnych długościach fali.



Rysunek 2.6. Rozpraszanie boczne
Źródło: `http://probes.invitrogen.com/resources/education/`

M - macierz o elementach $M_{i,j}, i, j = 1 \ldots n$, której elementy oznaczają procentowy wpływ wartości fluorochromu j na wartość odczytaną jako świecenie fluorochromu i (sposób wyznaczania macierzy opisany jest w dalszej części tekstu),

C - zbiór wartości parametrów komórki po kompensacji, C_i oznacza wartość zmierzonego świecenia fluorochromu i po uwzględnieniu kompensacji, gdzie $i = 1 \ldots n$.

Dla każdej komórki można obliczyć

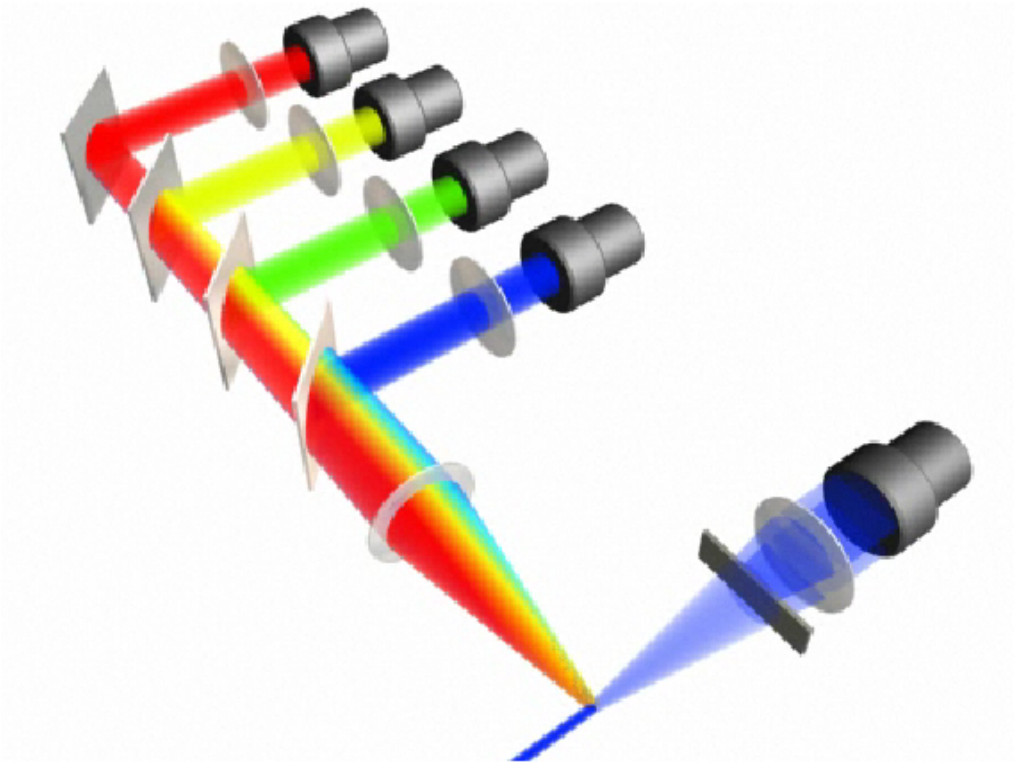
$$\forall i = 1 \ldots n\colon\ C_i = P_i - \sum_{j \neq i, j=1 \ldots n} M_{i,j} * P_j \tag{2.1}$$

Macierz M wyznaczana jest podczas kalibracji aparatu. Analizie poddawane są wówczas próbki oznaczone tylko jednym fluorochromem. Obliczane wartości macierzy M zależą od stosunku intensywności świecenia rejestrowanej na poszczególnych detektorach do wartości odczytanej przez detektor właściwy dla danego fluorochromu.

Wyznaczanie skompensowanych wartości może zachodzić w samym cytofluorymetrze lub później, przy użyciu odpowiedniego oprogramowania. W ostatnim przypadku macierz M zostaje dopisana przez urządzenie do pliku z danymi.

Ze względu na zjawisko nakładania się zakresów długości fal charakteryzujących poszczególne fluorochromy ograniczona jest liczba możliwych do jednoczesnego badanych antygenów powierzchniowych. Obecnie oceniania się cztery do sześciu markerów na tej samej komórce[3]. Jednak w wielu przypadkach, m.in. w diagnostyce rozrostów nowotworowych, konieczna jest analiza obejmująca większą ilość antygenów. Badanie cytometryczne wykonuje się wówczas dla wielu próbek tego

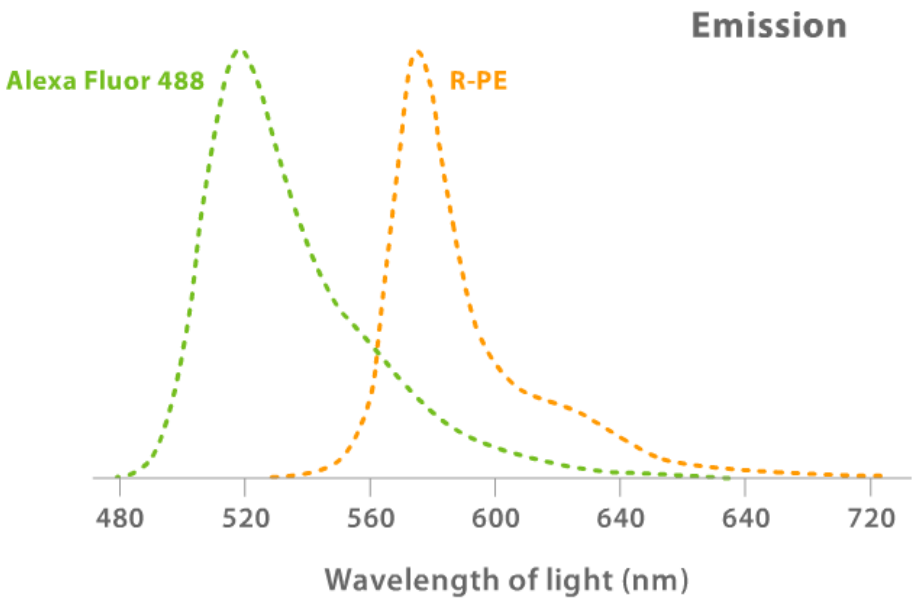
[3]Względy ekonomiczne powodują jednak, że najczęściej w Zakładzie stosuje się do czterech różnych przeciwciał na raz.



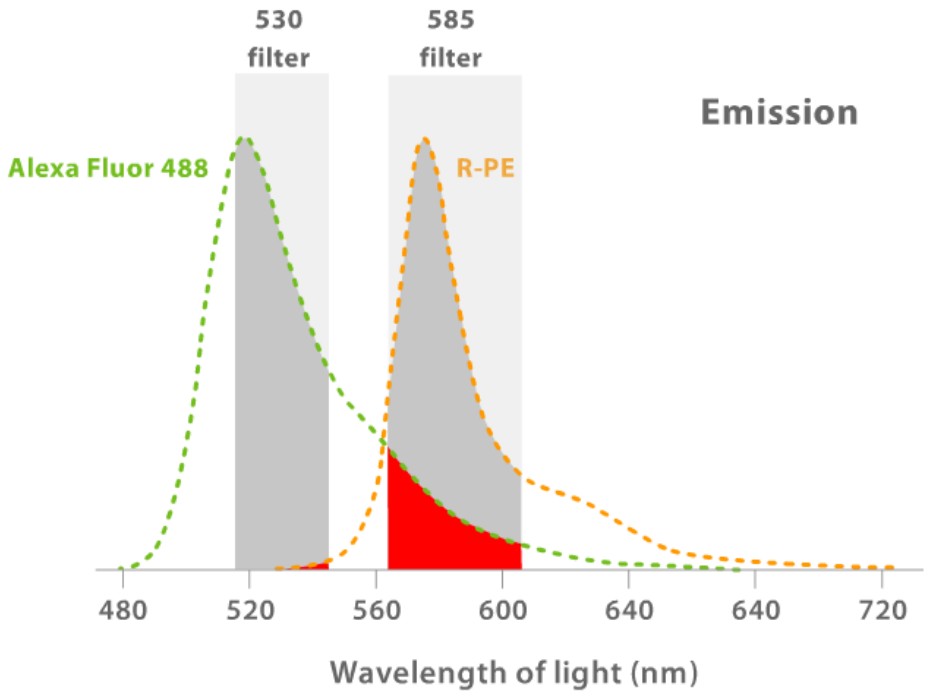
Rysunek 2.7. Mierzenie fluorescencji.
Źródło: `http://probes.invitrogen.com/resources/education/`

samego materiału wyznakowanych różnymi przeciwciałami. Można w takiej sytuacji stosować w kolejno analizowanych probówkach odmienne przeciwciała monoklonalne znakowane identycznym fluorochromem.

Istotną dla dalszej analizy częścią badania jest wykonanie próby kontrolnej ujemnej. Do jednej z probówek z zawiesiną komórek nie zostają dodane żadne przeciwciała. Na podstawie wartości rejestrowanych przez poszczególne detektory ustala się odpowiednie progi, powyżej których daną wartość będziemy uznawać za dodatnią. Pomiary poniżej progu są traktowane jako ujemne. Kolejny problem związany z wykorzystaniem cytometrii przepływowej to rejestrowanie przez detektory obiektów (zdarzeń), które odpowiadają komórkom uszkodzonym lub ich fragmentom. Materiał biologiczny używany w badaniu nie jest utrwalany, komórki często ulegają uszkodzeniu i rozpadowi. Zjawisko to nasila się w niektórych procesach nowotworowych. Uszkodzone komórki oraz ich fragmenty łączą się z przeciwciałami (nieswoiście) i są rejestrowane przez detektory. Wystąpienie takiego zdarzenia nie musi być zatem związane z obecnością na ich powierzchni badanego antygenu. Częściowym rozwiązaniem tego problemu jest odrzucenie z analizowanego zbioru zdarzeń elementów o rozmiarach mniejszych niż komórka. Jednak w wielu przypadkach ocena materiału z dużą ilością rozpadających się komórek jest bardzo trudna lub wręcz niemożliwa.



Rysunek 2.8. Wykres natężenia światła emitowanego w zależności od długości fali światła dla dwóch przykładowych fluorochromów.
Źródło: `http://probes.invitrogen.com/resources/education/`



Rysunek 2.9. Zastosowanie filtrów pasmowoprzepustowych.
Źródło: `http://probes.invitrogen.com/resources/education/`

Rozdział 3

Analiza skupień

3.1 Wiadomości wstępne

Analiza skupień (ang. data clustering) jest jedną z technik eksploracji danych (ang. data mining) i służy do podziału badanych obiektów na pewne klasy (grupy, skupienia) obiektów podobnych. Optymalny podział obiektów powinien spełniać następujące warunki:

- rozłączność klas - przedstawiciele dwóch różnych grup muszą wykazywać istotne różnice pod kątem uwzględnionych cech,
- zwartość klas - obiekty wewnątrz tej samej klasy nie powinny się znacznie różnić od siebie pod kątem uwzględnionych cech.

Dążenie do zapewnienia pierwszego warunku wydaje się być oczywiste. Sytuacja, w której obiekty z dwóch różnych klas wykazują znaczne podobieństwa, podważa sens przydzielenia ich do różnych grup.

Drugi z rozpatrywanych warunków ma również duże znaczenie praktyczne. Każda grupa powinna być w miarę możliwości jednorodna, w przeciwnym razie należy rozważyć podzielenie rozpatrywanej grupy na kilka mniejszych grup. Niezależnie od tego która własność grupowania jest pożądana, niezbędne jest uściślenie pojęcia podobieństwa obiektów. By móc automatycznie porównywać obiekty, należy najpierw zidentyfikować cechy (atrybuty) jakie są istotne przy ich porównywaniu. Porównanie obiektów odbywa się poprzez obliczenie odległości między nimi, a zatem wskazane jest by atrybuty je opisujące miały charakter ilościowy[1].

3.2 Normalizacja

Często badane atrybuty, nawet przedstawione za pomocą wartości liczbowych, nie mogą być porównywane ze sobą. Wzięcie pod uwagę śurowych"wartości może wiązać się z efektem skali, różnice na części atrybutów mogą przeważać różnice na innych atrybutach. Porównując przykładowo mieszkania opisywane przez cenę za m^2 i wielkość w m^2, na pierwszy rzut oka widać, że dowolna odległość między poszczególnymi obiektami będzie zależała w dużej mierze od ceny. Same wartości tych cech nie dają pełnej informacji o różnicach występujących między poszczególnymi obiektami. Dzieje się tak, gdyż cena przyjmuje znacznie większe wartości niż wielkość mieszkania. Dlatego też, zanim policzone zostaną jakiekolwiek odległości, należy zniwelować różnice w rozkładach jakie przyjmują dane cechy. Można to uzyskać stosując procedury normalizacji [dhTP]:

[1] Dzięki temu można skorzystać z dowolnie zdefiniowanej dla przestrzeni euklidesowej odległości.

- unitaryzacja

$$z_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}, \tag{3.1}$$

- standaryzacja

$$z_i = \frac{x_i - \bar{x}}{s}, \tag{3.2}$$

W każdym z powyższych wzorów przyjęto następujące oznaczenia:

x_i - wartość pewnego atrybutu dla i-tego obiektu przed normalizacją,

z_i - wartość pewnego atrybutu dla i-tego obiektu po normalizacji,

x_{max} - wartość maksymalna jaką przyjmuje dany atrybut w zbiorze badanych obiektów,

x_{min} - wartość minimalna jaką przyjmuje dany atrybut w zbiorze badanych obiektów,

$\bar{x}$ - wartość średnia atrybutu policzona na zbiorze badanych obiektów,

s - odchylenie standardowe atrybutu policzone na zbiorze badanych obiektów.

3.3 Odległości między obiektami

Po zastosowaniu jednej z wybranych metod od normalizacji wartości dla każdego z badanych atrybutów można przejść do następnego etapu, jakim jest zdefiniowanie odległości między obiektami. To na jej podstawie można stwierdzić czy obiekty różnią się znacznie czy też nie.

Niech X będzie niepustym zbiorem. Odległością nazywamy funkcję $d: X \times X \to R$, która dla dowolnych $x_1, x_2, x_3 \in X$ spełnia następujące warunki:

- $d(x_1, x_2) = 0 \iff x_1 = x_2$,
- $d(x_1, x_2) = d(x_2, x_1)$,
- $d(x_1, x_2) \leq d(x_1, x_3) + d(x_3, x_2)$.

Istnieje wiele modeli matematycznych opisujących odległości. Do najpopularniejszych należą (w poniższych wzorach założono, że każdy obiekt jest opisany za pomocą m atrybutów [Sta06]):

- odległość euklidesowa

$$d(x, y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}, \tag{3.3}$$

- odległość miejska

$$d(x, y) = \sum_{i=1}^{m} |x_i - y_i|, \tag{3.4}$$

- odległość Czebyszewa

$$d(x, y) = \max_{i=1}^{m} |x_i - y_i|, \tag{3.5}$$

- odległość potęgowa

$$d(x, y) = \sqrt[q]{\sum_{i=1}^{m} (x_i - y_i)^p}. \tag{3.6}$$

3.4 Metody analizy skupień

Metody klasyfikacji obiektów można podzielić pod względem mechanizmu grupowania na dwie zasadnicze kategorie [HMS01]:

- metody hierarchiczne,
- metody niehierarchiczne.

Metody hierarchiczne tworzą pewną hierarchiczną strukturę podziału, bądź grupowania obiektów. Wyniki poszczególnych iteracji tworzą dendrogram, z diagramu tego można odczytać przebieg całego procesu grupowania. Cechą metod hierarchicznych jest dokonywanie operacji podziału lub łączenia na całych skupieniach a nie pojedynczych obiektach.

3.4.1 Algorytm aglomeracyjny

Wśród metod hierarchicznych wyróżnia się podejście aglomeracyjne i deglomeracyjne. Pierwsze z nich zakłada, że na początku każdy obiekt stanowi odrębne skupienie. Algorytm grupowania w każdej iteracji wybiera dwa skupienia najmniej odległe od siebie i łączy je w większe skupienie. Po operacji połączenia dwóch skupień konieczne jest obliczenie odległości między nowym skupieniem a pozostałymi. Istnieje wiele metod, odległość ta zależy od odległości od łączonych skupień a także odległości między łączonymi skupieniami.

Przyjmijmy następujące oznaczenia:

p, q - indeksy skupień wybranych do połączenia,

$k+1$ - indeks nowego skupienia powstałego po połączeniu skupień p, q,

$d_{i.j}$ - odległość między skupieniami o indeksach i, j,

n_i - liczebność skupienia i.

Odległość między istniejącym skupieniem (o indeksie i) można wyznaczyć na wiele sposobów, poniżej zaprezentowano klika najbardziej popularnych [Sta06]:

- metoda najbliższego sąsiedztwa:

$$d_{k+1,i} = 0.5d_{p,i} + 0.5d_{q,i} - 0.5|d_{p,i} - d_{q,i}|, \tag{3.7}$$

- metoda najdalszego sąsiedztwa:

$$d_{k+1,i} = 0.5d_{p,i} + 0.5d_{q,i} + 0.5|d_{p,i} - d_{q,i}|, \tag{3.8}$$

- metoda mediany:

$$d_{k+1,i} = 0.5d_{p,i} + 0.5d_{q,i} + 0.25d_{p,q}, \tag{3.9}$$

- metoda średniej grupowej:

$$d_{k+1,i} = \frac{n_p}{n_p + n_q} d_{p,i} + \frac{n_q}{n_p + n_q} d_{q,i}, \tag{3.10}$$

- metoda środka ciężkości:

$$d_{k+1,i} = \frac{n_p}{n_p + n_q} d_{p,i} + \frac{n_q}{n_p + n_q} d_{q,i} - \frac{n_p n_q}{(n_p + n_q)^2} d_{p,q}, \tag{3.11}$$

- metoda Warda:

$$d_{k+1,i} = \frac{n_p + n_i}{n_p + n_q + n_i} d_{p,i} + \frac{n_q + n_i}{n_p + n_q + n_i} d_{q,i} - \frac{n_i}{n_p + n_q + n_i} d_{p,q}. \tag{3.12}$$

Algorytmy aglomeracyjne mogą być zakończone po osiągnięciu zadanej liczby skupień, bądź w momencie kiedy odległości pomiędzy poszczególnymi skupieniami są na tyle duże, że dalsze ich łączenie mija się z celem.

Rozpatrując złożoność obliczeniową tego podejścia należy zauważyć, że:

- na wstępie należy obliczyć odległość między każdą parą obiektów, zatem dla n obiektów jest to $\frac{n^2}{2}$ operacji,
- w każdej iteracji należy wybrać dwa najbliższe obiekty - przejrzenie całej macierzy odległości $\frac{(n-k)^2}{2}$ operacji, gdzie k jest liczbą skupień w danej iteracji,
- w każdej iteracji należy policzyć nowe odległości - kolejne $(n-k)$ operacji,
- w najgorszym wydaniu algorytm połączy wszystkie obiekty w jedno skupienie, zrobi to po przejściu $(n-1)$ iteracji.

Biorąc pod uwagę powyższe spostrzeżenia można stwierdzić, że algorytm ma złożoność obliczeniową rzędu $O(n^3)$.

3.4.2 Algorytm deglomeracyjny

Podejście deglomeracyjne jest przeciwieństwem podejścia aglomeracyjnego. Na samym wstępie wszystkie obiekty należą do jednej grupy. W każdym kolejnym kroku, dla każdego skupienia wybierane są obiekty najdalej położone od siebie. Następnie z tych par obiektów wybierana jest ta, dla której odległość jest największa. W ten sposób skupienie, do którego należały wybrane obiekty zostaje wyznaczony do podziału, a obiekty trafiają do nowych skupień. Ostatnim krokiem pojedynczej iteracji algorytmu jest przydzielenie obiektów do nowych grup [HMS01]. W tym celu należy policzyć odległość każdego obiektu od obu skupień oraz przydzielić go do bliższego z nich. Odległość obiektu od skupienia można policzyć na wiele sposobów, kilka z nich zaproponowano poniżej:

- odległość od najbliższego obiektu w skupieniu,
- odległość od najdalszego obiektu w skupieniu,
- odległość od obiektu średniego (sztucznego obiektu, którego wartości wszystkich atrybutów są średnią wartości ze skupienia).

Algorytmy deglomeracyjne mogą być zakończone po osiągnięciu zadanej liczby skupień, bądź w momencie, kiedy odległości wewnątrz poszczególnych skupień są na tyle małe, że ich dalsze dzielenie mija się z celem.

Obie metody mają swoje wady i zalety. W przypadku dużej ilości obiektów i stosunkowo niewielkiej docelowej liczbie dużych skupień, znacznie wydajniejsze wydaje się stosowanie metod deglomeracyjnych. Z drugiej strony gdy na wyjściu pożądana jest duża liczba niewielkich grup, bardziej uzasadnione jest wykorzystanie metod aglomeracyjnych.

Rozpatrując złożoność obliczeniową podejścia deglomeracyjnego należy zauważyć, że:

- w każdej iteracji należy obliczyć odległość między każdą parą obiektów w każdym skupieniu, zatem w najgorszym przypadku (kiedy większość obiektów należy do jednego skupienia) dla n obiektów i k skupień to jest $\frac{(n-k)^2}{2}$ operacji,
- w każdej iteracji należy przydzielić wszystkie obiekty z dzielonego skupienia do nowych skupień - w tym samym przypadku $(n-k)$ operacji,
- w najgorszym przypadku algorytm podzieli skupienia na pojedyncze obiekty, zrobi to po przejściu $(n-1)$ iteracji.

Na podstawie powyższych uwag można stwierdzić, że algorytm ma złożoność obliczeniową rzędu $O(n^3)$.

3.4.3 Algorytm k-średnich

W metodach niehierarchicznych w zupełnie inny sposób podchodzi się do grupowania obiektów. Tu przez cały czas działania algorytmu liczba skupień jest stała i zadana z góry. Poszczególne iteracje polegają na realokacji obiektów pomiędzy skupieniami tak, by nowy podział dążył do podziału optymalnego. W ogólności problem ten jest obliczeniowo trudny, znalezienie optymalnego podziału wymaga przejrzenia całej przestrzeni stanów. Dlatego stosuje się pewne heurystyki, które bazują na optymalizowaniu lokalnych rozwiązań, tym samym zbliżając się do rozwiązania optymalnego globalnie.

Jednym z tego typu algorytmów jest algorytm k-średnich. Na wstępie wybieranych jest losowo k obiektów, które będą stanowiły środki skupień. Na podstawie odległości każdego z pozostałych obiektów do wybranych środków, są one przydzielane do poszczególnych skupień. W każdej kolejnej iteracji, w każdym skupieniu obliczany jest sztuczny element środkowy. Element ten, zwany centroidem, na każdym atrybucie ma średnią arytmetyczną z wartości jakie ten atrybut przyjmuje w danym skupieniu. Dla tak wyznaczonych środków wszystkie obiekty po raz kolejny są przydzielane do nowych grup. Proces wyznaczania centroidów i przyporządkowywania obiektów do skupień kończy się, gdy wyznaczone centroidy nie zmieniają się [HMS01].

Rozpatrując złożoność obliczeniową tego podejścia (w najprostszym wydaniu) należy zauważyć, że w każdej iteracji należy przydzielić wszystkie obiekty do jednego ze skupień - $(n-k)k$ operacji, gdzie n jest liczbą obiektów a k liczbą klastrów.

Okazuje się, że złożoność tego podejścia jest znacznie niższa niż dwóch pozostałych i jest rzędu $O(np)$, gdzie p jest liczbą iteracji.

Rozdział 4

Wykorzystane technologie

4.1 Język C# i platforma .NET

C# jest językiem programowania zorientowanym obiektowo. Zaprojektowany został przez firmy Microsoft. W chwili obecnej jest on uznany za standard *ECMA* oraz *ISO*.

Język C# charakteryzuje się obiektowością o jednym elemencie nadrzędnym (`System.Object`). W szczególności oznacza to, że typy proste (np. `int`) również są obiektami z właściwymi sobie metodami. Podobnie jak w języku Java, odzyskiwaniem (czyszczeniem) pamięci zajmuje się środowisko uruchomieniowe. W odróżnieniu od innych języków programowania, C# pozwala na definiowanie wielu elementów składowych klas oprócz pól i metod - można definiować właściwości klasy, indeksery, delegacje i zdarzenia. Od wersji drugiej języka możliwe jest wykorzystanie typów ogólnych (generycznych) - jest to mechanizm zbliżony do szablonów z C++. Standard języka C# opisany jest w dokumencie [HWG06].

Programy napisane w języku C# kompilowane są do postaci kodu pośredniego (ang. *Common Intermediate Language*). W chwili uruchomienia, program kompilowany i wykonywany jest przez kompilator *JIT* (ang. *just-in-time compiler*) będący częścią środowiska uruchomienowego platformy *.NET*.

Platforma *.NET* jest środowiskiem, które ma umożliwiać bezproblemowe wytwarzanie i uruchamianie aplikacji okienkowych (bazujących na Windows Forms), aplikacji internetowych oraz aplikacji mobilnych (z wykorzystaniem platformy Compact Framework). Platforma *.NET* jest przenośna. Każdy program napisany dla niej da się uruchomić w dowolnym systemie operacyjnym działającym na dowolnej platformie sprzętowej posiadającym zainstalowaną odpowiednią wersję środowiska. Oprócz platformy .NET oferowanej przez Microsoft, dostępna jest jej otwarta implementacja - platforma Mono. Dostępna jest na wszystkie wiodące systemy operacyjne (m. in. Linux, Mac OS X, Windows).

Pod względem architektury platformę *.NET* można podzielić na: wspólne środowisko uruchomieniowe (*CLR*, ang. *Common Language Runtime*) oraz bibliotekę klas platformy (*FCL*, ang. *Framework Class Library*). Dokładny opis architektury znaleźć można w książce [Per06].

Istotną cechą platformy jest jej niezależność od konkretnego języka programowania. Częścią platformy jest wspólny system typów (*CTS*, ang. *Common Type System*), który specyfikuje jakie typy i struktury danych są obsługiwane przez platformę i jak mogą ze sobą współdziałać. Dzięki temu, kod programów na platformę *.NET* można pisać w każdym języku programowania spełniającym wymogi *CTS*.

4.2 Visual C# 2005 Express Edition

Visual C# to zintegrowane środowisko programistyczne (*ang. Integrated Development Environment, IDE*) wykorzystywane do tworzenia aplikacji dla platformy .NET. Tworzenie aplikacji okienkowych w tym środowisku jest znacznie ułatwione dzięki możliwości wizualnego projektowania interfejsu użytkownika (umieszczanie kontrolek na formatce odbywa się metodą *drag'n'drop*). Edytor kodu wykorzystuje technologię `IntelliSense`, która pozwala na uzupełnianie kodu. Środowisko umożliwia wstawianie szkieletów bloków kodu (ang. *snippets*), a nawet częściowo je wypełnia - np. blok `switch`. Technologia `IntelliSense` pozwala również na łatwą refaktoryzację kodu.

Visual C# 2005 Express Edition jest darmową wersją środowiska Visual Studio pozwalającą na tworzenie oprogramowania komercyjnego. W odróżnieniu od pełnych wersji środowiska, edycje Express obsługują tylko jeden język programowania oraz nie mają kilku dodatkowych narzędzi. Dla potrzeb realizowanego projektu możliwości edycji Express są wystarczające.

4.3 Subversion

Subversion (`SVN`) to system kontroli wersji, czyli oprogramowanie do zarządzania kolejnymi wersjami plików. Wspomaga ono pracę zespołu programistów poprzez łączenie i śledzenie zmian w dokumentach modyfikowanych przez różne osoby w różnych momentach czasu. Projekt `SVN` powstał jako alternatywa dla systemu `CVS` mająca implementować i rozszerzać jego funkcjonalność przy uniknięciu jego wad. Subversion jest udostępniany na licencji `Apache`. Adres projektu: `http://subversion.tigris.org/`.

4.4 Biblioteka ZedGraph

Biblioteka ZedGraph została napisana w języku C#. Służy do tworzenia wykresów dwuwymiarowych, zarówno liniowych jak i słupkowych. Wykresy są w dużym stopniu modyfikowalne - praktycznie każda własność wykresu może być modyfikowana przez programistę. Kontrolka z wykresem umożliwia użytkownikowi przybliżanie oraz oddalanie obrazu, możliwy jest również wydruk wykresu, jego eksport do pliku graficznego lub skopiowanie do schowka. Biblioteka udostępniona jest na licencji `LGPL`. Adres projektu: `http://zedgraph.org`.

4.5 DockPanel Suite Release 2.2

Biblioteka DockPanel Suite pozwala na tworzenie zaawansowanych interfejsów użytkownika. Umożliwia ona stosowanie mechanizmu zarządzania okienkami w aplikacji *MDI* (ang. *Multi Document Interface*), dzięki któremu użytkownik może w dowolny sposób modyfikować układ okienek. Z punktu widzenia użytkownika mechanizm ten przypomina podobny, zaimplementowany w środowisku Visual Studio 2005. Biblioteka ta posiada także wiele innych funkcji, wśród nich możliwość zapisywania aktualnej konfiguracji interfejsu użytkownika do pliku *XML*.

Wadą tej biblioteki jest brak kompletnej dokumentacji. W zamian, autorzy dostarczyli przykładową aplikację korzystającą z DockPanel Suite, która jest niezbędnym (i niestety jedynym) źródłem wiedzy na temat korzystania z tej biblioteki. Mimo to łatwość, z jaką można budować aplikacje *MDI* za pomocą DockPanel Suite, daje tej bibliotece dużą przewagę nad konkurencyjnymi rozwiązaniami (często komercyjnymi jak np. MagicLibrary). Biblioteka jest udostępniona na licencji *MIT*. Strona projektu: `http://sourceforge.net/projects/dockpanelsuite`.

4.6 Baza danych Oracle 10g

Baza danych Oracle 10g umożliwia przechowywania danych w postaci relacyjnych tabel będących zbiorem rekordów o identycznej strukturze. Tabele (a także inne obiekty bazy: perspektywy, indeksy) tworzone są w ramach schematu. Właścicielem schematu jest użytkownik, którego nazwa jest taka sama jak nazwa schematu. Baza Oracle umożliwia dostęp do danych przy użyciu języka *SQL*.

Użytkownicy mogą wykonywać działania na bazie w ramach posiadanych uprawnień. Uprawnienia nadawane są bezpośrednio lub poprzez rolę (nazwaną grupę uprawnień). W bazie Oracle występują dwie kategorie uprawnień: systemowe i obiektowe

Istnieje także możliwość tworzenia perspektyw. Umożliwiają one dostęp do danych z jednej lub wielu tabel. Perspektywa nie przechowuje danych, lecz pobiera je z tabel na których jest zbudowana. W bazie Oracle stosowane są również synonimy, które pozwalają odwoływać się do obiektów bazy (w tym tabel) przy pomocy innych nazw niż nazwa obiektu.

4.7 Język PL/SQL

PL/SQL jest proceduralnym językiem programowania dostępnym z bazą Oracle. W PL/SQL można umieszczać polecenia manipulacji danymi (`SELECT`, `INSERT`, `UPDATE`, `DELETE`) a także deklarować stałe, zmienne, typy, podprogramy (funkcje i procedury), stosować instrukcji warunkowe, pętle. Podstawowe jednostki (procedury, funkcje, anonimowe bloki) tworzące program PL/SQL mają strukturę bloków. Blok PL/SQL składa się z trzech sekcji: deklaracji, wykonania i obsługi wyjątków. Procedury i funkcje PL/SQL mogą być przechowywany w bazie danych (procedury i funkcje składowane). PL/SQL pozwala na grupowanie logicznie powiązanych funkcji i procedur w pakiety składowane w bazie danych. Pakiet składa się z dwóch części: specyfikacji (interfejsu) i ciała (implementacji). Dostęp użytkownika do bazy może być kontrolowany przez przyznanie uprawnień do wykonywania właściwych procedur, funkcji i pakietów bez nadawania uprawnień do tabel i perspektyw.

4.8 Standard FCS

Standard *FCS* (ang. *Flow Cytometry Standard*) jest formatem pliku powszechnie używanym przez producentów sprzętu, twórców oprogramowania oraz naukowców zajmujących się cytometrią przepływową. Dzięki elastyczności standardu nie ma problemów z kompatybilnością. Nowsze wersje standardu są w pełni zgodne ze starszymi. Dodatkowo, producenci sprzętu i oprogramowania mogą rozszerzać standard, definiując na własne potrzeby dodatkowe informacje umieszczane w pliku.

Aparatura wykorzystywana w Zakładzie generuje pliki zgodne z najnowszą, trzecią wersją standardu opisaną w dokumencie [SBB+96].

W pojedynczym pliku mogą znaleźć się wyniki kilku różnych badań. W pliku zapisana jest sekwencja bloków danych. Każdy blok reprezentuje jedno badanie i można w nim wyróżnić kilka segmentów:

HEADER segment ten zawiera informacje na temat położenia pozostałych segmentów w bloku danych oraz następnego bloku.

TEXT opisuje parametry detektorów oraz zapisanych danych, np. datę badania, nazwy wykorzystanych odczynników i mierzonych właściwości komórek. Wpisy w tym segmencie przyjmują

postać par: słowo kluczowe (nazwa atrybutu) - wartość atrybutu.

DATA segment ten zawiera dane zebrane podczas badania zapisane w sposób wyspecyfikowany w segmencie TEXT.

ANALYSIS segment ten może być pusty. Zapisywane są w nim wyniki wcześniej wykonanej analizy bloku danych.

Rozdział 5

Architektura systemu

5.1 Reprezentacja pliku FCS

Plik formatu *FCS* odczytany z dysku reprezentowany jest w programie przez obiekt klasy `FCS_System`. Klasa ta udostępnia jedną własność - zbiór bloków danych zapisanych w pliku. Konstruktor klasy przyjmuje jako parametr ścieżkę do pliku. W przypadku błędu przy odczycie pliku klasa `FCS_System` zgłasza wyjątek `FCS_Error` z informacją o błędzie. Możliwe błędy odczytu zawarte są w typie wyliczeniowym `ErrorCode`.

Klasa `FCS_DataSet` reprezentuje jeden blok danych, czyli wynik badania jednej próbki materiału. Segment *TEXT* znajduje odzwierciedlenie w obiekcie klasy `FCS_Keywords` oraz kolekcji obiektów klasy `FCS_Parameter`. Segment *DATA* reprezentowany jest w tablicy `Data`. Zapis `Data[i][j]` oznacza wartość j-tego parametru i-tego zdarzenia - jeden wiersz tablicy oznacza jedno zdarzenie[1].

Klasa `FCS_Parameter` reprezentuje parametry detektora oraz wykrywanej przez niego właściwości komórki. Dla detektorów wykrywających świecenie, z własności `FullName` i `ShortName` można odczytać nazwę wykrywanego fluorochromu oraz antygenu z którym łączy się znakowane barwnikiem przeciwciało. Parametr `Display` informuje w jakiej skali powinny być wyświetlane wartości odczytane przez ten detektor (może być np. liniowa lub logarytmiczna). Klasa `FCS_Keywords` przechowuje wszystkie słowa kluczowe[2] wraz z wartościami w słowniku `AllKeywords`. Dane niezbędne do odczytu bloku *DATA* przechowywane są również jako właściwości klasy. Warte wyszczególnienia są:

- `Events` - liczba zdarzeń odczytanych w czasie badania,
- `Parameters` - liczba parametrów opisujących każde zdarzenie,
- `Mode` - kolejność zapisu parametrów zdarzeń w segmencie *DATA*,
- `DataType` - sposób zakodowania w segmencie *DATA* wartości liczbowych[3],
- `Src` - nazwa badania.

Właściwość `DataType` może przyjąć jedną z następujących wartości:

- `ASCII` - wartości liczbowe zapisane są w postaci zwykłego ciągu znaków,

[1] Przyjęto, że zdarzenie oznacza każdy sygnał zarejestrowany przez cytofluorymetr w trakcie analizy. Zdarzenie może zatem odpowiadać całej komórce, ale może być też wynikiem błędnego pomiaru, spowodowanego np. obecnością zanieczyszczeń czy fragmentów uszkodzonych komórek.

[2] Poza tymi, które opisują parametry detektorów w klasie `FCS_Parameter`

[3] Właściwości `Mode` oraz `DataType` przyjmują wartości zdefiniowane w typach wyliczeniowych przedstawionych na diagramie 5.1.

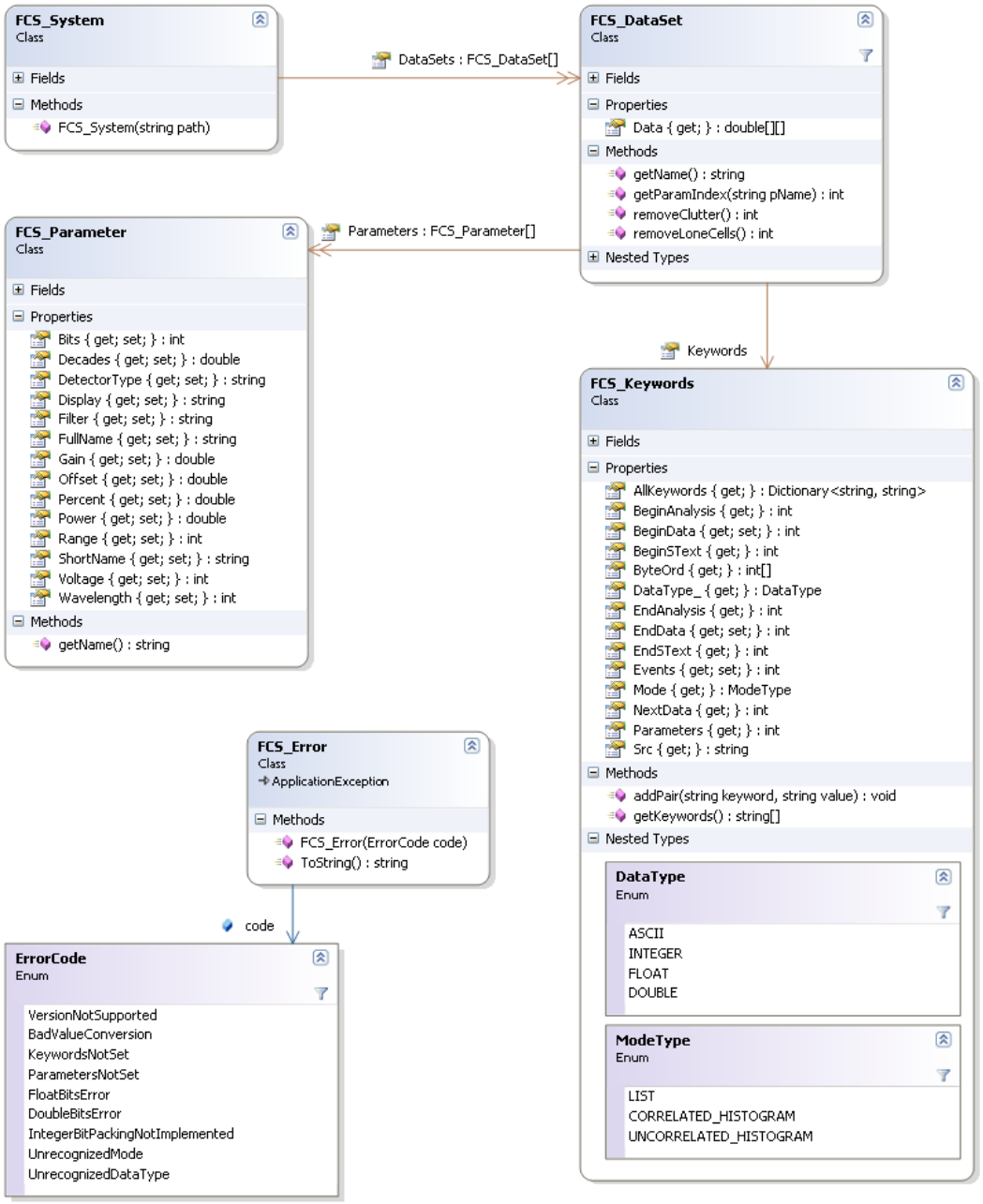
- `INTEGER` - wartości zapisane są w postaci liczb całkowitych,
- `FLOAT` - wartości zapisane w postaci liczb zmiennoprzecinkowych pojedynczej precyzji,
- `DOUBLE` - wartości zapisane w postaci liczb zmiennoprzecinkowych podwójnej precyzji.

Właściwość `Mode` może przyjąć jedną z poniższych wartości:

- `LIST` - w segmencie `DATA` dane zapisane są w postaci listy kolejnych zdarzeń, a każde zdarzenie opisane jest odczytanymi wartościami kolejnych parametrów. Odnosząc tą formę do klasy `FCS_Dataset` można powiedzieć, że dane zapisane są wierszami tablicy `Data`,
- `CORRELATED_HISTOGRAM` - w segmencie `DATA` dane zapisane są w następujące sposób - odczytane wartości parametru pierwszego dla każdego zdarzenia po kolei, następnie wartości parametru drugiego dla wszystkich zdarzeń itd. dla wszystkich parametrów. Odnosząc tą formę do klasy `FCS_Dataset` można powiedzieć, że dane zapisane są kolumnami tablicy `Data`,
- `UNCORRELATED_HISTOGRAM` dane w segmencie `Data` zapisane są po pewnym przetworzeniu w cytofluorymetrze. Dla każdego parametru zapisany jest histogram rozkładu liczby wystąpień zdarzeń według tego parametru.

Tryb zapisu `UNCORRELATED_HISTOGRAM` nie pozwala na odtworzenie poszczególnych zdarzeń, a w konsekwencji nie pozwala na wykonywaną przez program analizę. Próba odczytu takich plików spowoduje zgłoszenie błędu przez aplikację. Obecnie standardem wśród producentów cytofluorymetrów jest zapisywanie danych w pliku w trybach `FLOAT` i `LIST`.

Znaczenie poszczególnych słów kluczowych i parametrów opisane jest w standardzie formatu *FCS* [SBB+96].



Rysunek 5.1. Diagram klas reprezentujących plik formatu FCS.

5.2 Klasy wykorzystywane w analizie

Moduł do analizy skupień został zaprojektowany tak, aby w łatwy sposób można było go rozwijać. Jak już wspomniano we wstępie teoretycznym, analiza skupień obejmuje różne metody, każda z nich może opierać się na innych metrykach. Niezbędne jest zatem umożliwienie dodawania nowych klas implementujących kolejne definicje odległości i algorytmy. Wszystkie klasy znajdują się w bibliotece ClusterAnalysis.

Podstawową jednostką danych przetwarzaną przez klasy w tej bibliotece jest zdarzenie, reprezentowane jako tablica typu double. Dla oznaczenia zdarzenia zastosowano termin *obiekt* i jest on stosowany w dalszym opisie biblioteki.

W bibliotece ClusterAnalysis obiekt jest reprezentowany jako tablica typu `double`, gdzie pod

indeksem 0 znajduje się numer zdarzenia w tablicy `Data` analizowanego obiektu klasy `FCS_DataSet`. Pozostałe elementy tablicy zawierają wartości parametrów branych pod uwagę przy podziale zdarzeń na skupienia.

Klasą, na której operują wszystkie klasy implementujące poszczególne metody analizy skupień, jest klasa `Cluster` przedstawiona na rysunku 5.2. Reprezentuje poszczególne skupienia i dostarcza metody wykorzystywane przy ich analizie. Klasa posiada następujące własności:

- `Objects` - lista obiektów (zdarzeń) znajdujących się w skupieniu,
- `ScoringEntries` - słownik, którego każdy element składa się z klucza (nazwa antygenu) i wartości (liczba zdarzeń związanych z dodatnią ekspresją tego antygenu),
- `Type` - typ skupienia ustalany podczas rozpoznawania skupień, może przyjmować poniższe wartości:
  - `Limf` - limfocyty,
  - `Mono` - monocyty,
  - `Neutr` - neutrofile,
  - `Other` - inne,
- `TypeName` - nazwa typu opisanego w poprzednim punkcie, w zależności od wartości Type może przyjmować wartości - limfocyty, monocyty, neutrofile, inny.

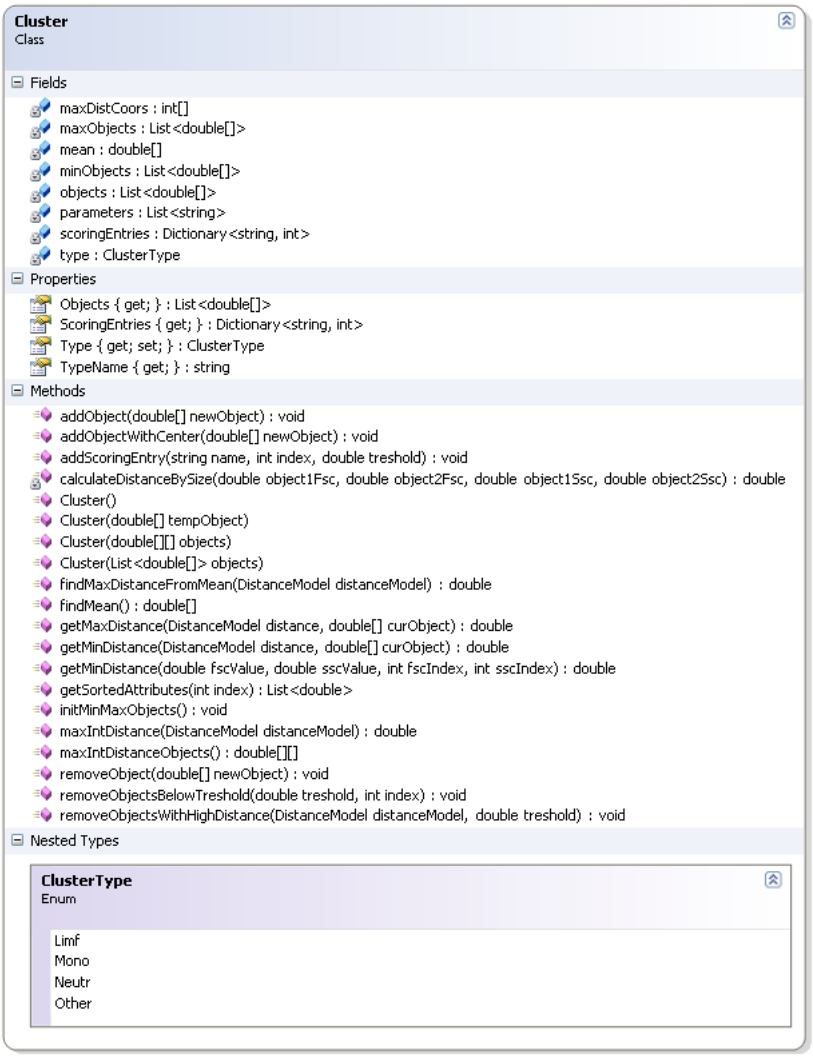
Klasa `Cluster` dostarcza wiele metod wykorzystywanych w analizie. Najważniejsze z nich są:

- metody do dodawania i usuwania pojedynczych obiektów ze skupień (`addObject`, `removeObject`),
- metody do obliczania odległości zadanego obiektu od skupienia (`getMaxDistance`, `getMinDistance`, `getDistanceFromMean` - odległość od środka skupienia)
- metoda znajdująca środek skupienia (`findMean`),
- metoda znajdująca najbardziej odległe obiekty w skupieniu (`maxIntDistanceObjects`).

Każda metoda obliczająca odległości pobiera jako parametr obiekt klasy implementującej interfejs `DistanceModel`. Interfejs deklaruje jedną metodę - `calculate`, której zadaniem jest obliczenie odległości między dwoma podanymi obiektami. W projekcie zaimplementowano jedynie model odległości euklidesowej reprezentowany przez klasę `EuclidesDistanceModel`.

Za podział obiektów na grupy odpowiadają klasy dziedziczące po abstrakcyjnej klasie `Analyzer` (rysunek 5.3). Każda klasa dziedzicząca po klasie `Analyzer` powinna implementować jedną z dwóch wersji metody `analyse`. Dwa pierwsze z przyjmowanych parametrów są takie same dla obu metod, różnica leży w trzecim. Pierwsza metoda przyjmuje początkową listę skupień, a druga tablicę obiektów do podziału na skupienia. Poniżej zaprezentowano parametry jakie przyjmują obie metody:

- `distanceModel` - obiekt opisujący model odległości wykorzystywany w analizie (realizujący interfejs `DistanceModel`),
- `clusterCount` typu `int` - docelowa liczba skupień,
- w zależności od sposobu wykorzystania:
  - `clusters` - początkowa lista skupień,



Rysunek 5.2. Klasa `Cluster`.

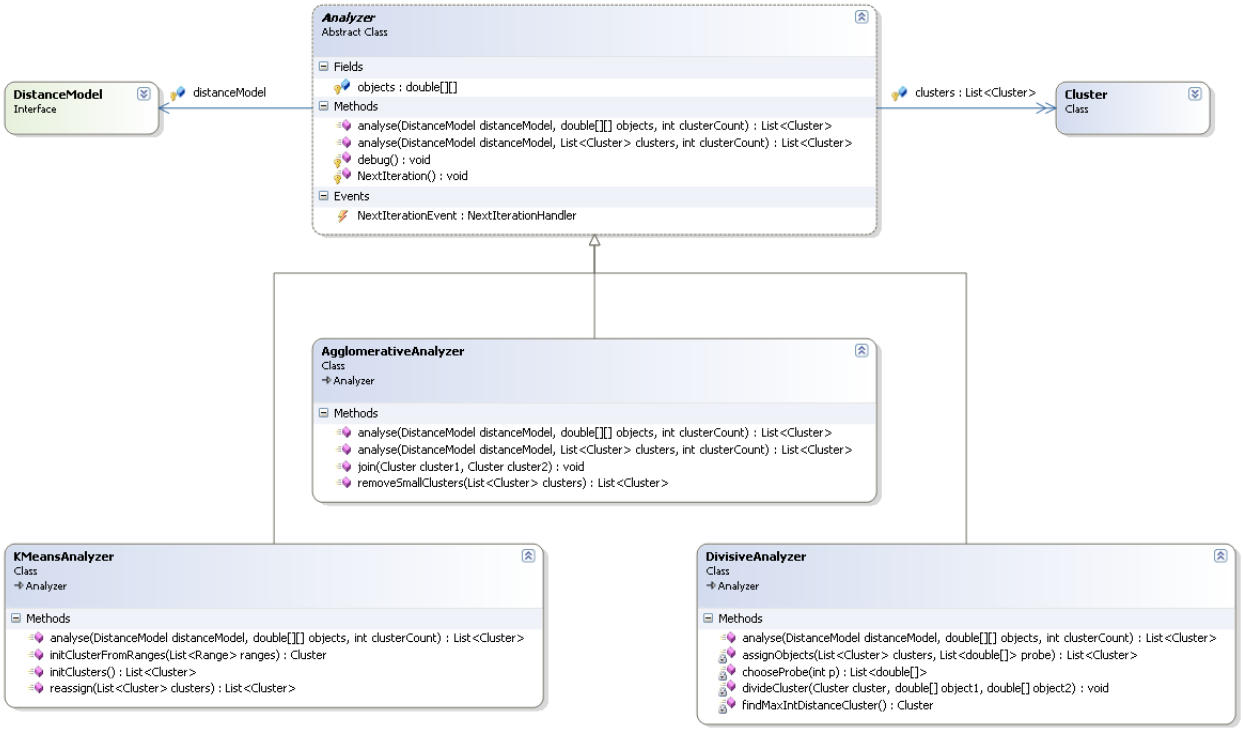
– `objects` - tablica obiektów do podziału na grupy.

W projekcie zaimplementowano następujące klasy odpowiedzialne za podział analizowanych obiektów na skupienia:

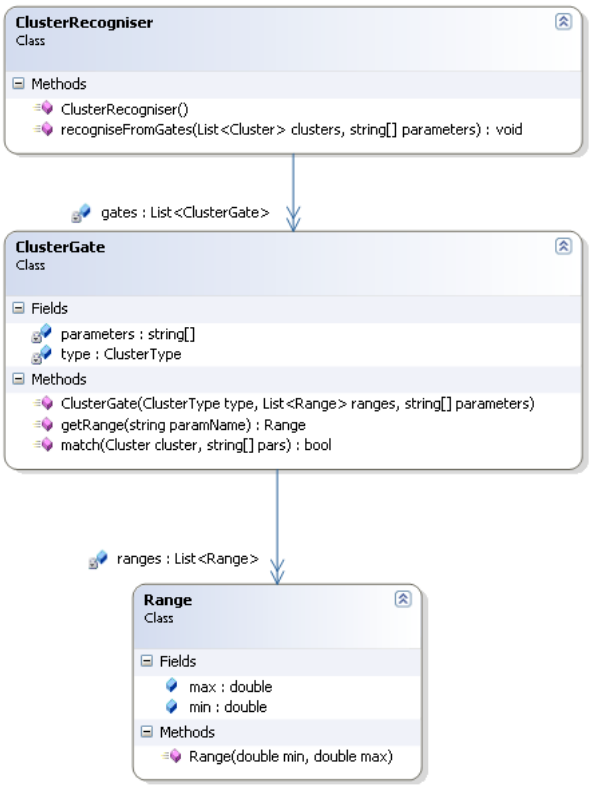
- `DivisiveAnalyzer` - klasa implementująca algorytm deglomeracyjny,
- `AgglomerativeAnalyzer` - klasa implementująca algorytm aglomeracyjny,
- `KMeansAnalyzer` - klasa implementująca algorytm k-średnich.

Ostatnia grupa klas wykorzystywanych w analizie skupień służy do rozpoznawania rodzaju skupienia (rysunek 5.4).

Klasą odpowiedzialną za rozpoznawanie typu skupienia jest klasa `ClusterRecogniser`. Proces rozpoznawania skupień zaimplementowany jest w metodzie `recogniseFromGates` i korzysta ze zdefiniowanych bramek dla każdego rozpoznawanego typu skupienia (limfocyty, monocyty i neutrofile). Pojedyncza bramka opisuje przedziały wartości na każdym parametrze dla poszczególnych subpopulacji. Definicja bramki reprezentowana jest przez klasę `ClusterGate` i składają się z typu skupienia, listy parametrów i przedziałów wartości dla każdego parametru (klasa `Range`).



Rysunek 5.3. Schemat klas implementujących algorytmy analizy skupień.



Rysunek 5.4. Schemat klas służących do rozpoznania skupienia.

5.3 Baza danych

Stworzony system może współpracować z dowolną bazą danych, do której istnieje dostęp za pomocą mechanizmu *ADO.NET*. Za komunikację między aplikacją a bazą danych odpowiadają klasy przedstawione na diagramie 5.5.

Klasa `Connection` odpowiada za bezpośredni dostęp aplikacji do bazy danych - wykonywanie zapytań, wywoływanie składowanych procedur i funkcji oraz zapis wyników zapytania do obiektów typu `DataSet`. Umożliwia ona wyszukiwanie pacjentów, lekarzy i pracowników laboratoryjnych, odnajdywanie wyników badania dla konkretnego pacjenta oraz ich aktualizację.

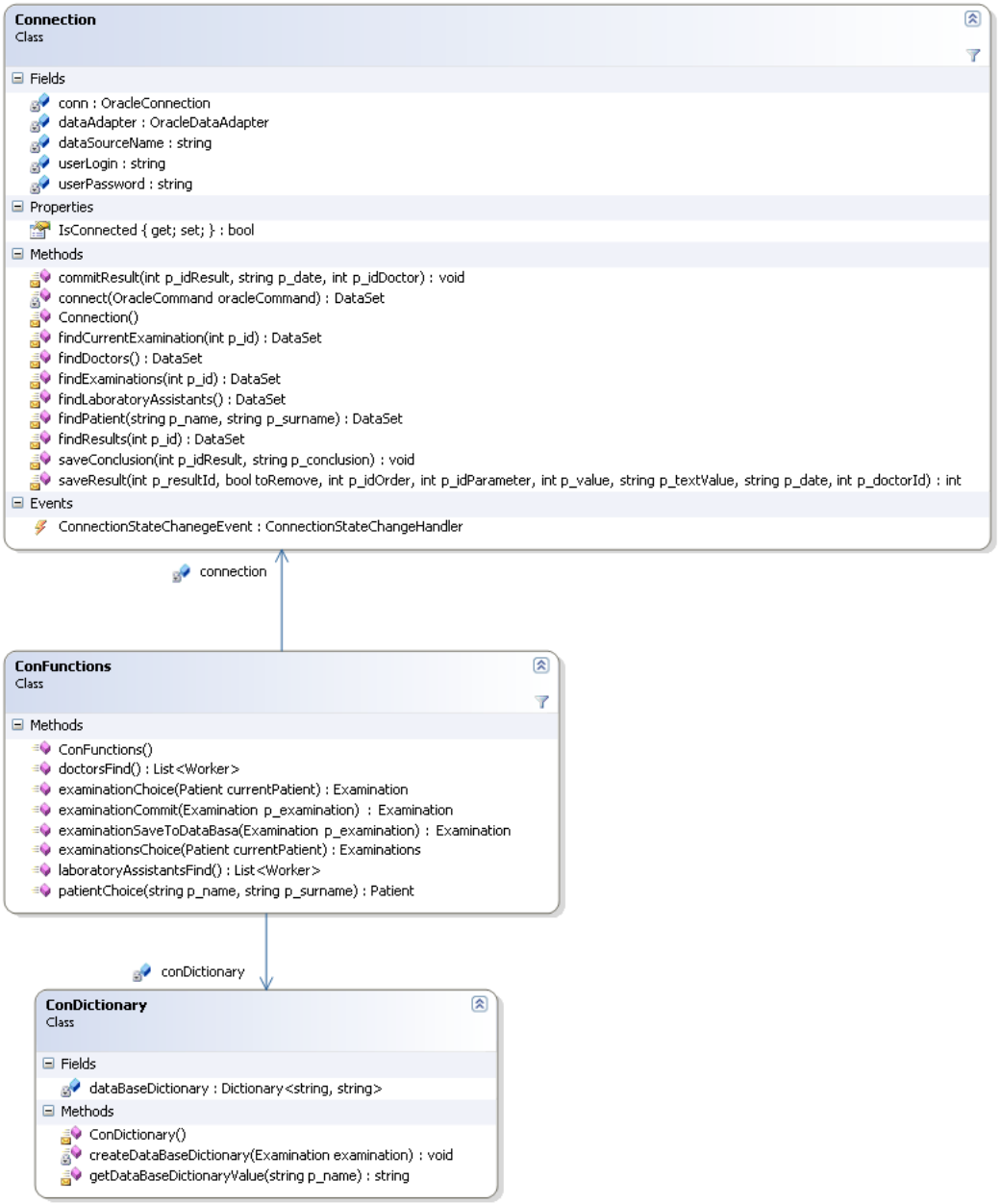
Dane z obiektów `DataSet` są następnie przetwarzane przy użyciu metod klasy `ConFunctions` do obiektów klas zaimplementowanych w programie w przestrzeni nazw `DataModel`. Klasy te zaprezentowano na rysunku 5.6. Ponieważ nazwy konkretnych wartości z badania obecne w bazie danych mogą się różnić od nazw stosowanych w aplikacji stworzono klasę `ConDictionary`, która umożliwia właściwe przyporządkowanie wartości do obiektów w aplikacji.

Sposób implementacji tej części projektu pozwala na łatwe i szybkie przystosowanie aplikacji do współpracy z różnymi bazami danych.

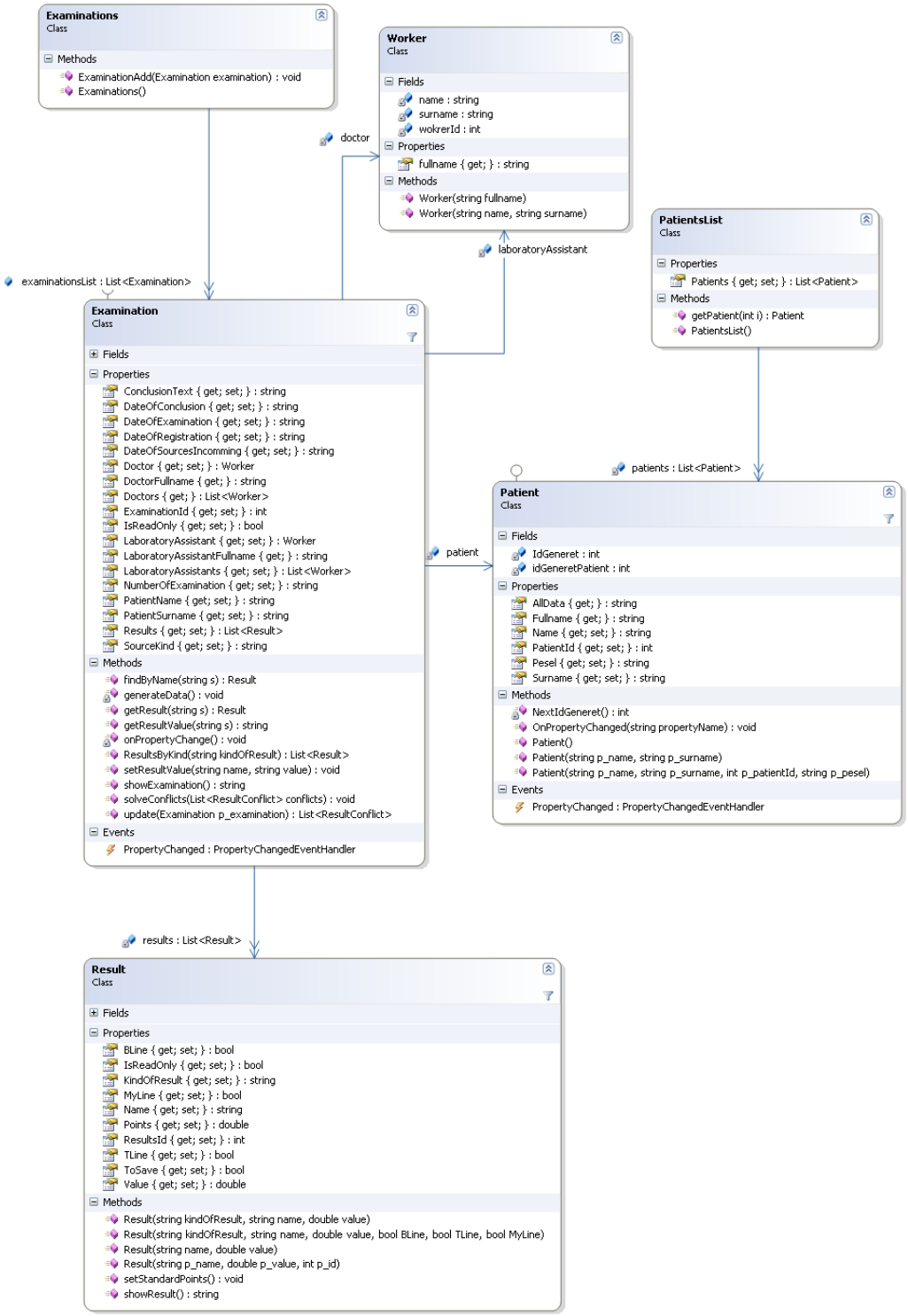
W ramach wdrożenia systemu odpowiednie metody klasy `Connection` zaimplementowana tak, by umożliwić współpracę z obecną w Zakładzie bazą danych Oracle 10g. Baza ta jest częścią systemu KS-SOLAB firmy KAMSOFT [4]. Po nawiązaniu kontaktu z firmą uzyskano zgodę na dostęp do bazy poprzez specjalnie utworzonego użytkownika oraz widoki i funkcje. Nadane uprawnienia odpowiadają potrzebom stworzonego systemu [5].

[4] "KS-SOLAB - Zintegrowany System Zarządzania Laboratorium", `http://www.kamsoft.pl/prod/solab/index.htm`.

[5] W czasie testów koniecznych przy implementacji korzystano z wersji demonstracyjnej systemu KS-SOLAB udostępnionej przez KAMSOFT na czas trwania projektu.



Rysunek 5.5. Diagram klas odpowiadających za komunikację z bazą danych.



Rysunek 5.6. Diagram klas w przestrzeni nazw `DataModel`.

Rozdział 6

Działanie aplikacji

6.1 Odczyt danych

Pierwszym etapem pracy z programem jest odczytanie wyników badania. Do tego celu służy kontrolka o nazwie Drzewko Katalogów przedstawiona na rysunku 6.1. Katalogi wyróżnione pomarańczową ikoną zawierają w sobie pliki formatu *FCS*. Wybranie takiego katalogu powoduje odczyt tych plików oraz pojawienie się w drzewie pozycji odpowiadających pacjentom oraz wszystkim blokom danych zawartych w plikach z danymi poszczególnych pacjentów. Wybranie pacjenta powoduje wywołanie kreatora analizy opisanego w następnej sekcji, natomiast wybranie bloku danych umożliwia eksport danych do pliku formatu *CSV*(ang. *Comma Separated Values*).

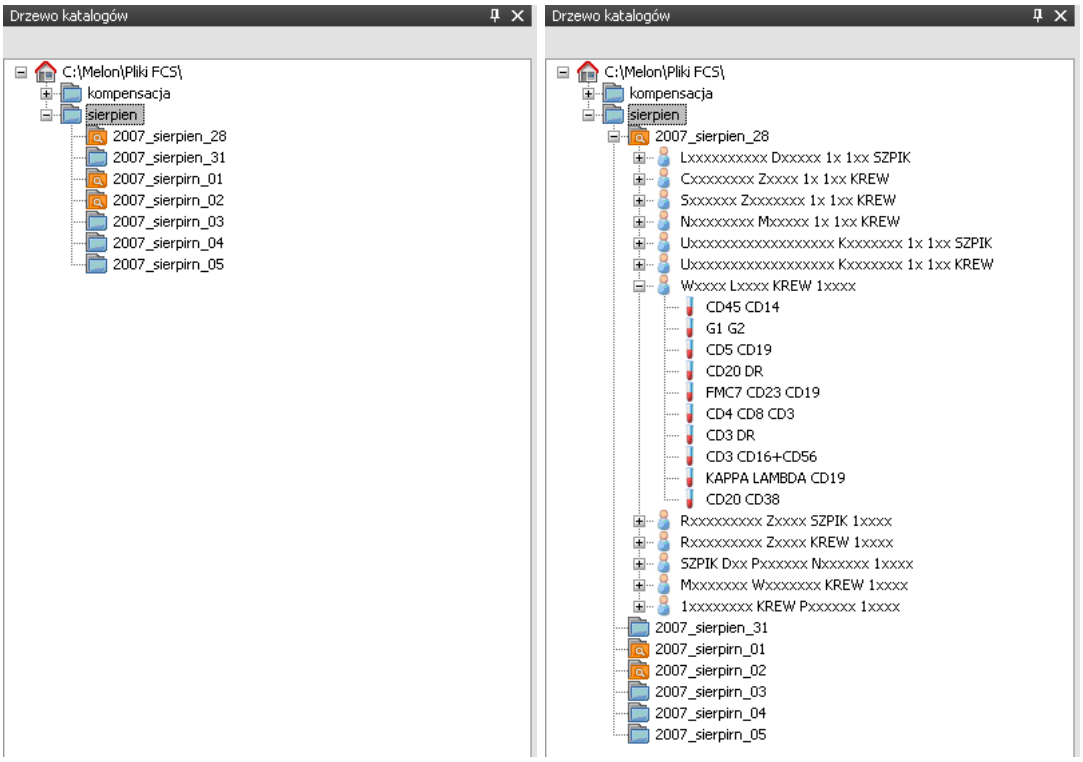
Przeprowadzenie badania dla jednego pacjenta wymaga zazwyczaj analizy wielu próbek dostępnego materiału. Powoduje to zapisanie przez cytofluorymetr wielu plików *FCS* i w konsekwencji prowadzi do analizy wielu plików przez oprogramowanie.

Kiedy użytkownik uruchamia proces wczytywania plików w katalogu wykonywany jest algorytm 1.

LOA 1: Odczyt plików z katalogu

1: odczytaj wszystkie pliki *FCS* w katalogu
2: **for all** odczytany blok danych **do**
3: **if** blok zawiera macierz kompensacji **then**
4: popraw zmierzone świecenie komórek zgodnie ze wzorem 2.1
5: **end if**
6: **end for**
7: pogrupuj bloki danych z plików na podstawie parametru `Src` // *Badania jednego pacjenta mają identyczna wartość tego parametru*
8: **for all** pacjent **do**
9: dodaj pacjenta do drzewka
10: dodaj bloki danych pacjenta do drzewka // *Jako nazwę bloku danych przyjmuje się nazwy antygenów zmierzonych w bloku*
11: **end for**

Struktura drzewka nie odpowiada dokładnie strukturze danych na dysku - w jednym katalogu może znajdować się wiele plików z wynikami badań różnych pacjentów. Drzewko Katalogów pozwala natomiast na uporządkowaną prezentacje danych poprzez grupowanie wyników badań należących do poszczególnych pacjentów – wprowadzony zostaje zatem dodatkowy poziom porządkowania.



Rysunek 6.1. Drzewko katalogów. Po prawej kontrolka z odczytanymi plikami z jednego katalogu.

6.2 Analiza

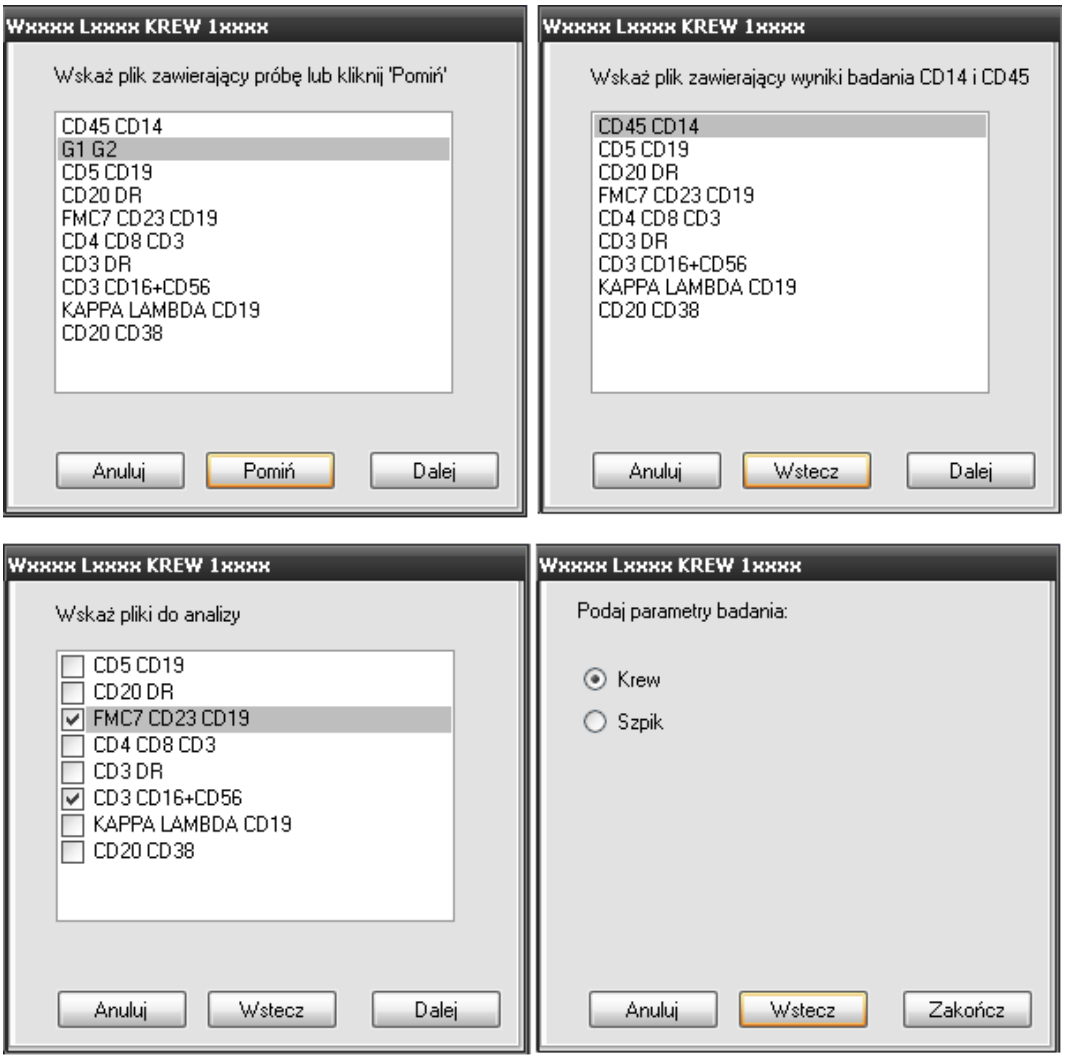
Aplikacja dzieli komórki na subpopulacje ze względu na typ komórki. Następnie dla wyznaczonych grup (subpopulacji) określa obraz immunofenotypowy, czyli charakteryzuje je poprzez parametry odpowiadające występowaniu badanych antygenów.

Podziału komórek dokonuje się na podstawie bloku danych zawierającego zmierzoną ekspresję antygenów CD45 i CD14. CD45 to antygen obecny na wszystkich leukocytach (*LCA*,ang. *leukocyte common antigen*). CD14 jest natomiast obecny na komórkach subpopulacji monocytów. Przeciwciała przeciw tym antygenom stosowane są często w tej samej próbce - oznaczonej *LG* (ang. *leuko gate*). Wyniki z niej uzyskane stosuje się do wyznaczania subpopulacji komórkowych.

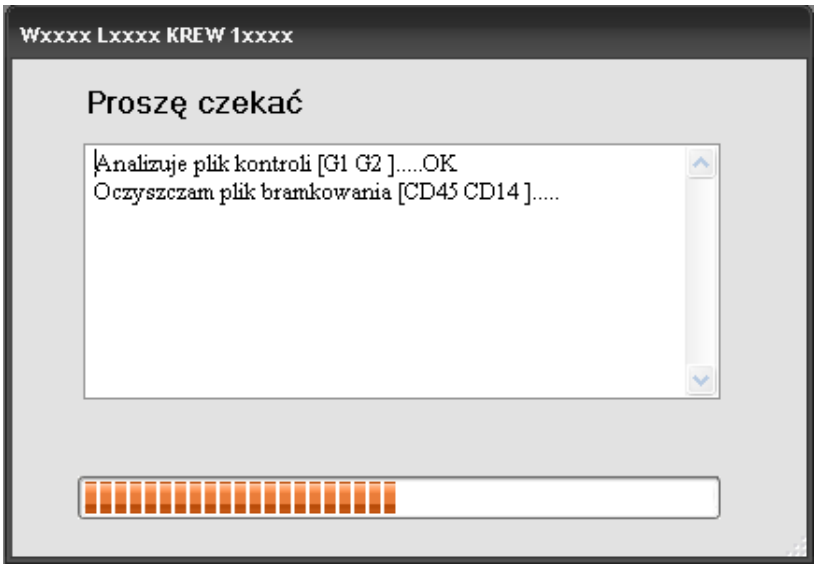
Kiedy użytkownik wybiera pacjenta do analizy wywołany zostaje kreator. Umożliwia on w kolejnych krokach:

1. wybór bloku danych zawierającego dane kontroli ujemnej (opcjonalnie),
2. wybór bloku danych będącego podstawą do podzielenia komórek na subpopulacje,
3. wybór bloków danych zawierających wyniki pomiarów antygenów, których ekspresję badamy,
4. określenie czy pliki pochodzą z badania krwi czy szpiku.

Kolejne kroki pracy z kreatorem przedstawione są na rysunku 6.2. Po wybraniu przycisku `Zakończ` rozpoczyna się analiza. Podczas przetwarzania danych użytkownik widzi okienko przedstawione na rysunku 6.3 informujące go o postępie obliczeń. Po zakończeniu analizy otwarte zostają dwa okienka: jedno z wykresami, drugie zawierające wypełniony formularz wyników badania.



Rysunek 6.2. Kreator.



Rysunek 6.3. Analiza.

Kontrola służy do obliczania wartości progowych dla wszystkich zmierzonych parametrów fluorescencji komórki. Wartości mierzone przez detektory w czasie kontroli skupione są wokół pewnych wartości. Komórki uszkodzone oraz odczytane artefakty nie będące w ogóle komórkami świecą zupełnie inaczej, dając dużo mniejsze lub dużo większe wartości. Standardowo, próg dla każdego detektora wyznacza się tak, żeby 5% zdarzeń z kontroli było większych od niego. W przypadku braku bloku kontroli, jako wartość progową przyjmuje się 10^3. Podział komórek na subpopulacje wykonywany jest na podstawie danych z badania antygenów CD45 i CD14. Przed grupowaniem komórek wykonywane jest oczyszczanie bloku danych, które przebiega zgodnie z algorytmem 2:

LOA 2: Oczyszczanie komórek z badania CD45/CD14

```
1: for all zdarzenie do
2:   if zdarzenie mniejsza niż minimalny rozmiar komórki then
3:     odrzuć // uznajemy zdarzenie za artefakt
4:   else if zdarzenie o bardzo dużej gęstości i małym rozmiarze then
5:     odrzuć
6:   else if zdarzenie ma ziarnistość lub rozmiar równy maksimum skali then
7:     odrzuć
8:   else
9:     zostaw
10:  end if
11: end for
12: for all zdarzenia pozostałe po poprzednim kroku do
13:   oblicz odległość od pozostałych zdarzeń // w przestrzeni dwuwymiarowej wyznaczonej przez
      parametry ziarnistości i wielkości
14:   if najbliższe zdarzenie jest za daleko then
15:     odrzuć
16:   else
17:     zachowaj // uznajemy zdarzenie za komórkę
18:   end if
19: end for
```

Komórki krwi wywodzące się z tej samej linii komórkowej wykazują podobne cechy, w tym wielkość i obecność cytoplazmatycznych ziarnistości. W szczególności oznacza to, że w dowolnym badaniu dla komórek jednego typu (również nowotworowych) wyznaczone parametry wielkości i ziarnistości nie będą różnić się w sposób istotny. Jeżeli odczytany artefakt jest daleko od innych, to znaczy że najprawdopodobniej nie jest prawidłową komórką i można go odrzucić.

Następnym krokiem po oczyszczeniu zbioru danych jest podział komórek na subpopulacje. Do tego celu zastosowano wspomniane w rozdziale trzecim metody analizy skupień. Etap rozpoczyna się od przygotowania danych do analizy. Subpopulacje komórek rozróżniamy uwzględniając parametry: wielkość, ziarnistość, ekspresję antygenu CD14 oraz ekspresję antygenu CD45.

Takie rozwiązanie stosujemy ze względu na rutynowe oznaczanie wymienionych antygenów w badaniach [w diagnostyce rozrostów nowotworowych] w jednej z próbek[1] oraz stałą obecność parametrów wielkość i ziarnistość w każdym pliku *FCS*. Zapewnia to możliwość uzyskania stosunkowo wielu informacji przy wykorzystaniu jedynie dwóch rodzajów przeciwciał.

U zdrowego pacjenta wśród leukocytów krwi obwodowej można wyróżnić trzy duże subpopulacje: neutrofile, limfocyty i monocyty. Natomiast w populacji leukocytów pochodzącej ze szpiku kostnego dominują komórki linii mieloidalnej (z której wywodzą się m.in. neutrofile), limfoidalnej i monocytoidalnej[2]. Suma komórek należących do tych dominujących subpopulacji powinna być zbliżona do odsetka komórek CD45 dodatnich (wykazujących ekspresję antygenu CD45 na swojej

[1] Przyjęto, że termin *próbka* oznacza każdą próbkę analityczną pochodzącą z dostarczonego do badania materiału biologicznego.

[2] W aplikacji przyjęto - naśladując używane obecnie w Zakładzie oznaczenia - takie same nazwy dla poszczegól-

LOA 3: Algorytm deglomeracyjny

```
przydziel wszystkie obiekty do jednego skupienia
while liczba skupień jest mniejsza od docelowej do
  for all skupienie do
    znajdź parę najbardziej odległych obiektów w skupieniu
    if odległość między wybranymi obiektami jest większa niż zapamiętana odległość maksymalna then
      zapamiętaj numer skupienia, obiekty i odległość między nimi jako odległość maksymalną
    end if
  end for
  stwórz dwa nowe skupienia i przydziel do każdego z nich po jednym z dwóch obiektów wybranych w poprzednim etapie
  for all obiekt w skupieniu do podziału do
    oblicz odległość obiektu od środków nowych skupień i przydziel obiekt do tego, dla którego odległość jest mniejsza
  end for
end while
```

powierzchni). Pojawienie się większej grupy leukocytów nienależących do wymienionych subpopulacji może świadczyć o procesie nowotworowym.

Specyfika zjawiska jakim jest fluorescencja sprawia, że ze względu na lepsze zobrazowanie różnic między poszczególnymi odczytami, zaleca się logarytmowanie wszelkich danych opisywanych przez świecenie fluorochromu.

W kolejnym kroku należy dokonać normalizacji by zrównoważyć wpływ każdego z uwzględnionych parametrów. Do tego celu wykorzystano standaryzację. Procedurze tej poddano wszystkie uwzględnione w analizie atrybuty (parametry komórek).

Tak przygotowane dane mogą zostać poddane dalszej analizie. Z opisanych algorytmów analizy skupień każdy ma pewne wady i zalety. Ze względu na złożoność obliczeniową najciekawszy wydaje się algorytm k-średnich. Wadą tego podejścia jest jednak odgórne założenie o znanej docelowej liczbie skupień. Ze względu na możliwość wystąpienia dodatkowych subpopulacji, nie jest wskazane stosowanie tego algorytmu.

Algorytmy aglomeracyjne wydają się być mało przydatne w analizie ze względu na stosunkowo dużą liczbę obiektów i małą liczbę docelowych skupień. Oznacza to ponad 5000 iteracji algorytmu, co wydaje się być liczbą zdecydowanie zbyt dużą.

Przesłanki przemawiające za odrzuceniem algorytmów aglomeracyjnych są jednocześnie przesłankami przemawiającymi za wykorzystaniem algorytmów deglomeracyjnych. Docelowa liczba skupień powinna zamknąć się w kliku - kilkunastu. Wadą tego podejścia jest dość specyficzne dzielenie poszczególnych skupień. Może się zdarzyć, że jedno wielkie skupienie zostanie podzielone dokładnie w połowie. Nadal aktualny pozostaje również problem wyboru docelowej liczby skupień.

Oba problemy można jednak w dość łatwy sposób rozwiązać. Zamiast dzielić zbiór obiektów na oczekiwaną liczbę skupień, można go podzielić na większą liczbę skupień, a następnie połączyć te znajdujące się blisko siebie. Do tego celu można wykorzystać np. algorytmy aglomeracyjne, jednak w projekcie zrezygnowano z łączenia skupień w taki sposób. W zamian każde z uzyskanych skupień w następnym etapie jest klasyfikowane, a skupienia są łączone na podstawie wyników klasyfikacji.

Algorytm został zaprezentowany na schemacie 3.

Dość duża złożoność obliczeniowa wyszukiwania pary najdalszych obiektów w skupieniu ($O(n^2)$,

nych subpopulacji pochodzących ze szpiku kostnego i z krwi. Jednak pod pojęciem `neutrofile`, `limfocyt` i `monocyty` przy ocenie szpiku należy rozumieć całą linię rozwojową, z której wywodzą się te komórki.

LOA 4: Obliczanie maksymalnej odległości w skupieniu

```
1: for all uwzględniany w analizie atrybut do
2:   posortuj obiekty względem tego atrybutu
3:   do zbioru potencjalnie małych obiektów dodaj t obiektów o najmniejszej wartości atrybutu
4:   do zbioru potencjalnie dużych obiektów dodaj t obiektów o największej wartości atrybutu
5: end for
6: for all obiekt ze zbioru potencjalnie małych obiektów do
7:   for all obiekt ze zbioru potencjalnie dużych obiektów do
8:     if odległość między wybranymi obiektami jest większa od odległości między obiektami
       pamiętanymi jako najbardziej odległe then
9:       zapamiętaj obiekty jako najbardziej odległe
10:    end if
11:  end for
12: end for
```

Nazwa	wielkość		ziarnistość		CD14		CD45	
	min	max	min	max	min	max	min	max
Limfocyty	40000	210000	0	55000	$-\infty$	8000	1600	∞
Monocyty	110000	∞	30000	110000	2000	100000	250	16000
Neutrofile	70000	250000	100000	∞	-600	2500	1000	10000

TABLICA 6.1: Wartości analizowanych atrybutów w populacjach limfocytów, monocytów i neutrofili

gdzie n jest liczbą obiektów w skupieniu), sprawiała, że algorytm ten dla przeciętnej liczby analizowanych obiektów wykonywał się w ciągu kilkunastu minut, co w zauważalny sposób wpływało na długość całej analizy . Dlatego do obliczania tej odległości zastosowano heurystykę przedstawioną na schemacie 4.

Uzyskane w ten sposób skupienia podlegają procedurze rozpoznawania typu subpopulacji (klasyfikacji) na podstawie oczekiwanych wartości atrybutów dla każdej z subpopulacji. Wybrane wartości przedstawia tabela 6.1.

Procedura klasyfikacji sprowadza się do sprawdzenia do której z podanych subpopulacji należy centroid skupienia. Jeśli okaże się, że centroid nie należy do żadnej z podanych subpopulacji, skupienie klasyfikowane jest jako "Inne". Następnie skupienia tego samego typu są łączone w jedno skupienie. W ten sposób uzyskane populacje poddawane są kolejnej procedurze oczyszczania. Tym razem ze skupień usuwane są obiekty, które znajdują się daleko od środków uzyskanych skupień. Próg dla odległości, powyżej której obiekt jest usuwany, stanowi 70% odległości najdalszego obiektu od centroidu.

Jak już wspomniano wcześniej, subpopulacje krwi jednego typu będą cechować się podobną ziarnistością i wielkością. Dlatego też przydział komórek do subpopulacji w pozostałych analizowanych blokach danych odbywa się na podstawie tych wyznaczonych w badaniu CD45/CD14. Metodę przydziału opisano w algorytmie 5.

LOA 5: Przydzielanie komórek do subpopulacji

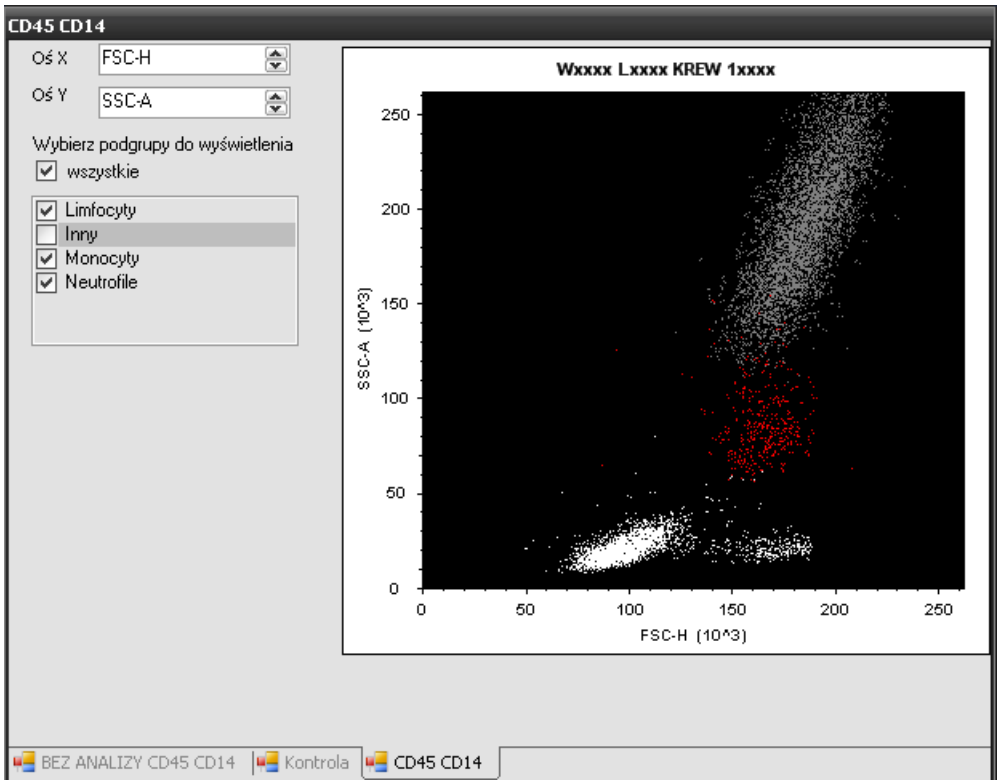
```
1: for all zdarzenie do
2:   minimal_distance ← ∞
3:   for all subpopulacja do
4:     wyznacz distance - odległość zdarzenia od najbliższej komórki z subpopulacji
5:     if distance < minimal_distance then
6:       minimal_distance ← distance
7:     end if
8:   end for
9:   if zdarzenie jest wystarczająco blisko najbliższej subpopulacji then
10:    przydziel zdarzenie do subpopulacji // komórka
11:  else
12:    odrzuć // artefakt
13:  end if
14: end for
```

Dla każdego analizowanego bloku danych (czyli wybranego w kroku 2 i 3 kreatora) i wykrytych w nim subpopulacji przeprowadzane jest obliczanie ilości komórek dodatnich względem badanych antygenów. Komórkę uznajemy za dodatnią, jeżeli wartość zmierzonego świecenia związanego z antygenem jest większa niż próg wyznaczony z kontroli.

Jeżeli materiałem do analizowanego badania jest szpik, to jako wyniki przyjmuje się procentowy udział komórek dodatnich wśród całej populacji. W badaniu krwi postępujemy podobnie, z tym że dla antygenów linii limfocytarnej określamy udział komórek o dodatniej ekspresji w subpopulacji limfocytów. Otrzymane wyniki zostają zapisane w oknie wyników badania.

6.3 Wizualizacja danych

Okno wykresów umożliwia użytkownikowi wizualizacje wyników analizy. Komórki różnego typu są na wykresach zaznaczone innym kolorami. Możliwe jest ukrywanie poszczególnych subpopulacji. W zależności od parametru detektora `Display` wartości na osiach przedstawione są w skali logarytmicznej bądź liniowej. Wykres można wyeksportować do pliku graficznego, wydrukować lub skopiować do schowka. Przykład okna z wykresem znajduje się na rysunku 6.4.



Rysunek 6.4. Przykładowy wykres przedstawiający ziarnistość i wielkość komórek.

6.4 Wyniki badania

Po wykonaniu całej analizy, uzyskane wyniki prezentowane są w oknie `Wyniki`. Okno to składa się z następujących części:

- `badanie` - dane dotyczące danego badania (numer badania, imię i nazwisko pacjenta, nazwisko osoby wykonującej badanie i data wykonania badania, nazwisko osoby oceniającej badanie i data oceny oraz materiał i data jego pozyskania),
- `morfologia` - zawartość procentowa poszczególnych subpopulacji (limfocyty, monocyty, neutrofile, bazofile, eozynofile, blasty),
- `wniosek` - wniosek wpisany przez lekarza oceniającego badanie,
- `cytometria` - wyniki badania immunofenotypowego ukazane w widoku standardowym (podobnym do obecnie używanych w Zakładzie szablonów, które uzupełnia się w trakcie oceny badania) - rysunek 6.5
- `scoring` - wyniki badania immunofenotypowego ukazane w specjalnym widoku ułatwiającym diagnostykę białaczek bifenotypowych - rysunek 6.6

Widoki `cytometria` i `scoring` prezentują te same dane, dlatego zmiana wartości w jednym z tych widoków pociąga za sobą automatyczną zmianę w drugim.

Poprzez okno `Wyniki` możliwa jest również komunikacja z bazą danych. Korzystając z menu `Baza danych` można:

- pobrać z bazy danych aktualne badanie (na podstawie wpisanego numeru lub danych osobowych pacjenta),

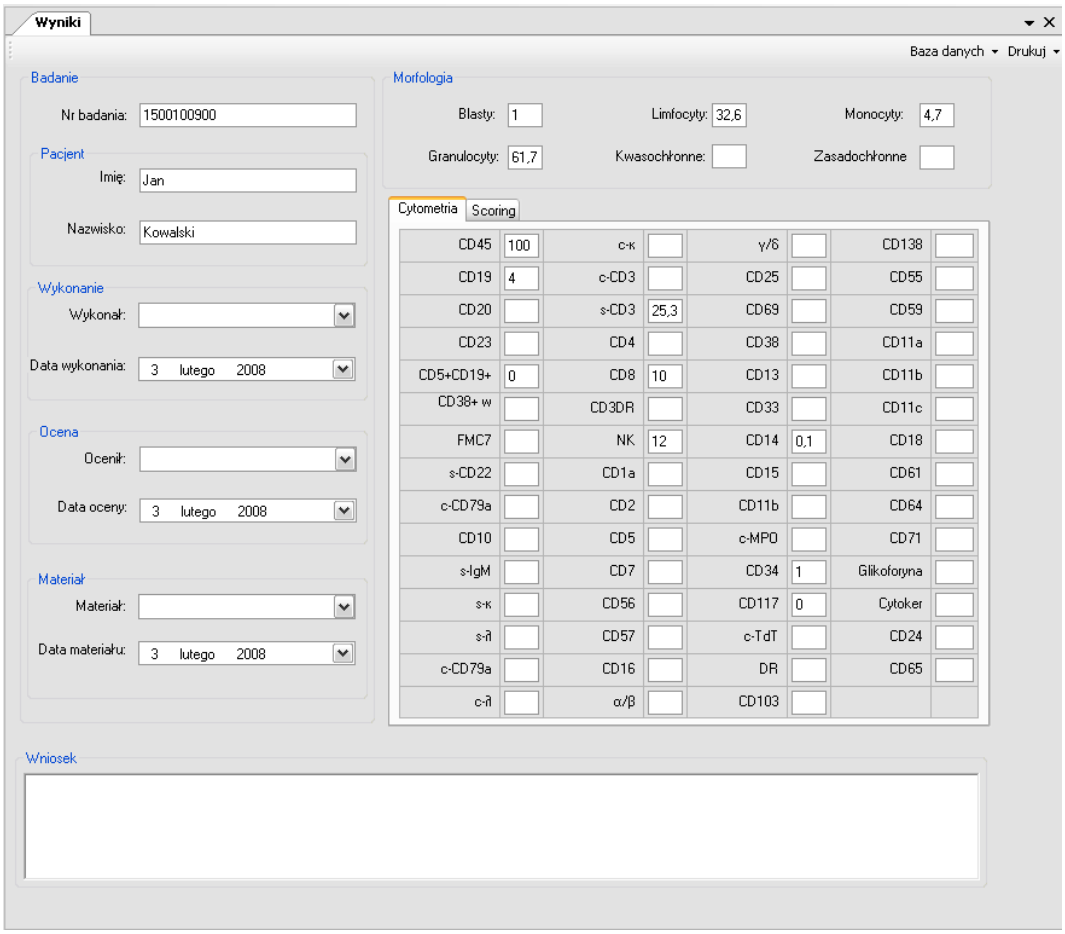
- pobrać z bazy poprzednie badania pacjenta (na podstawie danych osobowych pacjenta),
- zapisać aktualnie wyświetlane dane do bazy danych (jeśli wcześniej nastąpiło skojarzenie badania z obecnym w bazie danych zleceniem).

W przypadku, gdy w bazie danych znajduje się więcej niż jeden pacjent o podanym imieniu i/lub nazwisku, wyświetlane jest okienko z listą pacjentów a dopiero po wyborze konkretnego z nich, wyświetlane są wyniki badań. Podobnie, gdy w wyniku zapytania otrzymamy więcej niż jedno aktualne zlecenie dla danego pacjenta (np. w przypadku, gdy został on skierowany jednocześnie na badanie krwi i szpiku), użytkownik powinien wskazać, które dokładnie badanie chce edytować.

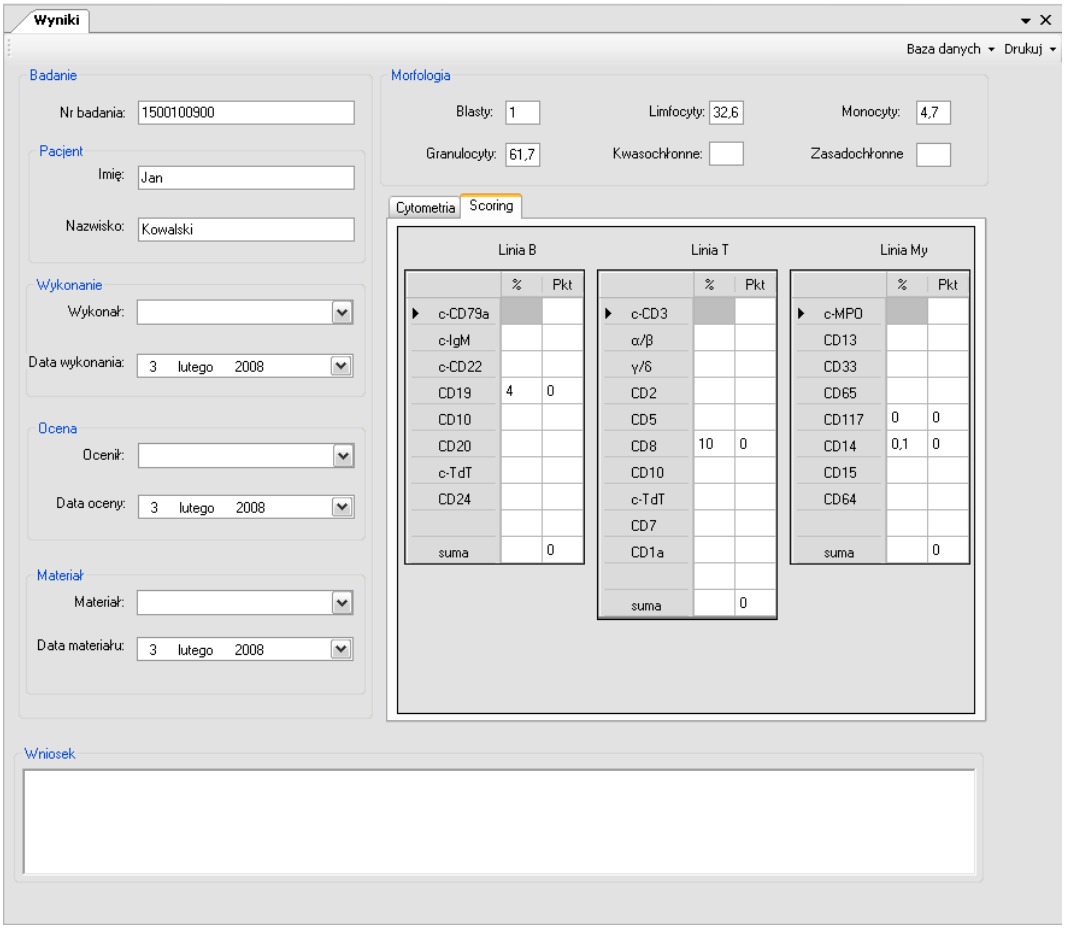
Okno `Wyniki` umożliwia również generowanie dokumentu *RTF* (ang. *Rich Text Format*) na podstawie aktualnie wyświetlanych danych. Korzystając z menu `Drukuj` można wygenerować następujące dokumenty:

- `Cytometria` - wydruk odpowiadający widokowi `cytometria`,
- `Scoring` - wydruk odpowiadający widokowi `scoring`,
- `Wnioski` - wydruk w formie przypominającej obecnie generowane w laboratorium wyniki badania przeznaczone dla lekarza kierującego na badanie

Szablony generowanych dokumentów mogą być niemalże dowolnie modyfikowane przez użytkownika. Użytkownik może zmieniać układ elementów na stronie, a także dodawać nowe, stałe dla każdego szablonu elementy, edytując je w dowolnym edytorze obsługującym pliki *RTF*. Dane związane z konkretnym badaniem są w tym szablonie wpisywane za pomocą parametrów, których nazwy rozpoczynają się i kończą symbolem $. Moduł do generowania dokumentów *RTF* odczytuje szablony, wstawia konkretne dane w miejsce parametrów i zapisuje wynikowy plik na dysku. Rozwiązanie takie umożliwia także łatwe dodawanie nowych szablonów wykorzystujących istniejące parametry.



Rysunek 6.5. Wyniki - widok `cytometria`.



Rysunek 6.6. Wyniki - widok `scoring`.

Rozdział 7

Zakończenie

7.1 Efekty wykonanej pracy

W ramach pracy stworzono system do wspomagania decyzji diagnostycznych. Sposób implementacji umożliwia łatwe i szybkie wdrożenie aplikacji i dostosowanie jej do specyficznych warunków (np. innych systemów wykorzystywanych w laboratorium).

System wdrożono w Katedrze i Zakładzie Immunologii Klinicznej Uniwersytetu Medycznego w Poznaniu.

Zrealizowano ustaloną wcześniej specyfikację wymagań i zaimplementowano następującą funkcjonalność:

- odczyt danych z plików *FCS*,
- usuwanie danych osobowych z plików *FCS*,
- rozpoznawanie subpopulacji komórkowych przy wykorzystaniu analizy skupień,
- różnicowanie zbioru zdarzeń na komórki i artefakty,
- wizualizacja danych,
- generowanie dokumentów *RTF*,
- komunikacja z bazą danych.

Aplikacja umożliwia odczyt plików *FCS* oraz ich eksport do formatu *CSV*. Stworzono dwie wersje modułu do odczytu danych. Niezależny program `FCSReader` do odczytu pojedynczych plików *FCS*, pochodzących z różnego rodzaju badań, umożliwia dostęp do wszystkich informacji zawartych w pliku. Druga, zintegrowana z całym systemem forma pozwala na przegląd katalogów w poszukiwaniu plików *FCS* oraz ich grupowanie w struktury hierarchiczne, odpowiadające poszczególnym badaniom (przeprowadzanym z wykorzystaniem większej ilości próbek i rezultatami zapisanymi w wielu plikach). Dodatkowa, pomocnicza aplikacja `FCSCleaner`, umożliwia usuwanie danych personalnych z pliku *FCS* bez naruszania jego struktury.

Różnicowanie rejestrowanych przez detektory cytometru zdarzeń na komórki oraz artefakty pozwala na normalną ocenę oczyszczonych w ten sposób danych. Taka funkcjonalność może okazać się szczególnie użyteczna przy ocenie badań z dużą ilością komórek uszkodzonych.

Funkcje analizy danych pozwalają wydzielić subpopulacje komórkowe w każdej z badanych próbek i określić ich immunofenotyp (w zakresie ekspresji badanych antygenów). Sposób implementacji algorytmów użytych w programie umożliwia zarówno łatwą zmianę zadanej ilości wyznaczanych skupień jak i sposobu kojarzenia ich z rzeczywistymi subpopulacjami komórkowymi.

Aplikacja pozwala na tworzenie własnych szablonów dla raportów z wynikami badań. Możliwość załączenia rysunków czy dodatkowych danych (np. norm) pozwala na lepsze przedstawienie wyników badania i usprawnienie komunikacji między lekarzem klinicystą a oceniającym badanie specjalistą.

Funkcje realizujące komunikację z bazą danych zapewniają swobodny dostęp do wyników poprzednich badań. Pozwala to na szybszą pracę przy ocenie postępu choroby czy wyników leczenia. Łatwość dokonywania porównań między aktualnym badaniem a poprzednimi może zachęcić lekarzy do częstszego uwzględniania zebranych już danych, co może poprawić efektywność stawianych diagnoz.

7.2 Przebieg realizacji

Istotną częścią projektu okazało się pogłębienie lub zdobycie wiedzy dotyczącej oceny rozrostów nowotworowych przy użyciu cytometrii przepływowej. Praca wymagała zrozumienia nie tylko aspektów medycznych procesu diagnostyki, ale również rozwiązań technicznych (konieczność kompensacji danych pochodzących z badania) i formalnych (sposób przyjmowania zleceń, zasady tworzenia formularzy z wynikami dla lekarza kierującego na badanie).

Zrozumienie i wielokrotna analiza sposobu podejmowania decyzji diagnostycznych przez lekarzy pozwoliła na sformalizowanie tego procesu i umożliwiła opracowanie oraz implementację odpowiednich algorytmów analizy danych. Konsultacje na każdym etapie pracy, zarówno wewnątrz zespołu jak i z lekarzami, pozwoliły na bieżące korekty w specyfikacji wymagań oraz programie.

Jednym z problemów na jaki natrafiono, było przeprowadzenie wiarygodnych testów części odpowiadającej za analizę danych. Pierwsze testy porównawcze (porównanie wyników działania systemu z oceną dokonaną przez lekarza) dały dobre rezultaty. Niestety, były one bardzo ograniczone ze względu na czas trwania projektu i dużą czasochłonność pojedynczego testu.

Dodatkowym utrudnieniem jest brak możliwości przeprowadzenia obiektywnych testów, gdyż jedyną kontrolą poprawności działania całego algorytmu jest ocena lekarza. Wyniki analizy tej samej próbki dokonanej przez różnych lekarzy czy w różnych laboratoriach mogą być odmienne.

7.3 Perspektywy dalszego rozwoju

W trakcie realizacji projektu wraz z coraz bardziej szczegółowym poznawaniem zasad diagnostyki pojawiały się kolejne pomysły, jednak ze względu na ograniczenia czasowe nie udało się ich jak dotąd zrealizować. Zachowanie otwartej struktury pozwala na rozbudowę całego systemu i łatwe przystosowanie go do nowych zadań w przyszłości.

Dotychczasowy sposób zapisywania wyników przez lekarzy analizujących badania uwzględnia jedynie binarną klasyfikację komórek pod względem obecności danego antygenu (antygen występuje lub nie), co nie w pełni odpowiada rzeczywistemu procesowi osłabiania lub wzmacniania jego ekspresji. Być może informacje dotyczące np. średniej wartości ekspresji danego antygenu (mierzonej na podstawie fluorescencji znakowanych przeciwciał) okażą się istotne klinicznie.

Dzięki uzyskanemu dostępowi do bazy danych z wynikami z przeprowadzonych już badań istnieje możliwość porównywania wyników generowanych przez algorytm z wyznaczonymi przez lekarzy. Po dopisaniu dodatkowego modułu testowego i zgromadzeniu odpowiednich danych źródłowych w postaci plików *FCS* możliwe stanie się sprawne prowadzenie testów dla wielu badań i zbieranie informacji, na ile i w jakich przypadkach zaimplementowany algorytm daje wyniki odmienne od oceny lekarza. Łatwość zmian w module analizy pozwoli w przyszłość przeprowa-

dzić testy dla różnych rozwiązań algorytmicznych i wybrać optymalne lub kilka optymalnych w zależności od kierunku diagnostyki.

Dodatek A

Słownik używanych w pracy pojęć i skrótów

antygen substancja wywołująca swoista odpowiedz układu odpornościowego (m.in. zdolna do reagowania z przeciwciałami)

blasty młode, niskozróznicowane komórki układu krwiotwórczego; ich obecność w krwi obwodowej lub zwiększona liczba w szpiku kostnym (powyżej 5% wszystkich leukocytów) może świadczyć o rozroście nowotworowym

CD antygen różnicowania (ang. *Cluster Determinants*), antygen wykrywany w sposób swoisty przy użyciu odpowiednich przeciwciał monoklonalnych

CD45 *LCA* (ang. *Leukocyte Common Antigen*), antygen różnicowania obecny na powierzchni wszystkich leukocytów z wyjątkiem plazmocytów

CD14 antygen różnicowania obecny na powierzchni monocytów

ekspresja antygenu obecność białka w komórce jako efekt aktywacji danego genu

fluorescencja zjawisko emisji światła zachodzące podczas przejścia wzbudzonej promieniowaniem świetlnym cząsteczki do stanu podstawowego polegające na emisji fali świetlnej o większej długości od wcześniej zaabsorbowanej

FSC (ang. *forward scatter*) tzw. przedni detektor światła rozproszonego w cytometrze przepływowym służący do pomiaru wielkości komórki

immunofenotyp charakterystyczny dla danej komórki (grupy komórek) układ antygenów związany z jej różnicowaniem, dojrzewaniem i funkcją

leukocyty krwinki białe, komórki krwi, których zadaniem jest ochrona organizmu przed obcymi, potencjalnie niebezpiecznymi czynnikami

limfocyty leukocyty uczestniczące w swoistej odpowiedzi immunologicznej

monocyty leukocyty wywodzące się z linii monocytoidalnej, biorące udzial w nieswoistej odpowiedzi odpornościowej, mogące ulec przekształceniu w komórki prezentujące antygen

neutrofile granulocyty obojetnochłonne, rodzaj leukocytów wywodzących się z mieloidalnej linii różnicowania i stanowiący najliczniejsza subpopulacje wśród wszystkich białych krwinek u zdrowego człowieka, biorące udział w nieswoistej odpowiedzi odpornościowej

przeciwciało immunoglobulina, wydzielane w przebiegu odpowiedzi immunologicznej przez limfocyty B białko, które ma zdolność do swoistego rozpoznawania antygenów

przeciwciała monoklonalne przeciwciała wytwarzane przez ten sam klon limfocytów B; wykazują one taką samą swoistość względem danego antygenu

SSC (ang. *side scatter*) tzw. boczny detektor światła rozproszonego w cytometrze przepływowym służący do pomiaru wielkości komórki

Literatura

[dhTP] Prof. dr hab. Tomasz Panek. Założenia konstrukcji metod taksonomicznych i metod analizy czynnikowej. [on-line] `http://www.sgh.waw.pl/instytuty/isd/publikacje/`.

[HMS01] J. Hand, H. Mannila, P. Smyth. *Pricinciples of Data Mining*. MIT Press, 2001.

[HWG06] Anders Hejlsberg, Scott Wiltamuth, Peter Golde. C# language specification. [on-line] `http://download.microsoft.com/download/3/8/8/388e7205-bc10-4226-b2a8-75351c669b09/CSharp%20Language%20Specification.doc`, 2006.

[KMLO02] Aldona Kaczmarek, Andrzej Mackiewicz, Ewa Leporowska, Tatsuya Osawa. Rola i miejsce cytometrii przepływowej w diagnostyce klinicznej. *Współczesna onkologia*, 6, 6 2002.

[pdhJe97] prof. dr. hab. Jan Żeromski, redaktor. *Metody immunologiczne: Przewodnik do ćwiczeń z immunologii dla studentów wydziału lekarskiego*. Wydawnictwo Uczelniane Uniwersytetu Medycznego w Poznaniu, 1997.

[Per06] Stephen C. Perry. *Core C# i .NET*. Wydawnictwo Helion, 2006.

[SBB+96] Larry Seamer, Bruce Bagwell, Luther Barden, Marc Christofferson, Marc Christofferson, Marc Christofferson, Doug Redelman Gary Salzman, James Wood, Robert Murphy. Data file standard for flow cytometry, version fcs3.0. [on-line] `http://www.isac-net.org/index.php?option=com_content&task=view&id=101&Itemid=46#4`, 1996.

[Sta06] StatSoft. Elektroniczny podręcznik statystyki pl. [on-line] `http://www.statsoft.pl/textbook/stathome.html`, 2006.

[vLvdVW+04] E.G. van Lochem, V.H.J. van der Velden, H.K. Wind, N.A.C. Westerdaal J.G. te Marvelde, J.J.M. van Dongen. Improving lab practice: Immunophenotypic differentiation patterns of normal hematopoiesis in human bone marrow: Reference patterns for age-related changes and disease-induced shifts. *Cytometry Part B: Clinical Cytometry*, 60B, 2004.

Instytut Informatyki, Wydział Informatyki i Zarządzania
Politechnika Poznańska

Skład przy użyciu systemu LaTeX.

BibTeX:

```
@mastersthesis{ key,
    author = "Maria Marchwicka \and Marek Lankauf \and Michał Nowak ",
    title = "{System wspomagania decyzji diagnostycznych w ocenie badania immunofenotypowego
rozrostów nowotworowych wywodzących się z komórek układu krwiotwórczego}",
    school = "Poznan University of Technology",
    address = "Pozna{\'n}, Poland",
    year = "2008",
}
```