Содержание

[1 Статистические методы алгоритмизации процессов селекции и распознавания](#_Toc420579674)

[*1.1 Основные положения*](#_Toc420579675)

[*1.2 Общий вид критериев согласия*](#_Toc420579676)

[*1.3 Стратегия Байеса*](#_Toc420579677)

[*1.4 Гарантирующая (минимаксная) стратегия*](#_Toc420579678)

[*1.5 Стратегия Неймана-Пирсона*](#_Toc420579679)

[2 Синтез системы классификации радиолокационных целей.](#_Toc420579680)

[*2.1 Смеси многомерных нормальных распределений*](#_Toc420579681)

[*2.2 Весовые коэффициенты*](#_Toc420579682)

[*2.2.1 Метод ранжирования*](#_Toc420579683)

[*2.2.2 Метод непосредственной оценки*](#_Toc420579684)

[*2.2.3 Анализ исследуемых методов*](#_Toc420579685)

[*2.3 Байесовская оценка весовых коэффициентов*](#_Toc420579686)

[*2.4 Алгоритм обучения*](#_Toc420579687)

[*2.5 Алгоритм классификации радиолокационных целей.*](#_Toc420579688)

[*2.6 Расчет коэффициентов обратным методом*](#_Toc420579689)

[*2.7 Задачи селекции и распознавания*](#_Toc420579689)

[3 Разработка компьютерной программы.](#_Toc420579691)

[*3.1 Общие требования и характеристики системы*](#_Toc420579692)

[*3.2 База данных*](#_Toc420579693)

[*3.3 Структура интерфейса пользователя*](#_Toc420579694)

**1 Статистические методы алгоритмизации процессов селекции и распознавания**

Говоря о статистических методах распознавания, мы предполагаем установление связи между отнесением объекта к тому или иному классу (образу) и вероятностью ошибки при решении этой задачи. В ряде случаев это сводится к определению апостериорной вероятности принадлежности объекта к определенному классу  при условии, что признаки этого объекта приняли значения .

Основное преимущество статистических методов распознавания состоит в возможности одновременного учета признаков различной физической природы, так как они характеризуются безразмерными величинами – вероятностью их появления.

***1.1 Основные положения***

Каждый объект обладает присущими ему признаками, по которым можно его идентифицировать[2]. Предположим, есть два класса объектов: А1 и А2, которые описываются условными вероятностями f1(x) и f2(x) значения признака x, а также некоторой априорной вероятностью Р(А1) и Р(А2) появления этого класса на входе системы. Пусть также известна матрица потерь , её также называют матрицей штрафов или платёжной матрицей. Её элементы описывают риски правильных или ошибочных решений[2]. Она имеет вид:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Для двух классов её можно представить в виде:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Система может выбирать правильные или неправильные решения, поэтому необходимо снижать совокупность всего ущерба связанного с ошибочным выбором решений. Пусть Rср – минимальный средний риск, тогда х0 – точка на оси Х (рисунок 1.1) соответствующая данному выбору риска, делящая пространство на два полупространства. Обозначим их Х1 и Х2. Область пересечения вероятностей f1(x) и f2(x) не позволяет точно определить принадлежность цели к определенному классу. Допустим, при x< х0 система определила принадлежность объекта к классу А1, в то время как он относиться к классу А2. Такая ошибка называется ошибкой второго рода. Обозначим ее как Q2. Противоположная ей ошибка первого рода Q1 происходит тогда, когда система соотнесла объект с классом А2, в то время как он относиться к классу А1[3]. С точки зрения селекции, если рассматривать класс А1 как истинный, а класс А2 как ложный, то ошибка первого рода – это ложная тревога, а ошибка второго рода – пропуск[4]. Это приведено в таблице 1.1.

*Таблица 1.1 — Результат выбранного решения при данной ситуации*

|  |  |  |
| --- | --- | --- |
| Решение | Истинная ситуация | |
| Относится к классу А1 | Относиться к классу А2 |
| Принят А1 | Правильно | Ошибка II рода (пропуск) |
| Принят А2 | Ошибка I рода (ложная тревога) | Правильно |

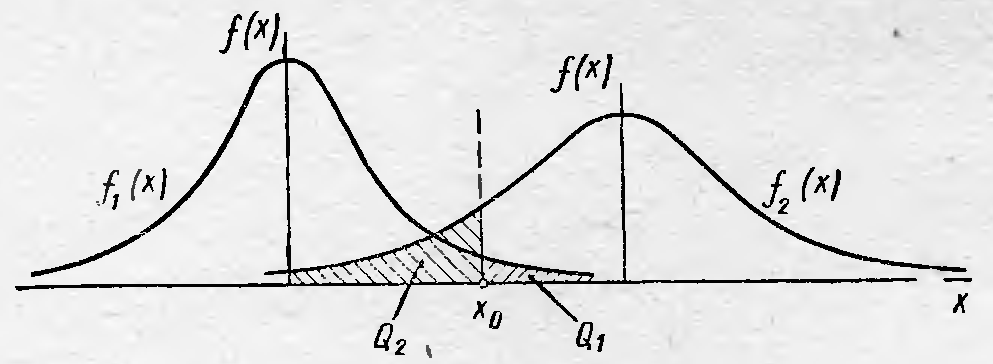
Пусть значение признака x подчинено нормальному закону распределения с математическими ожиданиями μ1 и μ2 и среднеквадратичными отклонениями σ1 и σ2 соответственно:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Как видно из рисунка 1 Q1 и Q2 находятся как:

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Исходя из формулы полной вероятности, определим условные вероятности правильных решений:



*Рисунок 1.1 — График условной плотности распределения вероятностей функций f1(x) и f2(x)*

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

Найдем значение среднего риска:

|  |  |  |
| --- | --- | --- |
|  |  | (1.1) |

Общее значение среднего риска для M классов определяется формулой:

|  |  |  |
| --- | --- | --- |
|  |  |  |

где Gl – это область признака для класса l.

Продифференцируем данное значение по x и приравняем производную к нулю, положив x равным х0:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Откуда:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Величину λ(x)= f2(x)/f1(x) называют коэффициентом правдоподобия. Подставим вместо обозначений f2(x) и f1(x) функции, подчинённые соответствующим законам нормального распределения N2(x2,μ2,σ2) и N1(x1,μ1,σ1):

|  |  |  |
| --- | --- | --- |
|  |  | (1.2) |

И выразим из выражения (1.2) х0:

|  |  |  |
| --- | --- | --- |
|  |  |  |

При σ2= σ1= σ:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Если с11=с22=0, с21=с12, Р(А1)=Р(А2), то х0=(μ2+μ1)/2.

Для всех трех случаев значение х0 определяет оптимальную границу разделения признакового пространства[4].

Из условия минимума среднего риска уравнение границы в многомерном признаковом пространстве между областями Gk и Gl, соответствующими классами Ak и Al,

|  |  |  |
| --- | --- | --- |
|  |  |  |

Если положить сkk=сll=0, сkl=сlk=1, то:

|  |  |  |
| --- | --- | --- |
|  |  |  |

***1.2 Общий вид критериев согласия***

Критериями согласия обычно называют критерии, предназначенные для проверки простой гипотезы H1={A=A1} при сложной альтернативе H2={A1 неверна}. В нашем случае H2={A=A2}. Критерии согласия принимают решения в пользу той или иной гипотезы, исходя из величины функции отклонения.

Опишем конструкцию критерия для случая двух классов. Пусть существует некая непрерывная функция λ(x), которая обладает следующим свойством:

* если верна гипотеза H1, т.е. объект относиться к классу A1, то λ(x)<C;
* если верна гипотеза H2, т.е. объект относиться к классу A2, то λ(x)≥C.

Величину С называют пороговым значениям отношения правдоподобия[5]. Пусть это будет значение функции λ(x) в некоторой точке x0. В виде формулы данный критерий можно представить так:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Критерий согласия «работает» по принципу: если при каком-то значении x функция отклонения превышает абсолютное значение, то это свидетельствует в пользу альтернативы, и наоборот[6]. В дальнейшем рассмотрим три критерия соответствующие трем разным стратегиям.

***1.3 Стратегия Байеса***

Данная стратегия выбора решений направлена на снижение значения средних потерь, т. е. поиск такого решения, чтобы средний риск достигал своего минимума. Такую стратегию называют стратегией Байеса, а минимальный средний риск – байесовским. Она является оптимальной в том случае, если для всех классов объектов, распознаваемых системой, известны функции плотности fi(х1, . . . ,хn), априорные вероятности Р(Аi) и матрица потерь С. Выбор любой другой стратегии приведет к увеличению значения среднего риска, вследствие увеличения числа ошибочных решений. Докажем данное утверждение.

Пусть даны два класса объектов А1 и А2, которые описываются условными вероятностями f1(x) и f2(x) значения признака x. Точка х0 определяет оптимальную решающую границу между классами, разделяя пространство Х на два полупространства Х1 и Х2 (рисунок 1.2). В соответствие со стратегией Байеса, если измеренное значение находиться в полупространстве Х1, то есть х<x0, объект относиться к классу A1, в обратном случае, если х>x0, объекту сопоставляется класс A2.



*Рисунок 1.2 — График условной плотности распределения вероятностей функций f1(x) и f2(x)*

Допустим, выбрана еще какая-то стратегия, которая разделяет пространство Х на два полупространства Х`1 и Х`2. Тогда решения выбираются следующим образом: если х0<x`0, объект относиться к классу A1, если х0≥x`0, объект относиться к классу A2.

Найдём разность средних рисков Rср`-Rср min . Использовав формулу 1.1 и приняв с11=с22=0 и с12=с1, с21=с2, Р(А1)c1= Р(А2)c2λ0, после некоторых преобразований получим:

|  |  |  |
| --- | --- | --- |
|  |  |  |

В области r2=(x0;x`0]X2, f2(x) > λ0f1(x), следовательно, Rср`-Rср min > 0, а значит Rср`>Rср min.

Допустим теперь, что выбрана стратегия, которая разделяет пространство Х на два полупространства Х``1 и Х``2. Тогда решения выбираются следующим образом: если х0<x``0, объект относиться к классу A1, если х0≥x``0, объект относиться к классу A2.

Найдём разность средних рисков Rср``-Rср min, используя те же обозначения и равенства, что и в предыдущем примере.

|  |  |  |
| --- | --- | --- |
|  |  |  |

В области r1=[x``0;x0)X1, f2(x) < λ0f1(x), следовательно, Rср``-Rср min > 0, а значит Rср``>Rср min.

Из доказательства следует, что любая из выбранных стратегий не достигает минимального значения среднего риска, кроме стратегии Байеса. Эту стратегию можно описать с помощью теоремы гипотез или формулы Байеса[2].

|  |  |  |
| --- | --- | --- |
|  |  | (1.3) |

где Р(Аi|an) – апостериорная вероятность принадлежности объекта к классу Аi при ситуации an, где х=x0.

Пусть M=2, тогда по формуле Байеса для двух классов находятся апостериорные вероятности Р(А1| x0) и Р(А2| x0). Условный риск, связанный с решением отнесения объекта к классу А1 равен:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Условный риск, связанный с решением отнесения объекта к классу А2 равен:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Для байесовской стратегии должна решаться задача с условным минимальным риском, поэтому, например, при принадлежности объекта к классу А2 должно выполняться условие

|  |  |  |
| --- | --- | --- |
|  |  |  |

Для апостериорных вероятностей формула соответственно будет выглядеть так[4]:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Когда объект характеризуется n признаками формула Байеса принимает вид:

|  |  |  |
| --- | --- | --- |
|  |  | (1.4) |

Таким образом, байесовское решение в разбираемых задачах описывается линейными неравенствами в апостериорных вероятностях P(Ai/an)[7].

В том случае, если признаки являются независимыми между собой[8], то:

|  |  |  |
| --- | --- | --- |
|  |  | (1.5) |

Например, при двух независимых признаках распределенных по нормальному закону функция будет иметь вид[9]:

|  |  |  |
| --- | --- | --- |
|  |  |  |

При распознавании каждого объекта, данные которых были переданы радиолокационными системами, классификатор может отнести его к одному из M возможных типов. Приведем формулу расчета условных средних потерь для каждого объекта[10]:

|  |  |  |
| --- | --- | --- |
|  |  |  |

С этими потерями классификатор причисляет объект к классу, которому соответствует наименьшие условные потери. Очевидно, что и математическое ожидание полных потерь на множестве всех решений также будет минимизировано. Классификатор, минимизирующий математическое ожидание общих потерь, называется байесовским[11].

***1.4 Гарантирующая (минимаксная) стратегия***

Стратегию Байеса можно применить только в том случае, если известна матрица потерь С, априорная вероятность P(Ai) и функции fi(x1,…,xn). Если же априорная вероятность неизвестна (на практике может быть известна только матрица потерь и функции плотности), в качестве действующей стратегии выбирают минимаксную.

Её принцип заключается в том, что выбирается такая априорная вероятность, при которой значение байесовского риска будет максимальным. Эта стратегия относиться к классу байесовских.

Рассмотрим два класса А1 и А2. Также примем во внимание, что Р(А2)=1-Р(А1), с11=с22=0 и с12=с1, с21=с2.

|  |  |  |
| --- | --- | --- |
|  |  | (1.6) |

График функции Rср min=f(Р(А1)) представлен на рисунке 1.3. В крайних точках графика Rср min=0. Обозначим точку, в которой график достигает своего максимального значения, как P(A1)=P`(A1). Причём средние потери будут определяться положением касательной в этой точке к кривой Rср min=f(Р(А1)). Выпишем уравнение этой касательной:

|  |  |  |
| --- | --- | --- |
|  |  |  |

где Q`1=Q1(P`(A1)) и Q`2=Q2(P`(A1)) – ошибки первого и второго рода соответственно при априорной вероятности P(A1)=P`(A1).

Так как в точке P(A1)=P`(A1) функция достигает максимума, касательная в этой точке параллельна, а значит, средние потери при данном состоянии остаются неизменны, даже если меняется значение априорной вероятности.

Таким образом, применение минимаксной теории оставляет постоянными при P(A1)>P`(A1) и P(A1)<P`(A1) средние потери, которые не будут превышать максимальные значения минимальных байесовских потерь.

Рассмотрим, что будет в том случае, если мы выберем значение P(A1)=P``(A1). Уравнение касательной к кривой будет иметь вид:

|  |  |  |
| --- | --- | --- |
|  |  |  |



*Рисунок 1.3 — График зависимости среднего риска Rср от априорной вероятности Р(Аi)*

где Q``1=Q1(P``(A1)) и Q``2=Q2(P``(A1)) – ошибки первого и второго рода соответственно при априорной вероятности P(A1)=P``(A1).

Проанализируем эту касательную. При P(A1)≤P```(A1) касательная не будет превышать значения средних потерь достигнутых при минимаксной теории, но если P(A1)>P```(A1), потери будут возрастать до чрезмерных значений. Выбор же минимаксной стратегии предохраняет от подобных потерь.

Перейдем к алгоритму принятия решений. Для этого продифференцируем выражение 1.6 по P(A1), приравняв производную нулю и исходя из определений ошибок первого и второго рода, получим:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Данное условие равенства средних рисков при ошибках первого и второго рода позволяет определить оптимальное значение х0 и построить следующий алгоритм классификации: если выбираемое значение признака х<x0, то объект принадлежит к классу A1, если же х≥x0, то к классу A2.

Если же мы выбираем значение априорной вероятности равной P(A1)=P`(A1), то, определив пороговое значение коэффициента правдоподобия по формуле (1.4), можем заключить, что при λ(x)< λ`0 объект принадлежит к классу A1, а при λ(x)≥ λ`0 - к классу A2.

|  |  |  |
| --- | --- | --- |
|  |  | (1.7) |

В общем случае можно отметить, что минимаксная стратегия - это стратегия Байеса при наихудших значениях априорных вероятностей, однако, имеющая при этом гарантированное значение среднего риска.

***1.5 Стратегия Неймана-Пирсона***

В тех случаях, когда нам известна лишь функции плотности fi(х1, . . . ,хn), байесовские стратегии рассматривать невозможно. В таких случаях рассматривают стратегию Неймана-Пирсона.

Эту стратегию к принятию решений основывается на геометрической модели областей классов в пространстве событий an с вероятностной мерой. Однако при этом некоторые значения ошибок жестко задаются наблюдателем, и строится наилучшее при таких условиях разбиение, оптимизирующее совокупность оставшихся свободными элементов. Наиболее распространен и очевиден этот подход в двухальтернативном случае. При этом закрепляется одна из двух ошибок на уровне h и строится разбиение, минимизирующее первую ошибку[12].

В зависимости от принимаемого решения при распознавании цели, находиться допустимое значение ошибки первого рода, которое не должно превышать какого-то значения h. При данном ограничении Q1=1(x)dx≤h, требуется найти решение х0 задачи = 2(x)dx. Этим условиям удовлетворяет уравнение 1(x)dx=h, из которого находиться х0. Если выбрать х`0>х0, Q2 будет возрастать. При х`0<х0 возникает противоречие условиям задачи.

Рассмотрим геометрическую интерпретацию пройденных критериев. Для этого построим рабочую характеристику (изображена пунктирной кривой на рисунке 1.4) – график зависимости максимальной мощности D2 = 1 - Q2 от Q1[8]. При этом отметим, что если Q1=0, Q2=1, D2=0, а при Q1=1,Q2=0, D2=1. По определению:

|  |  |  |
| --- | --- | --- |
|  |  |  |

Продифференцируем D2 по Q1:

|  |  |  |
| --- | --- | --- |
|  |  |  |



*Рисунок 1.4 — Рабочая характеристика приёмника*

∂D2/∂Q1 есть тангенс угла наклона α касательной к рабочей характеристике. Для критерия Байеса касательная при λ=λ0 изображена на рисунке. Абсцисса этой точки определяет условную вероятность ошибки первого рода.

Поиск D2 и Q1 для минимаксного критерия сводится к нахождению уравнения прямой путём дифференцирования по P(A1) выражения для Rср (формула 1.1).

|  |  |  |
| --- | --- | --- |
|  |  |  |

Выполнив несложные преобразование получим исходно уравнение прямой:

|  |  |  |
| --- | --- | --- |
|  |  |  |

На рисунке 1.4 это прямая AB. Коэффициент наклона прямой равен (с11-с12)/(с21-с22)=tgβ. Точка пересечения прямой AB с рабочей характеристикой определяет значения D2 и Q1 для минимаксного критерия.

Рассмотрев три стратегии можно выбрать одну основную, на которую будет опираться алгоритм распознавания. В таблице 1.2 приведена их сравнительная характеристика.

*Таблица 1.2 — Сравнение статистических методов селекции и распознавания*

|  |  |  |
| --- | --- | --- |
|  | Минимальный риск R | Объем априорных данных |
| Стратегия Байеса | всегда минимальный | максимальный |
| Гарантирующая стратегия | Всегда максимальный | Отсутствует априорная вероятность |
| Стратегия Неймана-Пирсона | Регулируется в зависимости от условий | Отсутствует априорная вероятность и матрица штрафов |

Для данной работы выберем стратегию Байеса, опирающуюся на формулу (1.4), поскольку она предоставляет нам максимально правильную классификацию объектов. При этом естественно предполагается, что у нас будет присутствовать полный набор априорных данных, необходимых для реализации алгоритма по этому критерию. Сложность у всех стратегий примерно одинакова и зависит от количества рассматриваемых классов и признаков объекта. С их возрастанием она будет увеличиваться экспоненциально, поэтому требуется модификация используемых формул из теории.

**2 Синтез системы классификации радиолокационных целей.**

В предыдущей главе было упомянуто о том, что при большом увеличении классов целей сложность алгоритма будет сильно возрастать, поэтому возникает необходимость в поиске оптимального решения задачи. Таким решением может служить смеси многомерных нормальных распределений.

***2.1 Смеси многомерных нормальных распределений***

Важнейшей задачей при разработке радиолокационных станций является определение способа «наилучшей» обработки данных, полученных в результате наблюдений. Наиболее часто при обработке радиолокационных сигналов применяется нормальный  (Гауссов) закон распределения случайной величины.

Так, например, смеси многомерных нормальных распределений позволяют приближать любые непрерывные плотности вероятности. Они являются универсальными аппроксиматорами плотностей. В задачах классификации это позволяет восстанавливать функции правдоподобия классов.

В общем случае функция правдоподобия представляет собой функцию, которая зависит от определенного параметра (признака) при фиксированном событии. Таким образом, она показывает, насколько правдоподобен выбранный параметр при заданном событии. Для определенного класса распознаваемого объекта с рассматриваемым признаком можно определить функцию правдоподобия, представляющую из себя смешанную модель и состоящую из конечного числа нормальных распределений, описываемую формулой:

|  |  |  |
| --- | --- | --- |
|  |  | (2.1) |

где – пропорции смешивания () и – условная плотность, распределенная по нормальному закону, которая зависит от вектора , – количество компонентов смеси [17].

***2.2 Весовые коэффициенты***

также можно назвать коэффициентом важности или весовым коэффициентом, поскольку он определяет важность (вес) компоненты, входящей в рассматриваемую смешанную модель.

Теперь остановимся подробнее на описании методов расчета самих весовых коэффициентов, но для начала опишем некоторые задачи, при которых используются веса :

1. нормировка признаков;
2. степень важности признаков, т.е. наиболее важному признаку или классу приписывается большее значение коэффициента;
3. отбор признаков (). Позволяет не рассматривать ненужные классы или признаки. Это упрощает задачу классификации, сокращая время работы некоторых алгоритмов [15].

Рассмотрим некоторые методы расчета весовых коэффициентов.

***2.2.1 Метод ранжирования***

Данный метод позволяет упорядочить компоненты по степени возрастания или убывания их влияния в зависимости от особенностей рассматриваемого события. Результаты ранжирования *n* компонент *m* экспертами можно представить в виде таблицы.

*Таблица 2.1 ― Результаты опроса экспертов по рассматриваемым компонентам.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Номер эксперта | Рассматриваемые компоненты | | | |
| *X*1 | *X*2 | *…* | *Xn* |
| 1 | *r*11 | *r*12 | *…* | *r*1*n* |
| 2 | *r*21 | *r*22 | *…* | *r*2*n* |
| … | *…* | *…* | *…* | *…* |
| *M* | *rm*1 | *rm*2 | *…* | *rmn* |

Оценку важности той или иной компоненты проводит группа специализированных экспертов, и каждый из них представляет свой вектор оценок по данной группе компонент, основываясь на знаниях в области слабо формализованных задач. Компоненты расставляются в порядке их важности по следующему правилу:

1. Эксперт располагает компоненты по убыванию их важности слева направо.
2. Каждой компоненте присваивается оценка от *n* до 1 (самой важной – *n* и далее по убыванию до 1).
3. Для каждой компоненты высчитывается сумма оценок, и далее высчитывается доля от всех полученных сумм. В виде формулы это можно представить так:

|  |  |  |
| --- | --- | --- |
|  |  |  |

где – весовой коэффициент *j*-й компоненты; – оценка, поставленная *j*-й компоненте *i*-м экспертом [16].

***2.2.2 Метод непосредственной оценки***

Часто бывает желательным определить, насколько одна компонента более значима, чем другие. Поэтому этот метод основан на том, что эксперты оценивают важность частной компоненты по определённой шкале, например, от 0 до 10. Именно поэтому метод непосредственной оценки иногда именуют также балльным методом или методом прямой расстановки. При этом разрешается оценивать важность дробными величинами или приписывать одну и ту же величину из выбранной шкалы нескольким компонентам. Таблица оценок представлена так же, как и в предыдущем методе [14].

Алгоритм расчета весовых коэффициентов следующий:

1. Каждый эксперт проставляет оценки по всем компонентам в рамках заданной шкалы.
2. Происходит пересчет оценок по формуле:

|  |  |  |
| --- | --- | --- |
|  |  |  |

1. Далее полученные оценки для каждой компоненты суммируются и нормируются, также как и в предыдущем методе.

|  |  |  |
| --- | --- | --- |
|  |  |  |

***2.2.3 Анализ исследуемых методов***

Итак, были рассмотрены два способа расчета весовых коэффициентов. Анализируя вычислительную сложность обоих методов, можно сделать вывод, что для метода ранжирования требуется меньшее количество элементарных арифметических операций, чем для метода непосредственной оценки.

Показатель эффективности для обоих случаев одинаков, так как зависит от количества экспертов и является функцией от параметра *m.* Следует также заметить, что оба способа по показателю качества дают невысокую точность измерения, поскольку возможна низкая согласованность экспертов.

Метод непосредственной оценки лучше применять в том случае, когда степень важности компонент сильно варьируется (в некоторых случаях компонента может вообще не учитываться). Метод ранжирования предназначен для тех случаев, когда необходима «строгая» расстановка приоритетов компонент, а также, когда нет возможности дать точную оценку важности между рассматриваемыми компонентами.

В некотором роде метод ранжирования можно рассматривать как частный случай метода непосредственной оценки.

Поскольку оба метода основаны на массовую теоретическую оценку, их применение в данной работе будет невозможна. Причиной отказа от этих способов расчета может также послужить и то, что в алгоритмах самообучения системы необходимо использование результатов работы предыдущих тактов системы. Поэтому возникает необходимость в разработке правильного алгоритма перерасчета весовых коэффициентов.

***2.3 Байесовская оценка весовых коэффициентов***

Весовые коэффициенты по своей сути очень схожи с априорной вероятностью, поэтому их начальная инициализация задается как 1/(количество эталонов).

Используя формулу Байеса в результате проведения моделирования системы с эталонной выборкой значений для определенного класса, мы получаем апостериорные вероятности, которые могут использоваться как достоверная информация для перерасчета весов.

Для такого обучения нам необходимо в формуле 1.3 воспринимать как функцию правдоподобия (формула 2.1). Если функцию правдоподобия представить как:

|  |  |  |
| --- | --- | --- |
|  |  | (2.2) |

M – число эталонов;

– функция распределения по Гауссу, в которой:

* – вектор измеренных значений j-го класса;
* - значение i-го эталона;
* – дисперсия эталона j-го класса.

Подставив её в формулу 1.3, получим формулу для расчета апостериорной вероятности по классам.

|  |  |  |
| --- | --- | --- |
|  |  | (2.3) |

– априорная вероятность;

N – количество классов;

Так как нам необходимо знать лишь апостериорную вероятность одной компоненты, то знак суммы в числителе формулы 2.3 убирается. Таким образом, получаем формулу для пересчета весовых коэффициентов.

|  |  |  |
| --- | --- | --- |
|  |  | (2.4) |

где – весовой коэффициент предыдущего такта обучения;

***2.4 Алгоритм обучения***

Представим формулу 2.4 в виде алгоритма. Он приведен на рисунке 2.1. Основной принцип алгоритма следующий. Сначала происходит формирование функции правдоподобия по каждому классу, далее происходит расчет апостериорной вероятности каждой компоненты, который складывается с текущим значением весового коэффициента. Также следует заметить, что после формулы 2.4 значения весовых коэффициентов будут неотнормированными, поэтому, согласно одному из свойств коэффициентов важности, их следует отнормировать. Данный алгоритм учитывает и это.

***2.5 Алгоритм классификации радиолокационных целей.***

Формула 2.3, которую мы получили в предыдущем пункте, рассчитывает апостериорную вероятность только по одному рассматриваемому признаку. Из этого следует, что мы должны расширить возможности данной формулы на произвольное количество признаков классов. Воспользуемся формулой 1.5. Пояснением к ней служит то, что функцию с вектором из k независимых признаков можно разложить на произведение k простых функций одного признака при условии, что эти признаки независимы между собой. Заменив простые функции выражением 2.2, получим следующее:

|  |  |  |
| --- | --- | --- |
|  |  | (2.5) |

Pr – количество признаков;

Теперь заменим в формуле 1.4 выражением из формулы 2.5. Исходя из всех этих выводов, получаем следующую формулу для алгоритма классификации.

|  |  |  |
| --- | --- | --- |
|  |  | (2.6) |

На рисунке 2.2 приведена схема алгоритма, описывающего формулу 2.6.



*Рисунок 2.1 — Алгоритм перерасчета весовых коэффициентов*



*Рисунок 2.2 — Алгоритм классификации радиолокационных целей*

Поясним принцип работы алгоритма. Когда на вход системы поступает вектор значений признаков объекта, система для каждого класса составляет функцию правдоподобия сначала по одному признаку. Далее, пройдя по всем признакам для всех классов, составляются общие функции правдоподобия и по формуле Байеса рассчитываются апостериорные вероятности.

***2.6 Расчет коэффициентов обратным методом***

Если необходимо рассчитать весовые коэффициенты, но при этом отсутствует обучающая выборка, необходимо решать поставленную задачу методом от противного. Если взглянуть на формулу 2.6, то можно увидеть, что если мы знаем значения апостериорной вероятности, то неизвестными в данном уравнении будут являться только весовые коэффициенты . Так нельзя точно рассчитать значения коэффициентов, то решение необходимо итерационно. Для этого воспользуется дельта правилом.

*η –* скорость обучения.

Δ – разность между значением функции на i-ом шаге и выходным значением.

x - величина входа, соответствующая синоптическому весу.

Стартовое значение коэффициента для каждого из классов есть , где N – количество параметров заданного класса.

Установив параметры для расчета – скорость обучения, выходное значение, синоптический вес, а также максимальное количество итераций, можно вычислить значение коэффициентов.

***2.7 Задачи селекции и распознавания***

Распознавание принадлежности радиолокационных целей определенному, из априори известных, классу является сложной проблемой современной радиолокации. Поэтому разработка методов и алгоритмов селекции и радиолокационного распознавания связана с решением ряда специфических задач. К таким задачам относится:

1. Задача изучения и классификации объектов, для распознавания которых предназначается разрабатываемая система. Необходимость объединения распознаваемых объектов в классы обусловлена, главным образом, практической потребностью оперировать более общими понятиями, чем понятие индивидуального объекта (цели).
2. Задача разработки словаря признаков, используемых как для априорного описания классов, так и для апостериорного представления принадлежности каждого распознаваемого объекта к определённому классу.

Помимо априорной вероятности появления какого-то объекта на входе (вероятностный признак), существует также сигнальные признаки (например, ЭПР, архитектура частей объекта) и траекторные признаки (скорость, дальность, высота полёта, направление полёта) – параметры целей, характеризующие её тактико-технические характеристики. Один из основных подходов к выбору признаков распознавания радиолокационных целей позволяет выделить независимые группы сигнальных и траекторных признаков, которые могут использоваться по отдельности или совместно.

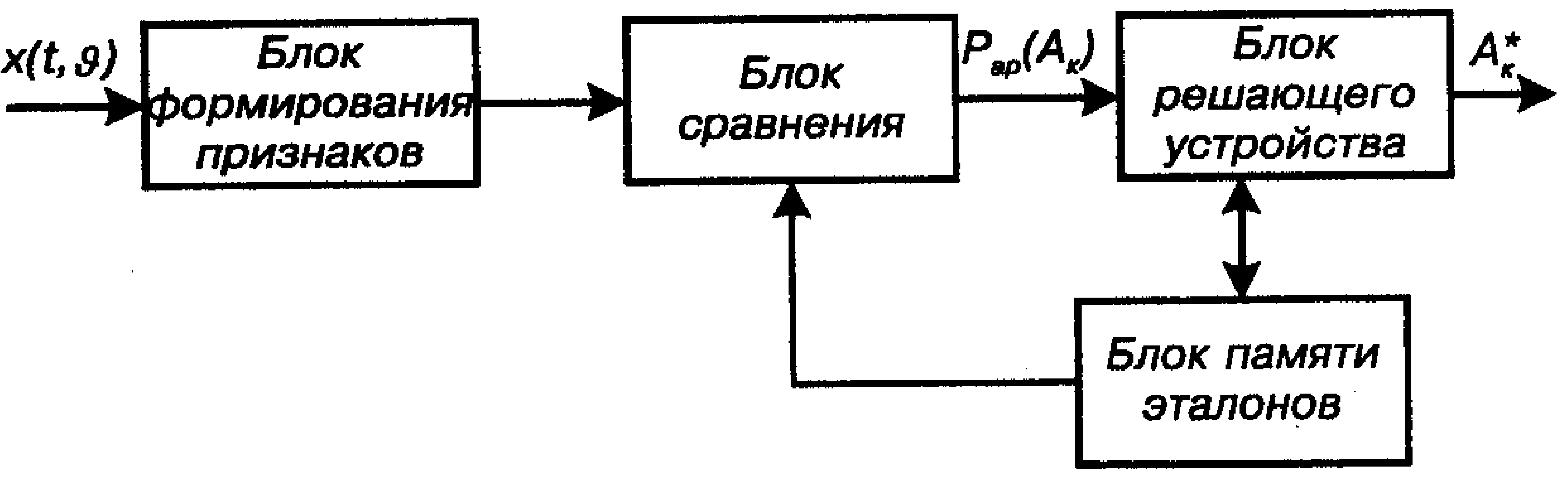
1. Задача разработки алгоритмов распознавания, определяющих процедуру отнесения распознаваемого объекта к тому или иному классу или типу объектов рассматриваемого класса.

Алгоритм распознавания определяет формульно-логическую схему вычисления решающей статистики, характеризующей некоторую меру близости (сходства) распознаваемого объекта и эталона. Соображения, которыми следует руководствоваться при принятии решения о классе (типе) наблюдаемого объекта, должны учитывать потери, связанные с правильными и ошибочными решениями. В качестве таких потерь можно использовать критерии среднего риска.

1. Задача разработки алгоритмов объединения результатов распознавания, полученных от различных источников информации. Следует также учесть, что эти источники могут иметь различные алфавиты классов.
2. Задача разработки специальных алгоритмов управления системой распознавания, для обеспечения в определённом смысле оптимальных условий её функционирования и достижения экстремальных значений показателей эффективности[13].

Упрощённую структурную схему распознающей системы можно представить на рисунке 2.1. На вход системы поступают реализации *X(t,θ)* сигналов (данных), полученных от распознаваемых объектов. Функцией обрабатывающего устройства является преобразование входных сигналов с целью выделения признаков распознавания на фоне мешающих воздействий помех. Предъявляемые к этому устройству требования сводятся к обеспечению необходимой точности, разрешающей способности и быстродействия.

Устройство (блок) сравнения в общем случае вычисляет вероятностные меры совпадения выделенных (сформированных) признаков распознавания объектов с эталонами, хранимыми в блоке памяти эталонов. Результаты работы этого блока могут быть представлены распределением апостериорной вероятности *Рар(Ак)* принадлежности распознаваемого объекта к предъявленным эталонам.



*Рисунок 2.1 — Упрощенна структурная схема распознающей системы*

Решающее устройство, в соответствии с принятым правилом решения (критерием распознавания), осуществляет сравнение предъявленной реализации с эталонами и выдает номер *k* класса распознаваемой цели или отказ в распознавании, если результаты сравнения не удовлетворяют критерию распознавания[13].

В данной работе основными взаимодействующими блоками алгоритма распознавания будут служить блок сравнения и блок памяти эталонов.

**3 Разработка компьютерной программы.**

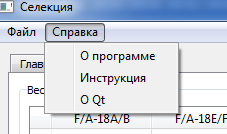
***3.1 Общие требования и характеристики системы***

Данная программа разрабатывается под операционной системой Windows с помощью приложения Qt Creator 2.0.1 на языке программирования C++. Система представляет собой приложение типа Desktop взаимодействующее с БД. Блок формирования признаков отсутствует. Вместо него имитация входных значений селектируемых объектов будет осуществляться путем построчного считывания данных из текстового файла с их последующей записью в БД. Вся результирующая информация будет также храниться в одной из таблиц БД.

Блок решающего устройства сильно упрощается. Его будет имитировать один из виджетов, показывающий в виде таблицы апостериорную вероятность исследуемого объекта по выбранным классам.

Обучение системы полуавтоматическое. Это значит, что оператор, управляющий данной программой, должен вручную устанавливать эталонные значения по каждому признаку. После подтверждения ввода программа пересчитывает весовые коэффициенты.

В системе предусмотрена справочная информация (Спрака), содержащаяся в главном меню. В ней будет содержаться информация по управлению программой (рисунок 3.1).



*Рисунок 3.1 — Справочная информация о программе*

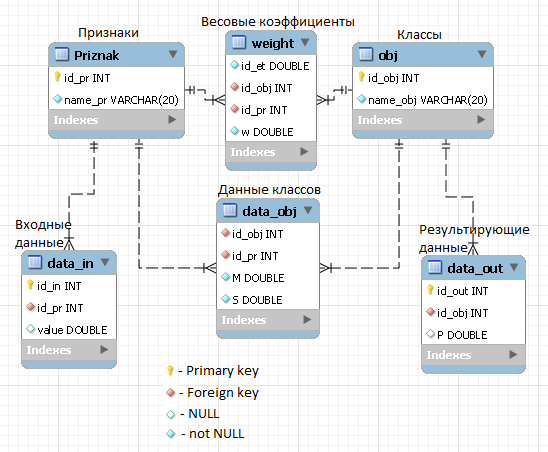
***3.2 База данных***

Как уже было сказано, приложение имеет тесное взаимодействие с БД. В данном случае она выполняет две основных функции:

* Хранение всех входных данных, информации о классах и результатов селекции;
* Имитация блока памяти эталонов.

Была разработана структура базы данных. Она приведена на рисунке 3.2.

Поясним значение каждой таблицы. Наименование признаков и классов содержится в таблицах Priznak и obj соответственно. Вся остальная информация (математическое ожидание и дисперсия) о классах содержится в таблице data\_obj. Эталоны и весовые коэффициенты по всем классам хранятся в структуре weight. И, наконец, входной вектор значений признаков заноситься в таблицу data\_in, а после прохождения его через алгоритм селекции, апостериорная вероятность по всем классам заноситься в таблицу data\_out.



*Рисунок 3.2 — Схема таблиц базы данных*

При проектировании в качестве СУБД используется SQLite. Выбор данной СУБД обусловлен тем, что SQLite легко встраивается в приложение. Эта система базируется на файлах, вследствие чего предоставляет довольно широкий набор инструментов для работы с ней, по сравнению с сетевыми СУБД. При работе с этой СУБД обращения происходят напрямую к файлам (в эти файлах хранятся данные), вместо портов и сокетов в сетевых СУБД. Именно поэтому SQLite очень быстрая, а также мощная благодаря технологиям обслуживающих библиотек.

Использование SQLite полностью оправдано в данном программном продукте и использует лучшие свои преимущества. Перечислим их:

1. Простота использования.
2. Отсутствие необходимости настройки сервера СУБД.
3. Возможность простого распространения со своим продуктом.
4. Полностью свободная лицензия.
5. Файловая структура – вся база данных состоит из одного файла, поэтому её очень легко переносить на разные машины.
6. Максимально возможный объём одной базы данных составляет 2 Тб.

Подключение автоматизированной системы к БД проходит с использованием драйвера QSQLITE. Далее указывается полное имя БД и после этого предпринимается попытка подключения к ней (Листинг 3.1). Следует также заметить, что, если БД с таким именем нет, то создается пустая (что для нашего случая является неприемлемым).

*Листинг 3.1— Создание соединения с БД.*

QSqlDatabase db = QSqlDatabase::addDatabase("QSQLITE");

db.setDatabaseName("C:\\selection.db");

if (!db.open()) {

…

}

Если по каким-то причинам соединения не происходит, то возникает окно с ошибкой с пояснением происходящего.

Взаимодействие с БД осуществляется при помощи библиотеки <QSqlQuery>.

***3.3 Программная реализация алгоритмов***

В главе 2 было рассмотрено два алгоритма. Перейдем к их программной реализации.

Сначала рассмотрим алгоритм селекции. Как видно из листинга 3.2, все обрабатываемые данные берутся прямо из запросов к БД. Одно из преимуществ SQLite – это быстрая обработка запросов типа SELECT. Данная часть программного кода выполняет следующие действия:

*Листинг 3.2— Реализация алгоритма селекции*

for (int j=0;j<3;j++){

if (classbox[j]->isChecked()){

f[j]=0;

query1.prepare("SELECT id\_et,w FROM weight WHERE id\_pr= :id1 and id\_obj= :id2 order by id\_et;");

query1.bindValue(":id1", i+1);

query1.bindValue(":id2", j+1);

query1.exec();

while (query1.next())

{

query2.prepare("SELECT S FROM data\_obj WHERE id\_pr= :id1 and id\_obj= :id2;");

query2.bindValue(":id1", i+1);

query2.bindValue(":id2", j+1);

query2.exec();

query2.next();

f[j] += (query1.value(1).toDouble())/(sqrt(2\*PI)\*query2.value(0).toDouble())\*exp(-(c-query1.value(0).toDouble())\*(c-query1.value(0).toDouble())/(2\*query2.value(0).toDouble()\*query2.value(0).toDouble()));

}

F[j]=F[j]\*f[j];

}

}

1. После поступления на вход алгоритма признака *c*, происходит накопление по всем выбранным классам функции правдоподобия f[j], j=1..n.
2. Далее это значение умножается с общей функцией F[j], j=1..n, и на вход поступает следующий признак. После этого повторяем пункт 1.

Далее происходит нормировка результатов (Листинг 3.3), и запись всех рассчитанных апостериорных вероятностей в БД. Чтобы видеть все изменения, таблицы входной и выходной информации обновляются после каждого обработанного объекта.

*Листинг 3.3— Нормировка апостериорной вероятности*

for (int i=0;i<3;i++){

if (F[i]!=1){

sum+=F[i];

}

}

for (int i=0;i<3;i++){

if (F[i]!=1){

F[i]=F[i]/sum;

}

}

for (int i=0;i<3;i++){

query.prepare("INSERT INTO data\_out (id\_obj, P) "

"VALUES (?, ?)");

query.addBindValue(i+1);

if (F[i]!=1){

query.addBindValue(F[i]);

}else

query.addBindValue(0);

query.exec();

}

resetTable1();

resetTable2();

Перейдем к рассмотрению программного кода алгоритма обучения. Он приведен на листинге 3.3. Сначала формируем знаменатель формулы 2.5 и записываем его в переменную sum, а следующим циклом получаем по этой же формуле новые значения весовых коэффициентов.

*Листинг 3.3— Реализация алгоритма самообучения*

QSqlQuery query,query1,query2;

query1.prepare("SELECT id\_et,w FROM weight WHERE id\_pr=:id1 and id\_obj= :id2 order by id\_et;");

query1.bindValue(":id1", ((PrEditor->currentIndex())+1));

query1.bindValue(":id2", ((NameEditor->currentIndex())+1));

query1.exec();

double sum=0;

while (query1.next())

{

double a=EtEditor->value();

double b=DispEditor->value();;

sum+=(query1.value(1).toDouble())/(sqrt(2\*PI)\*b)\*exp(-(query1.value(0).toDouble()-a)\*(query1.value(0).toDouble()-a)/(2\*b\*b));

}

query1.prepare("SELECT id\_et,w FROM weight WHERE id\_pr=:id1 and id\_obj= :id2 order by id\_et;");

query1.bindValue(":id1", ((PrEditor->currentIndex())+1));

query1.bindValue(":id2", ((NameEditor->currentIndex())+1));

query1.exec();

while (query1.next())

{

double rez;

double a=EtEditor->value();

double b=DispEditor->value();;

rez=query1.value(1).toDouble()+(query1.value(1).toDouble())/(sqrt(2\*PI)\*b)\*exp(-(query1.value(0).toDouble()-a)\*(query1.value(0).toDouble()-a)/(2\*b\*b))/sum;

query.prepare("UPDATE weight SET w = ? WHERE id\_pr = ? and id\_obj = ? and id\_et = ?;");

query.addBindValue(rez);

query.addBindValue((PrEditor->currentIndex())+1);

query.addBindValue((NameEditor->currentIndex())+1);

query.addBindValue(query1.value(0).toDouble());

query.exec();

}

После этого новые коэффициенты остается пронормировать (Листинг 3.4) и обновить таблицу БД.

*Листинг 3.4— Нормировка весовых коэффициентов*

query2.prepare("SELECT sum(w) FROM weight WHERE id\_pr= :id1 and id\_obj= :id2;");

query2.bindValue(":id1", ((PrEditor->currentIndex())+1));

query2.bindValue(":id2", ((NameEditor->currentIndex())+1));

query2.exec();

query2.next();

sum = query2.value(0).toDouble();

query1.prepare("SELECT id\_et,w FROM weight WHERE id\_pr=:id1 and id\_obj= :id2 order by id\_et;");

query1.bindValue(":id1", ((PrEditor->currentIndex())+1));

query1.bindValue(":id2", ((NameEditor->currentIndex())+1));

query1.exec();

while (query1.next())

{

query.prepare("UPDATE weight SET w = ? WHERE id\_pr = ? and id\_obj = ? and id\_et = ?;");

query.addBindValue(query1.value(1).toDouble()/sum);

query.addBindValue((PrEditor->currentIndex())+1);

query.addBindValue((NameEditor->currentIndex())+1);

query.addBindValue(query1.value(0).toDouble());

query.exec();

}