**Федеральное государственное образовательное**

**бюджетное учреждение**

**высшего образования**

**«ФИНАНСОВЫЙ УНИВЕРСИТЕТ**

**ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ**

**ФЕДЕРАЦИИ»**

**(Финансовый университет)**

**Факультет**

**информационных технологий и анализа больших данных**

**Направление «Прикладная математика и информатика»**

**Домашнее задание № 6**

«Логистическая регрессия, обучение SVM при помощи прямо-двойственного метода внутренней точки»

Студенты группы ПМ19-3:

Малахов Иван,

Верамьева Софья,

Кривоносова Дарья,

Соломатин Денис,

Ткаченко Анастасия,

Белуджев Артем,

Калимуллин Руслан

Руководитель:

Аксенов Дмитрий Андреевич

**Москва 2022**

Оглавление

[**1.**](#_gjdgxs) **Постановка задачи (физическая модель)** 4

[**2.**](#_30j0zll) **Математическая модель** 4

[**3.**](#_1fob9te) **Алгоритмы** 4

[**3.1.**](#_3znysh7) **Алгоритм 1** 4

[**3.1.1.**](#_tyjcwt) **Описание входных данных** 4

[**3.1.2.**](#_1t3h5sf) **Описание алгоритма решения** 5

[**3.1.3.**](#_2s8eyo1) **Описание выходных данных** 5

[**3.2.**](#_3rdcrjn) **Алгоритм 2** 6

[**3.2.1.**](#_lnxbz9) **Описание входных данных** 6

[**3.2.2.**](#_1ksv4uv) **Описание алгоритма решения** 7

[**3.2.3.**](#_2jxsxqh) **Описание выходных данных** 7

[**3.3.**](#_3j2qqm3) **Алгоритм 3** 7

[**3.3.1.**](#_1y810tw) **Описание входных данных** 8

[**3.3.2.**](#_4i7ojhp) **Описание алгоритма решения** 8

[**3.3.3.**](#_2xcytpi) **Описание выходных данных** 8

[**4.**](#_1ci93xb) **Варианты использования системы** 9

[**4.1.**](#_3whwml4) **ВИ 1** 9

[**4.2.**](#_2bn6wsx) **ВИ 2**

[**5.**](#_3as4poj) **Тестирование** 9

[**6.**](#_1pxezwc) **Работа методов** 10

[**6.1.**](#_49x2ik5) **Хорошая работа алгоритма** 10

[**6.2.**](#_2p2csry) **Плохая работа алгоритма** 10

[**6.3.**](#_147n2zr) **Нерабочий алгоритм** 10

[**7.**](#_3o7alnk) **Заключение** 10

1. **Постановка задачи (физическая модель)**

Компании ПМ19-3 поступил заказ на разработку модели по выявлению сахарного диабета у пациентов. Предварительная работа команды должна заключаться в реализации модели классификации на имеющихся данных и проведении её с оценки с помощью различных видов регрессии. Заказчик предоставил все необходимые данные в формате csv, а именно diabetes.csv

Условия заказчика:

Заказчик - ЦЕНТР КОРРЕКЦИИ ВЕСА СЕРСО - занимается вопросами поддержания здорового состояния тела и внешнего вида. В современных реалиях большое количество сахара в продукции оказывает огромную нагрузку на организм. Вымываются витамины группы В, нарушается баланс фосфора и кальция, возникают проблемы с сердцем, почками и другими жизненно-важными органами. И конечно употребление сахара становится причиной появления лишнего веса. Когда мы стимулируем организм сладким, оно сперва откладывается в печени в формате гликогена — это запасной источник энергии. Синтезировать в гликоген весь объем сахара, употребляемый за день среднестатистическим человеком невозможно, поэтому он начинает трансформироваться в жиры. Заказчику необходимо по вводимым параметрам устанавливать наличие сахарного диабета у пациента для дальнейшего незамедлительного лечения.

Постановка задачи:

На вводе подаются параметры: Pregnancies (Беременность), Glucose (Глюкоза), BloodPressure (Артериальное давление), SkinThickness (Толщина кожи), Insulin (Инсулин), BMI (), DiabetesPedigreeFunction (Родословная), Age (Возраст). Анализ и оценка производится через функции: 1.реализующие модель классификации на два класса на основе логистической регрессии;

2. реализующие модель классификации на два класса на основе логистической регрессии с радиальными базисными функциями; 3. реализующие модель классификации на два класса на основе логистической регрессии с регуляризацией L1; 4. реализующие модель классификации на два класса на основе метода опорных векторов; 5. реализующие модель классификации на два класса с применением прямо-двойственного метода внутренней точки для задачи обучения метода опорных векторов

1. **Математическая модель**

**Классификация** — разбиение множества объектов или наблюдений на априорно заданные группы, называемые [классами](https://wiki.loginom.ru/articles/class.html), внутри каждой из которых они предполагаются похожими друг на друга, имеющими примерно одинаковые свойства и признаки.

**Регрессионный анализ** — метод моделирования измеряемых данных и исследования их свойств. Данные состоят из пар значений зависимой переменной (переменной отклика) и независимой переменной (объясняющей переменной). Регрессионная модель есть функция независимой переменной и параметров с добавленной случайной переменной.

**Оценки качества классификации:**

* Матрица ошибок;
* Аккуратность — доля правильных ответов алгоритма;
* Точность — это доля правильных ответов модели в пределах класса;
* Полнота — это доля истинно положительных классификаций;
* F-мера - гармоническое среднее между точностью и полнотой;
* Кривая рабочих характеристик;
* Чувствительность к соотношению классов.

**Логистическая регрессия** — метод построения линейного классификатора, позволяющий оценивать апостериорные вероятности принадлежности объектов классам. Она позволяет вычислить вероятности возникновения интересующего события с помощью логистической функции. Логистическую регрессию относят к моделям бинарного выбора.

**Метод опорных векторов** применяется для решения задач классификации и регрессии. Основная идея метода заключается в построении гиперплоскости, разделяющей объекты выборки оптимальным способом. Алгоритм работает в предположении, что чем больше зазор между разделяющей гиперплоскостью и объектами разделяемых классов, тем меньше будет средняя ошибка классификатора.

**Регуляризация** — метод добавления некоторых дополнительных ограничений к условию с целью решить некорректно поставленную задачу или предотвратить переобучение. Чаще всего эта информация имеет вид штрафа за сложность модели.

**Параметр регуляризации** - коэффициент ( к ошибке прибавляются веса: либо сумма модулей либо сумма квадратов — L1 и L2 соответственно, они умножаются на коэффициент — параметр регуляризации. Чем он больше тем регуляризация сильнее. Если он равен 0 то регуляризации нет).

**BFGS с ограниченной памятью (L-BFGS)** — Программное обеспечение для крупномасштабной оптимизации с ограничениями

**LIBLINEAR —** библиотека для большой линейной классификации

**SAG** — минимизация конечных сумм с помощью стохастического среднего градиента.

**SAGA** — метод быстрого инкрементного градиента с поддержкой несильно выпуклых составных целей.

Ядро **RBF** является стационарным ядром, также известно как «квадратичное экспоненциальное» ядро. Параметризуется параметром шкалы длины *l > 0*, который может быть либо скаляром (изотропный вариант ядра), либо вектором с тем же числом измерений, что и входы X (анизотропный вариант ядра)

1. **Алгоритмы**
   1. **Алгоритм 1**

Логистическая регрессия. Алгоритм использовался с помощью библиотеки sklearn на языке программирования python

* + 1. **Описание входных данных**

На вход программы подается датасет данных о заболеваниях сахарного диабета, а также о соответствующих данных, которые могли повлиять на возникновение заболевания у человека. Формат данных pandas.dataframe.

В алгоритме также имеется возможность указать данные вручную, но данный способ будет достаточно трудоемок для пользователя, поскольку требует ручного ввода данных большого размера

Также на вход программа спрашивает у пользователя есть ли необходимость задавать дополнительные параметры для работы алгоритма. Дополнительные параметры: параметр регуляризации, необходимо ли строить график

* + 1. **Описание алгоритма решения**

Работа алгоритма начинается с того, что мы адаптируем данные для корректной реализации программы, таким образом необходимо выделить значения y, которые определяют наличие заболевания у человека, где 0 – отсутствие заболевания, а 1 – его наличие. Аналогичным образом выделяются параметры х. Затем производится разбивка данных на тестовую и обучающую выборки.

С помощью библиотеки мы строим модель логистической регрессии с указанием максимального количества итераций для того, чтобы модель успела обучиться, если набор данных этого требует, а также с заранее заданным параметром регуляризации. На основе этой модели производятся вычисления предсказанных значений y по тестовой выборке x

* + 1. **Описание выходных данных**

В формате выходных данных мы получаем первостепенно предсказанные значения y, матрицу смежности, которая может дать нам четкую картину, насколько хорошо работает классификация. Также в формате выходных данных мы получаем полный отчет по классификации, в котором содержится информация о необходимых метриках для оценки работы классификации. Помимо этого, представлены графики о работе регрессии

* 1. **Алгоритм 2**

Логистическая регрессия с радиальными базисными функциями. Данный алгоритм позволяет пользователю задавать базисные функции для вычисления логистической регрессии. Алгоритм использовался с помощью библиотеки sklearn на языке программирования python

* + 1. **Описание входных данных**

На вход программы подается датасет данных о заболеваниях сахарного диабета, а также о соответствующих данных, которые могли повлиять на возникновение заболевания у человека. Формат данных pandas.dataframe.

Далее на вход программа спрашивает у пользователя есть ли необходимость задавать дополнительные параметры для работы алгоритма. Дополнительные параметры: базисная функция {newton-cg, lbfgs, liblinear, sag, saga} и необходимо ли строить график, по умолчанию алгоритмом используется метод lbfg

* + 1. **Описание алгоритма решения**

Выделяются значения y, которые определяют наличие заболевания у человека, где 0 – отсутствие заболевания, а 1 – его наличие. Аналогичным образом выделяются параметры х. Затем производится разбивка данных на тестовую и обучающую выборки.

С помощью библиотеки мы строим модель логистической регрессии с указанием заранее определенной базисной функцией. На основе этой модели производятся вычисления предсказанных значений y по тестовой выборке x

* + 1. **Описание выходных данных**

В формате выходных данных мы получаем предсказанные значения y, матрицу смежности, которая может дать нам четкую картину, насколько хорошо работает классификация. После этого мы получаем полный отчет по классификации, в котором содержится информация о необходимых метриках для оценки работы классификации. Также в формате выходных данных представлены графике о работе регрессии

* 1. **Алгоритм 3**

Логистическая регрессия с регуляризацией L1. Данный алгоритм предусмотрен для соответствующей регуляризации, поэтому он имеет возможность задавать соответствующие параметры базисной функции. Алгоритм использовался с помощью библиотеки sklearn на языке программирования python.

* + 1. **Описание входных данных**

На вход программы подается датасет данных о заболеваниях сахарного диабета, а также о соответствующих данных, которые могли повлиять на возникновение заболевания у человека. Формат данных pandas.dataframe.

Помимо этого, на вход программа спрашивает у пользователя есть ли необходимость задавать дополнительные параметры для работы алгоритма. Дополнительные параметры: базисная функция {liblinear, saga} и необходимо ли строить график, по умолчанию алгоритмом используется метод liblinear. Данные базисные функции могут работать с использованием вида регуляризации L1

* + 1. **Описание алгоритма решения**

Выделяются значения y, которые определяют наличие заболевания у человека, где 0 – отсутствие заболевания, а 1 – его наличие. Аналогичным образом выделяются параметры х. Затем производится разбивка данных на тестовую и обучающую выборки.

С помощью библиотеки мы строим модель логистической регрессии с указанием заранее определенной базисной функцией на основе вида регуляризации. По этой модели производятся вычисления предсказанных значений y по тестовой выборке x

* + 1. **Описание выходных данных**

В формате выходных данных мы получаем предсказанные значения y, матрицу смежности, которая может дать нам четкую картину, насколько хорошо работает классификация. Далее мы получаем полный отчет по классификации, в котором содержится информация о необходимых метриках для оценки работы классификации. Также в формате выходных данных представлены графике о работе регрессии

1. **Варианты использования системы**
   1. **ВИ 1**
2. Использовать данные по умолчанию? 1 - да/0 - нет
3. Введите значения х для обучающей выборки через пробел
4. Введите значения х для тестовой выборки через пробел
5. Введите значения y для тестовой выборки через пробел
6. Введите значения y для тестовой выборки через пробел
7. Есть необязательные параметры? 1 - да/0 - нет
8. Введите параметр регуляризации {l1, l2, elasticnet, none}
9. Построить график? 1 - да/0 - нет
   1. **ВИ 2**
10. Использовать данные по умолчанию? 1 - да/0 - нет
11. Введите значения х для обучающей выборки через пробел
12. Введите значения х для тестовой выборки через пробел
13. Введите значения y для тестовой выборки через пробел
14. Введите значения y для тестовой выборки через пробел
15. Есть необязательные параметры? 1 - да/0 - нет
16. Введите параметр регуляризации {l1, l2, elasticnet, none}
17. Построить график? 1 - да/0 - нет

Можно задавать разные виды регуляризации. Выбор алгоритма зависит от исследуемых данных и параметров регуляризации:

* newton-cg - ['l2', 'none']
* lbfgs - ['l2', 'none']
* liblinear - ['l1', 'l2']
* sag - ['l2', 'none']
* saga - ['elasticnet', 'l1', 'l2', 'none']
* Параметр регуляризации по умолчанию none, имеется возможность задать его самостоятельно, однако необходимо учитывать, что для разного вида регуляризации необходим соответствующий набор данных.

1. **Тестирование**
   1. Функция, реализующая модель классификации на два класса на основе логистической регрессии.

Изображение выглядит как текст, квитанция

Автоматически созданное описание

Вывод:

1) матрица: 89 результатов предсказано верно по 0, 31 - верно по 1, 10 - неверно по 0, 24 - неверно по 1; модель тяготеет к классу 0, при этом ошибок больше у класса 1 (примерно 32% против 27%)

2) точность сотавляет 78%, это означает, что на выборке из 100 значений 78 окажутся верными, а 22 неверными

3) Также точность (precision) показывает, что 76% по 1 можно интерпретировать как долю объектов, названных классификатором положительными и при этом действительно являющимися положительными

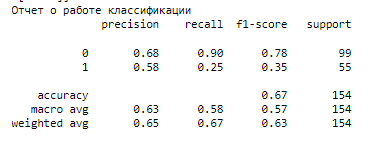
4) Полнота показывает, какую долю объектов положительного класса из всех объектов положительного класса нашел алгоритм, в нашем случае это 56%.

* 1. Функция, реализующая модель классификации на два класса на основе логистической регрессии с радиальными базисными функциями.

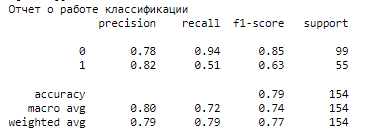


Однозначно можно сделать вывод о том, что первая модель лучше, чем модель с заданной базисной функцией sag, даже не смотря на то, что наша модель показала хорошие результаты по классу 0, она не показала должного результата по классу 1, очевидно, что модель тяготеет к классу 0, что делает её недостаточно хорошей, что подтверждают показатели точности в 68% и показатель полноты в 27%

* 1. Функция, реализующая модель классификации на два класса на основе логистической регрессии с регуляризацией L1.

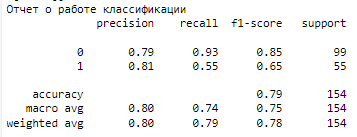
Данный вид модели с параметром L1 и функцией saga также не демонстриует удовлетворительного результата, модель схожа по результатам со второй исследуемой моделью

* 1. Функция, реализующая модель классификации на два класса на основе метода опорных векторов



Прямой метод опорных векторов показывает отличные показатели, которые лучше линейных моделей. Об этом говорит и матрица смежности и показатели точности, уровня доверия и полноты

* 1. Функция, реализующая модель классификации на два класса с применением прямо-двойственного метода внутренней точки для задачи обучения метода опорных векторов.



Показатели методом опорных векторов c двойственным методом показывают результаты, которые явно лучше предыдущих моделей, такие как: 92 точных результатов по 0, 30 точных результат по 1, при этом достоверность равна 79%, точность 81%, а плотность 55%, к тому же доли неправильных предсказаний составляют 25 и 7 соответственно

1. **Работа методов**
   1. **Хорошая работа алгоритма**

* Задача выпуклого квадратичного программирования хорошо изучена и имеет единственное решение.
* Метод опорных векторов эквивалентен двухслойной нейронной сети, где число нейронов на скрытом слое определяется автоматически как число опорных векторов.
* Принцип оптимальной разделяющей гиперплоскости приводит к максимизации ширины разделяющей полосы, а следовательно, к более уверенной классификации.
  1. **Плохая работа алгоритма\ Нерабочий алгоритм**
* Неустойчивость к шуму: выбросы в исходных данных становятся опорными объектами-нарушителями и напрямую влияют на построение разделяющей гиперплоскости.
* Не описаны общие методы построения ядер и спрямляющих пространств, наиболее подходящих для конкретной задачи.
* Нет отбора признаков.
* Необходимо подбирать константу Cz
* C при помощи кросс-валидации.

1. **Заключение**

На основе полученных данных заказчик сможет устанавливать наличие сахарного диабета у пациента для дальнейшего незамедлительного лечения основываясь на полученных данных. Наилучшим решением для поставленной задачи будет прямой метод опорных векторов так как действенность метода равна 79%, точность 81%, а плотность 55% что позволяет достаточно точно определить наличие диабета в крови. Также, с помощью нашей программы заказчик имеет возможность построить график, для возможности отслеживания развития заболевания.

В дальнейшем мы могли бы продолжить наше сотрудничество и усовершенствовать работу программы с учетом более полной информации при заказе.