

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное
образовательное учреждение
высшего образования
«КАЗАНСКИЙ (ПРИВОЛЖСКИЙ) ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ И
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА ТЕХНОЛОГИЙ ПРОГРАММИРОВАНИЯ

Направление: 01.03.04 – «Прикладная математика»

Курсовой проект по лабораторному практикуму ИКТ

ТЕМА

Работа завершена:

“ ____ ” _____ 2019 г. _____ (И.О. Фамилия)

Работа допущена к защите:

Научный руководитель

ученая степень, ученое звание,

должность

“ ____ ” _____ 2019 г. _____ (И.О. Фамилия)

Казань-2019

Задание к курсовому проекту по лабораторному практикуму ИКТ.

Создать web-сервис для обработки временных рядов методами машинного обучения

1. Выбрать предметную область для исследования. Согласовать выбор с преподавателем
2. Создать java-приложение,
3. Оформить отчет
4. Пройти процедуру защиты проекта в назначенное аудиторное время

Требования к функциональным возможностям разрабатываемого java-приложения.

Приложение с графическим пользовательским интерфейсом должно иметь «главное меню», обеспечивающее доступ ко всем пользовательским функциям.

Основные функции приложения

(пункты с маркерами *, ** - повышенной сложности, в случае их реализации повышают оценку работы).

1. Загрузка статистической информации по выбранной предметной области. Данные могут быть загружены одним или нескольких форматах (.csv, .xml, .json, .txt, ...)
 - а) из файла, указанного пользователем (выбор файла в стандартном диалоговом окне),
 - б) с сетевого ресурса по фиксированной ссылке
 - в*) с сетевого ресурса по указанной пользователем ссылке
 - г**) обеспечить возможность явной настройки пользователем структуры прочитанных данных
2. Просмотр загруженных данных в режиме таблицы или списка
- 3*. Выбор диапазона загруженных данных для анализа и построения графика
- 4*. Расчет характерных показателей предметной области (набор величин зависит от предметной области, согласовывается с преподавателем)
5. Отображение загруженных данных на графике

График представляет собой ломаную, в которой координата y узловой точки – значение временного ряда, координата x – время фиксации значения временного ряда. Должна быть возможность выбора цвета графика

*Режим виртуального маркера, позволяет движением/щелчком мышки отображать на графике маркер текстового объекта с информацией о значении абсциссы и ординаты точки на графике. Предусмотреть возможность стирать нарисованное маркером.

6. Анализ и прогнозирование временных рядов предметной области

По N периодам наблюдения методом наименьших квадратов построить линию тренда.

6.1. Создать однослойный персептрон со случайной первоначальной инициализацией весов, разделяющий точки подаваемые данные на два класса

6.2. Создать двухслойный персептрон, разделяющий точки подаваемых данных на два класса

6.3* Использовать элементы метода группового учета аргументов (МГУА) для прогнозирования показателей предметной области.

7. Управление статистическими данными предметной области помощью СУБД

Спроектировать, разместить в СУБД базу данных, не менее чем из трех взаимосвязанных таблиц, позволяющих аутентифицировать пользователя и логгировать его действия в приложении.

База данных хранит информацию об аккаунтах пользователей, загружаемых им данных, результатах проведенных анализов и прогнозов.

Приложение должно обеспечить при выборе соответствующей команды меню или кнопки

а) просмотр данных

б) сохранение в базу данных результатов последнего сеанса

Содержание отчета по курсовому проекту

Отчет по курсовому проекту должен включать следующие основные разделы:

1. Титульный лист
2. Содержание (включает порядок расположения отдельных частей курсового проекта с указанием страниц, на которых соответствующий раздел начинается).
3. Введение
4. Основная часть
5. Заключение
6. Список использованных источников

Правила оформления курсового проекта

Текст работы следует печатать на одной стороне листа белой бумаги формата А4 с размерами полей: сверху - 20 мм, снизу - 20 мм, справа - 15 мм, слева - 30 мм. Шрифт – Times New Roman, размер 14 пт. Абзацный отступ - 1,25 см. Цвет шрифта должен быть черным.

- заголовки основной части работы (введение, названия разделов, заключение, список использованных источников) пишутся без отступа, без точки в конце и с прописной буквы;
- заголовки подразделов и пунктов печатаются с прописной буквы без точки в конце;
- если заголовок включает несколько предложений, их разделяют точками; переносы в заголовках не допускаются; расстояния между заголовками и текстом должны быть не менее 2-х интервалов;
- разделы, подразделы, пункты и подпункты начинаются с арабских цифр, разделенных точками. Если раздел или подраздел имеет только один пункт, или пункт имеет один подпункт, то его нумеровать не надо. Текст работы должен быть выровнен по ширине;

- нумерация страниц работы выполняется арабскими цифрами в внизу в центре страницы. Нумерация страниц начинается с титульного листа, но номера страниц на титульном листе не ставятся. Поэтому номера страниц появляются только, начиная с содержания;

Страницы текста и включенные в работу иллюстрации и таблицы должны соответствовать формату А4 (210 x 297 мм).

Фамилии и собственные имена, названия учреждений в тексте приводят на языке оригинала. Из сокращенных названий учреждений и предприятий следует употреблять только общеизвестные. Малоизвестные сокращения необходимо расшифровывать при первом упоминании.

Иллюстрации. Все иллюстрации (фотографии, графики, чертежи, схемы, диаграммы и другие графические материалы) именуются в тексте рисунками. Иллюстрации следует располагать в тексте непосредственно после первого их упоминания или на следующей странице. На все иллюстрации должны быть даны ссылки в тексте работы. Для каждой иллюстрации приводится подрисуночная подпись.

Каждая подрисуночная подпись начинается с номера рисунка, например, «Рисунок 1». Далее без кавычек следует название рисунка. Иллюстрации следует нумеровать арабскими цифрами порядковой нумерацией в пределах всего проекта. В подрисуночной подписи необходимо привести расшифровку изображенных объектов в том случае, если на одном рисунке их несколько. При этом каждый объект на рисунке обозначается строчной буквой латинского алфавита, с соответствующей ссылкой в подрисуночной подписи. В конце подрисуночной подписи точка не ставится.

Рисунки, представленные в приложении, нумеруются в следующей последовательности: сначала указывается буква, обозначающая приложение, далее точка и номер рисунка, например, «Рисунок А.1».

Таблицы. Значительный по объему цифровой материал, используемый в курсовом проекте, оформляют в виде таблиц.

Нумерация таблиц в основной части работы обозначается арабскими

цифрами и является сквозной внутри всей работы или внутри раздела. Например: «Таблица 1» или «Таблица 1.1».

Нумерация таблиц приложений отдельная и состоит из буквы, обозначающей приложение, и цифры - номера таблицы. Например: «Таблица А.1». На все таблицы должны быть приведены ссылки в тексте.

Заголовки граф и строк таблицы следует писать с прописной буквы, а подзаголовки граф - со строчной буквы, если они составляют одно предложение с заголовком, или с прописной буквы, если они имеют самостоятельное значение. В конце заголовков и подзаголовков таблиц точки не ставят.

Таблицу в зависимости от ее размера помещают под текстом, в котором впервые дана ссылка на нее, или на следующей странице, а при необходимости в приложении. Допускается помещать таблицу вдоль длинной стороны листа выпускной работы.

Если строки или графы выходят за формат таблицы, ее делят на части, помещая одну часть под другой или рядом, при этом в каждой части таблицы повторяют ее головку. При делении на части допускается строки или столбцы головки заменять соответственно номером граф и строк. Слово «Таблица» указывают один раз над первой частью таблицы, над другими частями пишут слова «Продолжение таблицы» с указанием номера (обозначения) таблицы.

Формулы. Расчетные формулы должны помещаться на отдельной строке. Формулы нумеруются арабскими цифрами, помещаемыми в круглых скобках справа от формулы. Нумерация формул в пределах раздела, напр.: 4.2. - (формула вторая, четвертого раздела). После формулы ставится запятая и с новой строки после слова «где» идет расшифровка каждого обозначения. Расшифровке подлежат только обозначения, встречающиеся впервые. Ссылки на формулы в тексте обязательны.

Пример постановки задач, схема отчета и справочный материал по курсовому проекту

Содержание.

1. Платформа Java FX.
 - 1.1. Стандартные диалоговые окна.
 - 1.2. Графики.
 - 1.3. Создание меню.
2. Методы машинного обучения.
 - 2.1. Многослойный персептрон.
 - 2.2. Библиотека TensorFlow.
 - 2.3. Свёрточные нейронные сети. RGB-сети.
3. СУБД PostgreSQL.
 - 3.1. Создание базы данных. Хранение информации в JSON .
 - 3.2. Запросы. Изменения данных.
 - 3.3. Транзакции.
4. Web-программирование. Архитектура клиент-сервер.
 - 4.1. Сокеты.
 - 4.2. Протокол HTTP.
 - 4.3. Архитектура web-сервера. Сервлеты.

Глава 1. Создание графического интерфейса для взаимодействия с пользователем.

- 1.1. Стандартные диалоговые окна. Калькулятор для вычисления курса валют.
Задачи.
 1. Создать метод, принимающий на вход значение суммы в рублях и конвертирующий эту сумму в запрашиваемую валюту.
 2. В диалоговом окне предусмотреть кнопки для списка валют, ввода значения, подтверждения действий пользователя.
- 1.2. График изменения временного ряда.
Задачи.
 1. Считайте данные из файла. Для выбора имени файла используйте соответствующее стандартное диалоговое окно.
 2. Постройте ломаную, в которой координата y узловой точки – значение временного ряда, координата x – время фиксации значения временного ряда.

3. Создайте виртуальный маркер, используя движения мышки.
 4. Предусмотрите возможность стирать нарисованное маркером.
- 1.3. Использование меню.
- Задачи.
1. На панели с калькулятором предусмотрите меню с двумя пунктами.
 2. При выборе первого пункта должна происходить отрисовка графика, причём должна быть возможность выбора цвета.графика.
 3. При выборе второго пункта должна появляться возможность использования виртуального маркера, а также динамически должен появляться новый пункт меню, использование которого предоставит возможность прекратить использование маркера и очистки окна от следов маркера.

Глава 2. Применение искусственных нейронных сетей для анализа временных рядов финансового рынка.

- 2.1. Обучение однослойного персептрона, разделяющего значения цен закрытия, лежащих выше и ниже линии тренда.
- Задачи.
1. По 30 периодам наблюдения цен закрытия методом наименьших квадратов постройте линию тренда.
 2. Создайте однослойный персептрон со случайной первоначальной инициализацией весов, разделяющий точки, лежащие выше и ниже линии тренда.

Справочный материал.

Однослойный персептрон, реализующий логическое «или».

```
public class Perceptron {
    double[] x;
    double y;
    double[] w;
    double[][] pat = { { 0, 0, 0 }, { 0, 1, 1 }, { 1, 0, 1 }, {
1, 1, 1 } };
    public Perceptron() {
        x = new double[2];
        w = new double[x.length];
        for (int i = 0; i < x.length; i++) {
            w[i] = Math.random() * 0.2 + 0.1;
        }
    }

    public void cy() {
        y = 0;
    }
}
```



```

        for (int i = 0; i < x.length; i++) {
            y += x[i] * w[i];
        }
        if (y > 0.5)
            y = 1;
        else
            y = 0;
    }

    public void study() {
        double gEr = 0;
        int m=0;
        do {
            gEr = 0;
            for (int p = 0; p < pat.length; p++) {
                x = java.util.Arrays.copyOf(pat[p],
pat[p].length - 1);
                cy();
                double er = pat[p][2] - y;
                gEr += Math.abs(er);
                for (int i = 0; i < x.length; i++) {
                    w[i] += 0.1 * er * x[i];
                }
            }
            m++;
        } while (gEr!=0);
        System.out.println("m="+m);
    }

    public void test() {
        study();
        for (int p = 0; p < pat.length; p++) {
            x = java.util.Arrays.copyOf(pat[p], pat[p].length
- 1);

            cy();
            System.out.println("y="+y);
        }
    }

    public static void main(String[] args) {

        new Perceptron().test();

    }
}

```

Пример программы, считывающей из файла.

```
import java.io.BufferedReader;
```

```
import java.io.File;
import java.io.FileInputStream;
import java.io.FileWriter;
import java.io.InputStreamReader;

public class readFile {

    public static void main(String[] args) {
        BufferedReader br;
        double[] a = new double[10];
        int i = 0; // счетчик цикла
        try {
            br = new BufferedReader(new InputStreamReader(new
FileInputStream(new File("a.csv"))));
            String line;
            // Считываем файл построчно
            while ((line = br.readLine()) != null) {
                line.replace(',', '.');
                a[i++] = Double.parseDouble(line);
            }
            System.out.println(a[1]);
        } catch (Exception e) {
            e.printStackTrace();
        }

        try {
            FileWriter writer = new FileWriter("b.csv", false);
            // запись всей строки
```

```

        for (int j = 0; j < i; j++) {
            String text = String.valueOf(a[j]).replace('.', ',');
            writer.write(text);

            // запись по символам
            writer.append('\n');
        }

        writer.flush();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
}

```

2.2. Обучение двухслойного персептрона.

Задача.

1. Создайте двухслойный персептрон, на вход которого подаются цена закрытия и цена открытия. Двухслойный персептрон должен разделять подаваемые данные на два класса. В первом классе находятся наблюдения, в которых для белых свечей цена закрытия больше половины цены открытия не менее, чем на d единиц, а для чёрных свечей разность цены закрытия и половины цены открытия меньше d . Во втором классе находятся все остальные наблюдения.

2.3 Использование элементов МГУА для прогнозирования цены закрытия финансового инструмента.

Задачи.

1. Разделите выборку данных из 60 наблюдений на две части: обучающую и тестовую.
2. Постройте 4 семейства моделей:

$$Y \approx b_0 + b_1 \cdot X_i;$$

$$Y \approx b_0 + b_1 \cdot X_i + b_2 \cdot X_j;$$

$$Y \approx b_0 + b_1 \cdot X_i^2;$$

$$Y \approx b_0 + b_1 \cdot X_i^2 + b_2 \cdot X_j^2,$$

где $i, j = \overline{1,3}$, $Y = C$ - цена закрытия периода, $X_1 = O$ - цена открытия периода, $X_2 = H$ - наибольшая цена за период, $X_3 = L$ - наименьшая цена за период.

3. Для каждой построенной модели вычислите коэффициент детерминации.
4. Из всех построенных моделей выберете две наилучшие, руководствуясь значением коэффициента детерминации. Обозначим их Y_1 и Y_2 . Для прогноза цены закрытия используйте модель $C \approx b_0 + b_1 \cdot Y_1^2 + b_2 \cdot Y_2^2$.
5. Оцените отклонение прогнозного значения цены закрытия от статического, используя формулу $\Delta = \frac{|C_{\text{статистическое}} - C_{\text{прогнозное}}|}{C_{\text{статистическое}}} \cdot 100\%$ на данных обучающей и тестовой выборки.

Справочный материал.

```
package modeling;

import Jama.Matrix;

public class Matrice1 {
    double[] z=new double[3];

    public static void main(String[] args) {
        double[][] x=new
double[][]{{2,4,5,6,8},{4,7,3,2,1},{2,40,8,90,1},{4,5,60,7,80}};
        double[] y=new double[] { 20 , 40, 570,760,800};
        double[][]f=new double[3][5];
        for (int i = 0; i < 5; i++) {
            f[0][i]=1;
        }
        for (int i = 0; i < 4; i++) {
            for (int j = i+1; j < 4; j++) {
                System.out.println("i="+i);
                System.out.println("j="+j);
                for (int k = 0; k < 5; k++) {
                    f[1][k]=x[i][k];
                    f[2][k]=x[j][k];
                    System.out.println("f[1]["+k+"]="+f[1][k]);
                    System.out.println("f[2]["+k+"]="+f[2][k]);
                }
                Matrice1 H= new Matrice1();
                H.test(f, y);
                H.R(f, y);
            }
        }
    }
}
```

```

        public void test(double[][]ff, double[]yy){
            Matrix A1=new Matrix(ff);
            A1.print(10, 2);
            Matrix B1=A1.transpose();
            Matrix F1=A1.times(B1);
            Matrix F4=F1.inverse();
            Matrix F2=F4.times(A1);
            Matrix C=new Matrix(yy,5);
            Matrix F3=F2.times(C);
            z=F3.getColumnPackedCopy();
            for (int i = 0; i < 3; i++)
                System.out.println("z["+i+"]="+z[i]);
        }
    public double R(double[][]ff, double[]yy){
        double r=0, S1=0, S2=0, S3=0;
        double[] u1=new double[5];
        double[] u2=new double[5];
        for (int i = 0; i < 5; i++)
            {S3+=yy[i];
            u1[i]=0;}
        S3=S3/5;
        for (int m = 0; m < 5; m++){
            u1[m]=(z[0]+z[1]*ff[1][m]+z[2]*ff[2][m]-
yy[m])*(z[0]+z[1]*ff[1][m]+z[2]*ff[2][m]-yy[m]);
            S1+=u1[m];
            u2[m]=(S3-yy[m])*(S3-yy[m]);
            S2+=u2[m];
        }
        r=1-S1/S2;
        System.out.println("r="+r);
        return r;
    }
}

package modeling;

public class Matrice2 {
    double[] z=new double[3];

    public static void main(String[] args) {
        double[][] x=new
double[][]{{2,4,5,6,8},{4,7,3,2,1},{2,40,8,90,1},{4,5,60,7,80}};
        double[] y=new double[] { 20 , 40, 570,760,800};
        double[][]f=new double[3][5];
        for (int i = 0; i < 5; i++) {
            f[0][i]=1;
        }
        for (int i = 0; i < 4; i++) {
            for (int j = i+1; j < 4; j++) {
                System.out.println("i="+i);
                System.out.println("j="+j);
            }
        }
    }
}

```

```

        for (int k = 0; k < 5; k++) {
            f[1][k]=x[i][k]*x[i][k];
            f[2][k]=x[j][k]*x[j][k];
            System.out.println("f[1] [" +k+"]="+f[1][k]);
            System.out.println("f[2] [" +k+"]="+f[2][k]);
        }
        Matrice1 H= new Matrice1();
        H.test(f, y);
        H.R(f, y);
    }
}
}

```

Глава 2 (альтернативная). Обработка статистических данных финансового рынка.

2.1. Обучение однослойного персептрона, разделяющего значения цен закрытия, лежащих выше и ниже линии тренда.

Задачи.

1. По 30 периодам наблюдения цен закрытия методом наименьших квадратов постройте линию тренда.
2. Создайте однослойный персептрон со случайной первоначальной инициализацией весов, разделяющий точки, лежащие выше и ниже линии тренда.
- 2.2. Прогнозирование цены закрытия.

Задачи.

1. Постройте методом наименьших квадратов уравнение множественной линейной регрессии:

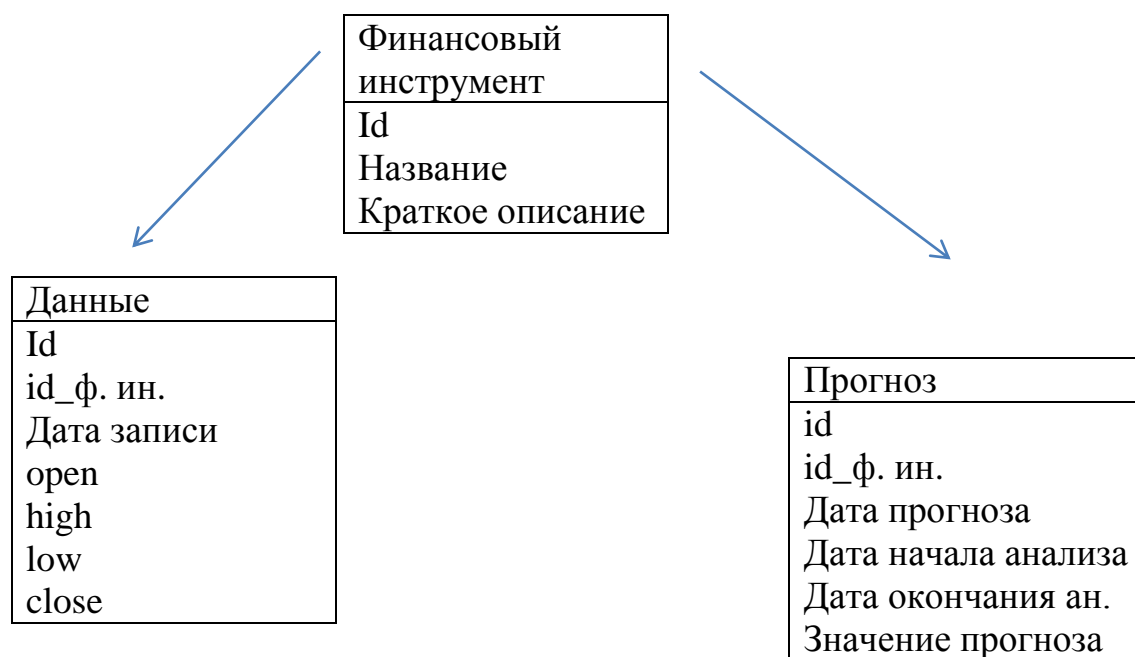
$$C = \beta_0 + \beta_1 \cdot H + \beta_2 \cdot O + \varepsilon$$

2. Вычислите и проанализируйте значения коэффициента детерминации R^2 .
3. Используйте построенную модель для прогноза цены закрытия на следующий день рассматриваемого периода.

Глава 3. Управление статистическими данными финансового рынка помощью СУБД PostgreSQL

3.1. Проектирование ER-модели базы данных, создание таблиц в pgAdmin с автоматической генерацией первичного ключа. Задачи.

1. Создайте ER-модель. Примерный образец.



2. Создайте в базе данных не менее трёх взаимосвязанных таблиц с автоматической генерацией первичного ключа.

3.2. Управление базой данных через java-программы с использованием JDBC.

Задачи.

1. Установите подключение к БД.
2. Протестируйте следующие запросы:
 - 1) Insert;
 - 2) Insert с использованием цикла;
 - 3) Insert (одновременная запись в две таблицы);
 - 4) Delete;
 - 5) Update;
 - 6) Select;
 - 7) Select (из двух таблиц).

3.3. Заполнение таблиц базы данных.

Задачи.

1. Заполните одну из таблиц, считывая данные из файла.
2. Заполните не менее трёх строк таблицы с прогнозом, вычисляя прогноз в java программе и передавая значение в базу данных.

Пример программы, считывающей данные из файла и записывающей в базу данных

```
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStreamReader;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.text.ParseException;
import java.text.SimpleDateFormat;

public class readFile {

    public static void main(String[] args) {
        String url = "jdbc:postgresql://localhost:5432/finance";
        try (Connection conn = DriverManager.getConnection(url,
"postgres", "post")) {
            conn.setAutoCommit(false);

            conn.setTransactionIsolation(Connection.TRANSACTION_READ_COM
MITTED);

            BufferedReader br;
            br = new BufferedReader(new InputStreamReader(new
FileInputStream(new File("pairs.csv"))));
            String line;
            String[] v;
            String sql = "insert into eur_usd (pair,
open_value, close_value,day) values (?, ?, ?, ?)";
            PreparedStatement ps = conn.prepareStatement(sql);

            while ((line = br.readLine()) != null) {
                line=line.replace(',', ' ');
                v = line.split(";");
                ps.setString(1,v[0]);
                ps.setDouble(2, Double.parseDouble(v[2]));

                try {
                    // String -> java.sql.Date
                    // 12.10.2010 -> java.util.Date -> 2010-
10-12 -> java.sql.Date
                    java.util.Date d;
                    d = new
SimpleDateFormat("dd.MM.yyyy").parse(v[1]);
```



```
String sd = new SimpleDateFormat("yyyy-MM-dd").format(d);  
ps.setDate(4,  
java.sql.Date.valueOf(sd));  
} catch (ParseException e) {
```