Факультет електроніки та комп’ютернихтехнологій

Кафедра радіофізики та комп’ютерних технологій

**Звіт**

про виконання лабораторної роботи №3

на тему “Використання методу найменших квадратів для згладжування даних”

Виконав:

Кравченко Ярослав

Група: ФЕІ-34

Перевірив:

доц. Любунь З. М.

Львів – 2019

**Мета:** реалізувати програму первісного оброблення даних з використанням методу найменших квадратів для послідовності значень отриманих у результаті табуляції функції при =1, 10, 100 на інтервалі[0 ;0.5] з кроком 0.1, де випадкове число з рівномірним розподілом на інтервалі [0; 0.5] .Обчислити значення апроксимуючих функцій для випадку лінійної та квадратичної емпіричної формули при різних значеннях

**Програмна реалізація**

<html>

<head>

<script type="text/javascript" src="cramer.js"></script>

<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>

<script type="text/javascript">

google.charts.load('current', {'packages': ['line']});

google.charts.setOnLoadCallback(drawChart);

var x0 = -5;

var xn = 5;

var step = 0.1;

var xArr = [];

var yArr = [];

var yrArr = [];

var data;

var alpha = 100;

var points = [];

class MyPoint {

constructor(x,y){

this.x = x;

this.y = y;

}

}

function Fr(x){

return (2 - x - 2\*x\*x) + Math.random() \* alpha;

}

function F(x) {

return 2 - x - 2\*x\*x;

}

function linearSquares(p){

var n = p.length;

var sx = 0;

var sy = 0;

var sxx = 0;

var sxy = 0;

for (var i = 0; i < n; i++){

var pt = p[i];

sx += pt.x;

sy += pt.y;

sxx += pt.x \* pt.x;

sxy += pt.x + pt.y;

}

var k = (n \* sxy - sx \* sy)/(n \* sxx - sx \* sx);

var b = (sy \* sxx - sxy \* sx)/(n \* sxx - sx \* sx);

return new MyPoint(k,b);

}

function nonLinear(p){

var n = p.length;

var sx4 = 0;

var sx3 = 0;

var sx2 = 0;

var sx = 0;

var sx2y = 0;

var sy = 0;

var sxy = 0;

for(var i = 0; i < n; i++){

var pt = p[i];

sx4 += Math.pow(pt.x, 4);

sx3 += Math.pow(pt.x, 3);

sx2 += Math.pow(pt.x, 2);

sx += pt.x;

sy += pt.y;

sx2y += Math.pow(pt.x, 2) \* pt.y;

sxy += pt.x \* pt.y;

}

var a = [

[sx4, sx3, sx2],

[sx3, sx2, sx],

[sx2, sx, n]

];

var b = [sx2y, sxy, sy];

return cramersRule(a, b);

}

function drawChart() {

data = new google.visualization.DataTable();

data.addColumn('number', 'X');

data.addColumn('number', 'Without random');

data.addColumn('number', 'With random');

data.addColumn('number', 'Linear Squares');

data.addColumn('number', 'Non-Linear');

var i = 0;

for (var x = x0; x < xn; x += step, i++)

{

xArr[i] = x;

yArr[i] = F(x);

yrArr[i] = Fr(x);

points.push(new MyPoint(x,yrArr[i]));

}

var ls = linearSquares(points);

var nls = nonLinear(points);

console.log(nls);

for (var i = 0; i < points.length; i++){

var x = xArr[i];

var ly = ls.x \* x + ls.y;

var nly = nls[0] \* x \* x + nls[1] \* x + nls[2];

data.addRow([xArr[i], yArr[i], yrArr[i], ly, nly]);

}

var options = {

chart: {

title: 'Методи первісних обробок даних'

},

width: 900,

height: 500,

axes: {

x: {

0: {side: 'top'}

}

}

};

var chart = new google.charts.Line(document.getElementById('line\_top\_x'));

chart.draw(data, google.charts.Line.convertOptions(options));

}

</script>

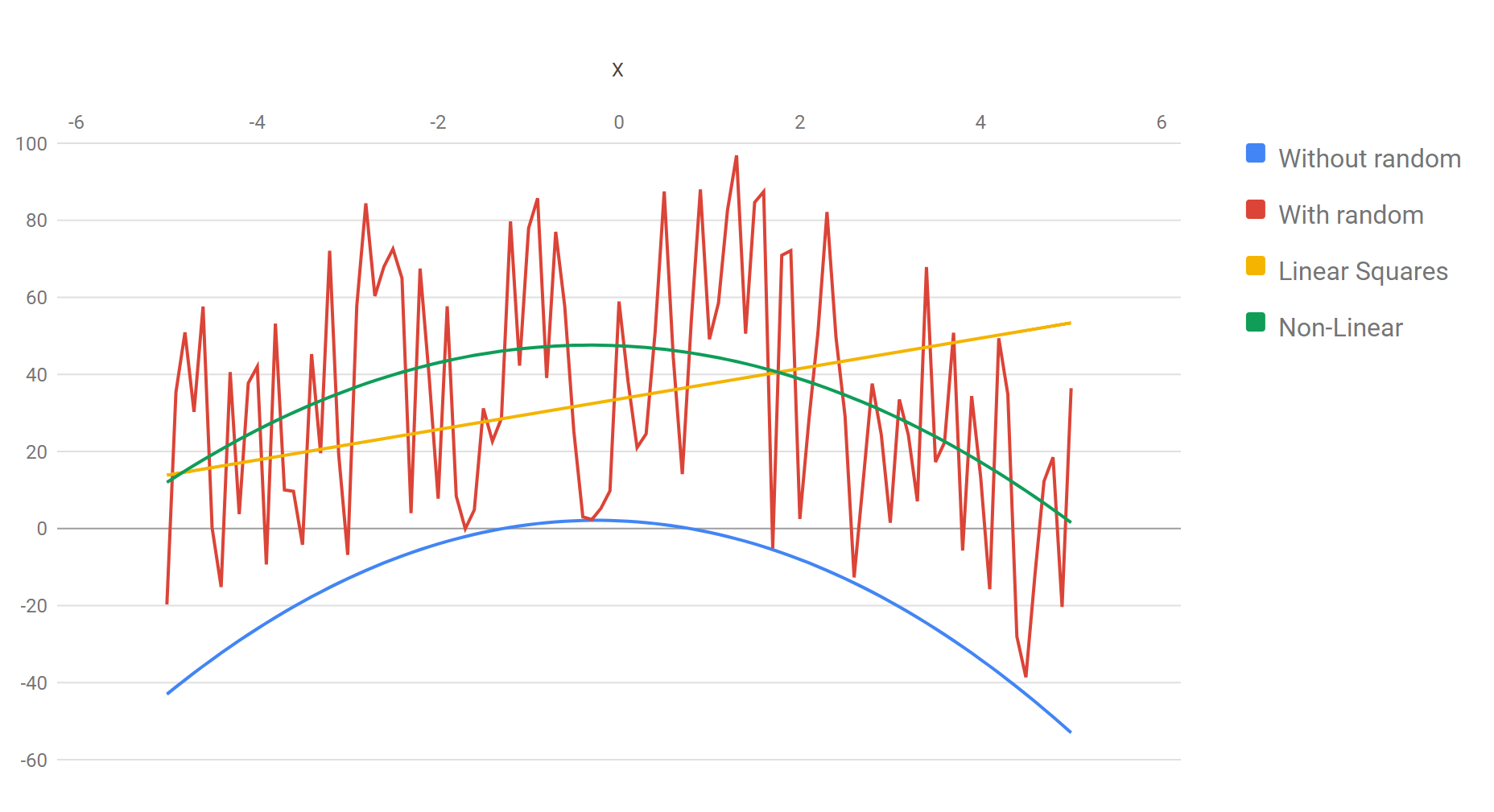
</head>

<body>

<div id="line\_top\_x"></div>

</body>

</html>



**Висновок**

Під час виконання даної лабораторної роботи було реалізовано програму первісної обробки даних з використанням методу найменших квадратів. Знайдено коефіцієнти полінома для квадратичної та лінійної емпіричної залежності для різних значень параметра а, побудовано їхні графіки. Найкращий результат згладжування для заданої функції у квадратичної емпіричної формули, так як сума квадратів відхилень у неї майже втричі менша порівняно з лінійною.