%-- 18/08/2020 11:41 --%

find\_lines

%-- 22/09/2020 21:30 --%

Adjustment

P = eye(n);

Adjustment

%-- 27/10/2020 11:51 --%

i=1;

i=+1

i+=1

i++

A = ]1 2 3 4; 0 1 2 3; 1 -1 0 0; -1 -2 3 2];

A = [1 2 3 4; 0 1 2 3; 1 -1 0 0; -1 -2 3 2];

b = [3;1;1;3];

b\A

A\b

85\*7

51\*32

ans+595

ans/32

2227/32

85/32

L = [1 0 0;4 1 0;12 2.65625 1];

U = [1 7 -4; 0 -32 -7; 0 0 69.59375];

B = L\*U;

B

51-(85\*25)/32

B = L\*U;

B

62+4\*51

266\*2.65625

8+12\*51-266\*2.65625

ans/15.40625

A = B;

b = [-51; 62; 8];

A\b

(266-5.62\*25)/-32

1/1.5

4+124/2

2\*66

/3

2\*66/3

ans+14

58/(4/3)

(66+43.5)/1.5

(124+73)/2

A\b

Untitled

sum(r,r0)

sum(r,r0')

sum(r',r0')

commandhistory

Untitled

w = subs(ans,x,1)

Untitled3

Untitled4

HW1\_3

%-- 04/11/2020 22:31 --%

Q1\_main

Untitled

log(12)

Q2

Untitled

fxy = diff(sin(x)\* log(x\*y),x,x,y)

fxx = diff(sin(x)\* log(x\*y),x,2)

fxx = diff(sin(x)\* log(x\*y),x,x)

Q2

double(total)

vpa(total,3)

Q2

%-- 06/11/2020 20:12 --%

Q2

%-- 07/11/2020 00:26 --%

Q3

eqns=[x^2+y^2=4,exp(x)+y=1];

eqns=[x^2+y^2==4,exp(x)+y==1];

clear

eqns=[x^2+y^2==4,exp(x)+y==1];

syms x y

eqns=[x^2+y^2==4,exp(x)+y==1];

S = solve(eqns,[x y])

S.x

S.y

Q3

syms x y

eqns=[x^2+y^2==4,exp(x)+y==1];

S = solve(eqns,[x y])

S.x

S.y

Q3

Untitled2

S.x^2+S.y^2

exp(S.x)+y

exp(S.x)+S.y

Q3

1e5

EqSolve

Q3

1e-5

ans-0.00001

EqSolve

Q22

Q1\_main

%-- 08/11/2020 14:59 --%

Q22

simple(taylor1)

expand(taylor1)

T1

T3

HW1\_Q2

T3

expand(T3)

expand(taylor3)

vpa(expand(taylor3,4))

expand(taylor3,5)

format short

expand(taylor3,5)

expand(taylor3)

expand(T3)

double(ans)

format long

digits(5)

expand(T3)

HW1\_Q2

digits(5)

T1 = expand(vpa(taylor1))

T2 =expand(vpa(taylor2))

T3 =expand(vpa(taylor3))

HW1\_Q2

500/17

200/21

30/22

x1 = 29.411764705882351;

x2=9.523809523809524;

x3 = 1.363636363636364;

x11 = (500+2\*x2+3\*x3)/17

x22=(200+5\*x1+2\*x3)/21

x33=(30+5\*x1+5\*x2)/22

(500+2\*x22+3\*x33)/17

(200+5\*x11+2\*x33)/21

(30+5\*x11+5\*x22)/22

45q6

45/6

2+4\*7.5

5\*7.5

ans+80

ans/12

(45-8\*32)/6

ans\*4-9.791666666666666+2

digits(3)

ans

(80+5\*-35.166666666666664)/12

(45-8\*-148.458333333333)/6

48/8

(80-2.5)/12

(-2+6+ans)/4

(45-6\*ans)/8

(80+5\*2.61458333333333)/12

(-2+3.6640625+7.75607638888889)/4

(45-6\*ans)/8

(80+5\*2.35503472222222)/12

pdetool

%-- 17/11/2020 07:19 --%

syms x

f = x^5+2\*x^3-1+1;

subs(f,-1)

(-1)^5

2\*(-1)^3

f=x^5+2\*x^3-x+1;

subs(f,-1)

subs(f,0)

subs(f,-0.5)

double(ans)

-1+0.5

ans/2

subs(f,-0.25)

double(ans)

-1-0.5

ans/2

subs(f,-0.75)

double(ans)

(-1-0.75)/2

double(subs(f,ans))

f=x+sin(x)

sin(0)

sin(PI/2)

sin(pi/2)

sin(pi)

0.5+sin(0.5)

0.25+sin(0.25)

diff(f)

double(subs(f,0.25))

g=cos(x)+1;

double(subs(g,0.25))

0.25-0.497403959254523/1.96891242171064

double(subs(f,-0.00262878824359597))

double(subs(g,-0.00262878824359597))

-0.00262878824359597+0.00525757345947402/2

-1.51\*10^-9

double(subs(f,-1.5138589600644e-09))

double(subs(g,-1.5138589600644e-09))

-1.5138589600644e-09+3.0277179201288e-09/2

-sin(0.25)

-sin(ans)

f = x+sin(x)\*exp(x)/sqrt(1-x^3);

subs(f,0)

subs(f,-1)

double(subs(f,-1))

double(subs(f,1))

double(subs(f,-0.5))

double(subs(f,0.5))

x1=-0.5;

x2=0.5;

x\_new=x1-((x2-x1)/(subs(f,x2)-subs(f,x1)))\*subs(f,x1)

x\_new=double(x1-((x2-x1)/(subs(f,x2)-subs(f,x1)))\*subs(f,x1))

double(subs(f,x\_new))

x1=-0.251907213134179;

x\_new=double(x1-((x2-x1)/(subs(f,x2)-subs(f,x1)))\*subs(f,x1))

double(subs(f,x\_new))

x1=ans;

x\_new=double(x1-((x2-x1)/(subs(f,x2)-subs(f,x1)))\*subs(f,x1))

double(subs(f,x\_new))

x0=-0.5;

x1=0.5;

x = double(x1-(x1-x0)\*subs(f,x1)/(subs(f,x1)-subs(f,x0))

x = double(x1-(x1-x0)\*subs(f,x1)/(subs(f,x1)-subs(f,x0)))

x0=x1;

x1=-0.134689229559212;

x = double(x1-(x1-x0)\*subs(f,x1)/(subs(f,x1)-subs(f,x0)))

x0=x1;

x1=-0.03456988667307;

x = double(x1-(x1-x0)\*subs(f,x1)/(subs(f,x1)-subs(f,x0)))

HW2\_Q3\_main

solve(((2/3)^k)\*1.2/(1/3)=e-4)

solve(((2/3)^k)\*1.2/(1/3)==e-4)

solve(((2/3)^k)\*1.2/(1/3)==10e-4)

solve(((2/3)^k)\*1.2/(1/3)==10e-4,k)

solve(((2/3)^k)\*1.2/(1/3)==10e-4,'k')

syms k

solve(((2/3)^k)\*1.2/(1/3)==10e-4,'k')

solve(((2/3)^k)\*1.2/(1/3)==10e-4,k)

double(ans)

solve(((2/3)^k)\*1.2/(1/3)==10^-4,k)

solve(((2/3)^k)\*1.2/(1/3)==10^(-4),k)

double(ans)

roundup(ans)

round(ans)

solve(x^2-2==0)

solve('x^2-2==0')

solve('x^2-2==0',x)

A = [4 -1 1;-1 3 1; 1 1 5];

D = diag(diag(A));

C\_l = -tril(A,-1);

C\_u = -triu(A,1);

G = D\(C\_l+C\_u);

c = [12;1;-14];

x0 = [4;3;-3];

[n,er] = itr\_est(x0,G,c,10^-4,2)

x1 = G\*x0 + c

x1 = [4.5,2.667,-4.2]';

norm\_x1\_x0 = norm(x1-x0,'inf');

c = D\b;;

c = D\c;;

[n,er] = itr\_est(x0,G,c,10^-4,2)

HW2\_Q4\_main

est\_error

HW2\_Q4\_main

set(0,'RecursionLimit',67000)

HW2\_Q4\_main

%-- 19/11/2020 15:46 --%

HW2\_Q4\_main

%-- 19/11/2020 15:53 --%

HW2\_Q3\_main

HW2\_Q4\_main

set(0,'RecursionLimit',5000)

sol\_itr = iterative\_solver(G,c,x0,5000 )

set(0,'RecursionLimit',5000)

set(0,'RecursionLimit',50000)

sol\_itr = iterative\_solver(G,c,x0,50000 );

sol\_itr = iterative\_solver(G,c,x0,49999 );

set(0,'RecursionLimit',70000)

sol\_itr = iterative\_solver(G,c,x0,65000 );

HW2\_Q4\_main

%-- 19/11/2020 18:07 --%

HW2\_Q4\_main

HW2\_Q3\_main

for i=1:10

x0 = x0 + b -A\*x0;

end

A = [A(2,:);A(3,:);A(1,:)];

x0 = zeros(n,1)

for i=1:10

x0 = x0 + b -A\*x0;

end

HW2\_Q3\_main

max(abs(eig(G\_r)))

max(abs(eig(A)))

max(abs(eig(I-A)))

max(abs(eig(eye(n)-A)))

max(abs(eig(eye(n)-5\*A)))

max(abs(eig(eye(n)-0.5\*A)))

HW2\_Q3\_main

sol

AB(i,:)

AB(i,:)-(AB(i,j)/AB(j,j)\*AB(:,j))

T = AB(i,:)-(AB(i,j)/AB(j,j)\*AB(:,j));

sol

S=AB(i,:);

T = AB(i,:)-(AB(i,j)/AB(j,j)\*AB(:,j));

sol

S=AB(i,:);

T = AB(i,:)-(AB(i,j)/AB(j,j)\*AB(:,j));

sol

HW2\_Q3\_main

gauss\_elim

sol

gauss\_elim

time

syms y t

f=(2/t)+(t^2)\*exp(t)

f=(2/t)\*y+(t^2)\*exp(t)

subs(f,[t,y],[0,1])

subs(f,{t,y},{0,1})

subs(f,[t,y],[0,1])

subs(f,[t,y],[1,0])

z = (t^2)\*(exp(t)-exp(1))

double(answer)

double(ans)

double(z)

double(t^2\*(exp(t) - 3060513257434037/1125899906842624)

double(t^2\*(exp(t) - 3060513257434037/1125899906842624)))

double(t^2\*(exp(t) - 3060513257434037/1125899906842624))

subs(f,[y,t],[1.1,exp(0.1)])

double(ans)

double(ans\*0.1+exp(0.1))

double(subs(f,[t,y],[1.1,exp(0.1)]))

double(exp(0.1)+ans\*0.1)

double(subs(f,[t,y],[1.2,ans]))

1.66961517389601+0.1\*ans

double(subs(f,[t,y],[1.3,ans]))

2.42598120641941+0.1\*ans

double(subs(f,[t,y],[1.4,ans]))

3.41932022115774+ans\*0.1

double(subs(f,[t,y],[1.5,ans]))

4.70261373196754+0.1\*ans

double(subs(f,[t,y],[1.6,ans]))

6.33800893705594+ans\*0.1

double(subs(f,[t,y],[1.7,ans]))

8.39823635483308+0.1\*ans

double(subs(f,[t,y],[1.8,ans]))

10.9682349574932+0.1\*ans

double(subs(f,[t,y],[1.9,ans]))

14.1470135090178+0.1\*ans

subs(z,t,2)

double(ans)

double(subs(z,t,1.4))

double(subs(z,t,1.1))

double(subs(z,t,1))

double(subs(z,t,1.1))

0.7\*exp(1)

0.1\*exp(1)

double(subs(f,[t,y],[1.1,exp(0.1)]))

(0.5\*exp(1)+0.5\*5.644)\*0.1

k1=double(subs(f,[t,y],[1.1,ans]))

0.418114091422952

0.418114091422952+0.1\*k1

k2=double(subs(f,[t,y],[1.2,ans]))

y2 = 0.418114091422952+(0.5\*k1+0.5\*k2)\*0.1

k1=double(subs(f,[t,y],[1.2,y2]))

k2=double(subs(f,[t,y],[1.3,y2+k1\*0.1]))

y3 = y2 +(0.5\*k1+0.5\*k2)\*0.1

k1=double(subs(f,[t,y],[1.3,y3]))

k2=double(subs(f,[t,y],[1.4,y3+k1\*0.1]))

y4 = y3 +(0.5\*k1+0.5\*k2)\*0.1

k1=double(subs(f,[t,y],[1.4,y4]))

k2=double(subs(f,[t,y],[1.5,y4+k1\*0.1]))

y5 = y4 +(0.5\*k1+0.5\*k2)\*0.1

k1=double(subs(f,[t,y],[1.5,y5]))

k2=double(subs(f,[t,y],[1.6,y5+k1\*0.1]))

y6 = y5 +(0.5\*k1+0.5\*k2)\*0.1

k1=double(subs(f,[t,y],[1.6,y6]))

k2=double(subs(f,[t,y],[1.7,y6+k1\*0.1]))

y7 = y6 +(0.5\*k1+0.5\*k2)\*0.1

k1=double(subs(f,[t,y],[1.7,y7]))

k2=double(subs(f,[t,y],[1.8,y7+k1\*0.1]))

y8 = y7 +(0.5\*k1+0.5\*k2)\*0.1

k1=double(subs(f,[t,y],[1.8,y8]))

k2=double(subs(f,[t,y],[1.9,y8+k1\*0.1]))

y9 = y8 +(0.5\*k1+0.5\*k2)\*0.1

k1=double(subs(f,[t,y],[1.9,y9]))

k2=double(subs(f,[t,y],[2,y9+k1\*0.1]))

y10 = y9 +(0.5\*k1+0.5\*k2)\*0.1

%-- 24/11/2020 12:48 --%

HW2\_Q3\_main

norm(G\_r,'inf')

HW2\_Q3\_main

norm(G\_r,'inf')

HW2\_Q3\_main

clc

gauss\_elim(A ,b , 1)

gauss\_elim(A ,b , 0)

HW2\_Q3\_main

clear

A=[0 1.055555556 0.0231672 2.130617275 -2.133375281 7.587033497 -15.29437708 40.69993331 -98.29592331 249.7991406 -623.9663625

0 0.0521262 1.163436299 -1.064864888 4.601040065 -9.080009972 25.38250066 -61.03758593 156.7276962 -390.764009 982.8207234

0 0.053497942 0.053701163 1.279875298 -2.292981936 8.489256729 -20.05032533 54.17983316 -133.9007416 337.4250601 -831.3427046

0 0 0.05647005 0.094257876 1.301830529 -1.927588543 7.187612337 -14.83207386 36.77277081 -77.47045291 161.69261

0 0 0.101051669 0.050226648 1.934211993 -3.968950328 14.50296283 -37.71037163 107.5234862 -289.9710957 793.3065277

0 0.106995885 0.099705329 1.997136402 -3.558848253 13.20426056 -32.22660186 89.51090002 -231.4677294 611.9470788 -1600.263657

0 0.108367627 1.996562177 -1.770789963 7.844691609 -15.47349723 43.48895371 -105.9334807 276.5224458 -706.4821786 1830.86492

0 1.997256516 0.065073075 4.038058996 -4.009674825 14.38644347 -29.07754374 77.69777099 -189.437652 487.9971853 -1243.78043

];

norm(A(:,2)-A(:,1))

norm(A(:,3)-A(:,2))

clc

Untitled

A=[0 0.527777778 0.533569578 0.670848193 0.671894953 0.741743113 0.742505961 0.787077087 0.792993299 0.824360471 0.834059659

0 0.0260631 0.316922175 0.322727826 0.470668944 0.474110483 0.568516339 0.582210048 0.648789319 0.670148832 0.719618943

0 0.026748971 0.040174262 0.200184077 0.210169875 0.332112543 0.362726517 0.454261132 0.49462144 0.564882666 0.608094965

0 0 0.014117513 0.032958503 0.126105146 0.190276355 0.279402301 0.351959518 0.427391041 0.496863936 0.559371768

0 0 0.025262917 0.044172707 0.171339287 0.2254654 0.347230449 0.406561319 0.510543877 0.566202436 0.653431257

0 0.053497942 0.078424275 0.327155006 0.340994554 0.528544251 0.55775814 0.698414274 0.734692137 0.843220941 0.881063978

0 0.054183813 0.553324358 0.567999931 0.823398458 0.829672534 0.992205994 1.006396017 1.12095436 1.141723286 1.227259674

0 0.998628258 1.014896527 1.278130971 1.282626605 1.41736477 1.419012654 1.504773858 1.511204256 1.571586216 1.581802138

];

Untitled

A=[0 0.521341463 0.533350684 0.665883284 0.67172425 0.737582734 0.742300916 0.783402078 0.792263221 0.821044626 0.832741575

0 0.025745257 0.309867432 0.322389241 0.463576058 0.473690852 0.561803148 0.580799585 0.642500899 0.667576709 0.713758941

0 0.026422764 0.039844834 0.19439517 0.209551035 0.324838828 0.360289954 0.446628071 0.490481735 0.557319327 0.602655924

0 0 0.013775283 0.032271323 0.121627209 0.186322238 0.272934655 0.345716426 0.420041954 0.48920823 0.551702501

0 0 0.024650506 0.043478108 0.165231812 0.222059656 0.338466704 0.401309548 0.500574659 0.559848467 0.642988268

0 0.052845528 0.077812066 0.318154496 0.340113167 0.517359322 0.555279122 0.686738426 0.730739367 0.831670226 0.876005585

0 0.053523035 0.541216373 0.567248555 0.811149572 0.828928219 0.980647477 1.004740171 1.110171466 1.138970004 1.217228937

0 0.986449864 1.014353642 1.2686084 1.282180056 1.409335769 1.418604515 1.497702055 1.510262576 1.565228281 1.580242153

];

Untitled

A = [0 0.521341463 0.660134004 0.733388289 0.778623536 0.809629255 0.833499546 0.853146128 0.874456636 0.89662983 0.91821302

0 0.297542179 0.454584044 0.551558953 0.618028729 0.669201641 0.711319803 0.757005075 0.804539766 0.850809611 0.893830377

0 0.181543839 0.307565326 0.399505581 0.474556151 0.538170121 0.614151846 0.692238655 0.766649719 0.834754854 0.895645599

0 0.094646331 0.183363288 0.269966263 0.349148056 0.464963741 0.581305519 0.68770027 0.781990078 0.864445067 0.936059393

0 0.049343057 0.132244966 0.218114121 0.395300761 0.559792448 0.700607048 0.81957937 0.920228118 1.005699893 1.078532032

0 0.07857011 0.165864058 0.457217856 0.680412569 0.852260576 0.98840057 1.098787708 1.18986574 1.265963073 1.330108814

0 0.094484891 0.623115089 0.903565748 1.088128282 1.22260293 1.326280803 1.409043276 1.476652765 1.53276719 1.579855307

0 1.035708756 1.311305597 1.457516154 1.553736255 1.623843465 1.677895039 1.721042548 1.756290178 1.785544954 1.810093942

];

Untitled

A=[0 0.521341463 0.533350684 0.665883284 0.67172425 0.737582734 0.742300916 0.783402078 0.792263221 0.821044626 0.832741575

0 0.025745257 0.309867432 0.322389241 0.463576058 0.473690852 0.561803148 0.580799585 0.642500899 0.667576709 0.713758941

0 0.026422764 0.039844834 0.19439517 0.209551035 0.324838828 0.360289954 0.446628071 0.490481735 0.557319327 0.602655924

0 0 0.013775283 0.032271323 0.121627209 0.186322238 0.272934655 0.345716426 0.420041954 0.48920823 0.551702501

0 0 0.024650506 0.043478108 0.165231812 0.222059656 0.338466704 0.401309548 0.500574659 0.559848467 0.642988268

0 0.052845528 0.077812066 0.318154496 0.340113167 0.517359322 0.555279122 0.686738426 0.730739367 0.831670226 0.876005585

0 0.053523035 0.541216373 0.567248555 0.811149572 0.828928219 0.980647477 1.004740171 1.110171466 1.138970004 1.217228937

0 0.986449864 1.014353642 1.2686084 1.282180056 1.409335769 1.418604515 1.497702055 1.510262576 1.565228281 1.580242153

];

Untitled

A = [0 0.527777778 0.533569578 0.670848193 0.671894953 0.741743113 0.742505961 0.787077087 0.792993299 0.824360471 0.834059659

0 0.0260631 0.316922175 0.322727826 0.470668944 0.474110483 0.568516339 0.582210048 0.648789319 0.670148832 0.719618943

0 0.026748971 0.040174262 0.200184077 0.210169875 0.332112543 0.362726517 0.454261132 0.49462144 0.564882666 0.608094965

0 0 0.014117513 0.032958503 0.126105146 0.190276355 0.279402301 0.351959518 0.427391041 0.496863936 0.559371768

0 0 0.025262917 0.044172707 0.171339287 0.2254654 0.347230449 0.406561319 0.510543877 0.566202436 0.653431257

0 0.053497942 0.078424275 0.327155006 0.340994554 0.528544251 0.55775814 0.698414274 0.734692137 0.843220941 0.881063978

0 0.054183813 0.553324358 0.567999931 0.823398458 0.829672534 0.992205994 1.006396017 1.12095436 1.141723286 1.227259674

0 0.998628258 1.014896527 1.278130971 1.282626605 1.41736477 1.419012654 1.504773858 1.511204256 1.571586216 1.581802138

];

Untitled

A = [0 1.055555556 0.0231672 2.130617275 -2.133375281 7.587033497 -15.29437708 40.69993331 -98.29592331 249.7991406 -623.9663625

0 0.0521262 1.163436299 -1.064864888 4.601040065 -9.080009972 25.38250066 -61.03758593 156.7276962 -390.764009 982.8207234

0 0.053497942 0.053701163 1.279875298 -2.292981936 8.489256729 -20.05032533 54.17983316 -133.9007416 337.4250601 -831.3427046

0 0 0.05647005 0.094257876 1.301830529 -1.927588543 7.187612337 -14.83207386 36.77277081 -77.47045291 161.69261

0 0 0.101051669 0.050226648 1.934211993 -3.968950328 14.50296283 -37.71037163 107.5234862 -289.9710957 793.3065277

0 0.106995885 0.099705329 1.997136402 -3.558848253 13.20426056 -32.22660186 89.51090002 -231.4677294 611.9470788 -1600.263657

0 0.108367627 1.996562177 -1.770789963 7.844691609 -15.47349723 43.48895371 -105.9334807 276.5224458 -706.4821786 1830.86492

0 1.997256516 0.065073075 4.038058996 -4.009674825 14.38644347 -29.07754374 77.69777099 -189.437652 487.9971853 -1243.78043

];

Untitled

G = [0 1.055555556 0.0231672 2.130617275 -2.133375281 7.587033497 -15.29437708 40.69993331 -98.29592331 249.7991406 -623.9663625

0 0.0521262 1.163436299 -1.064864888 4.601040065 -9.080009972 25.38250066 -61.03758593 156.7276962 -390.764009 982.8207234

0 0.053497942 0.053701163 1.279875298 -2.292981936 8.489256729 -20.05032533 54.17983316 -133.9007416 337.4250601 -831.3427046

0 0 0.05647005 0.094257876 1.301830529 -1.927588543 7.187612337 -14.83207386 36.77277081 -77.47045291 161.69261

0 0 0.101051669 0.050226648 1.934211993 -3.968950328 14.50296283 -37.71037163 107.5234862 -289.9710957 793.3065277

0 0.106995885 0.099705329 1.997136402 -3.558848253 13.20426056 -32.22660186 89.51090002 -231.4677294 611.9470788 -1600.263657

0 0.108367627 1.996562177 -1.770789963 7.844691609 -15.47349723 43.48895371 -105.9334807 276.5224458 -706.4821786 1830.86492

0 1.997256516 0.065073075 4.038058996 -4.009674825 14.38644347 -29.07754374 77.69777099 -189.437652 487.9971853 -1243.78043

];

Untitled

G = [0 1.055555556 0.0231672 2.130617275 -2.133375281 7.587033497 -15.29437708 40.69993331 -98.29592331 249.7991406 -623.9663625

0 0.0521262 1.163436299 -1.064864888 4.601040065 -9.080009972 25.38250066 -61.03758593 156.7276962 -390.764009 982.8207234

0 0.053497942 0.053701163 1.279875298 -2.292981936 8.489256729 -20.05032533 54.17983316 -133.9007416 337.4250601 -831.3427046

0 0 0.05647005 0.094257876 1.301830529 -1.927588543 7.187612337 -14.83207386 36.77277081 -77.47045291 161.69261

0 0 0.101051669 0.050226648 1.934211993 -3.968950328 14.50296283 -37.71037163 107.5234862 -289.9710957 793.3065277

0 0.106995885 0.099705329 1.997136402 -3.558848253 13.20426056 -32.22660186 89.51090002 -231.4677294 611.9470788 -1600.263657

0 0.108367627 1.996562177 -1.770789963 7.844691609 -15.47349723 43.48895371 -105.9334807 276.5224458 -706.4821786 1830.86492

0 1.997256516 0.065073075 4.038058996 -4.009674825 14.38644347 -29.07754374 77.69777099 -189.437652 487.9971853 -1243.78043

];

Untitled

G = [0 0.527777778 0.533569578 0.670848193 0.671894953 0.741743113 0.742505961 0.787077087 0.792993299 0.824360471 0.834059659

0 0.0260631 0.316922175 0.322727826 0.470668944 0.474110483 0.568516339 0.582210048 0.648789319 0.670148832 0.719618943

0 0.026748971 0.040174262 0.200184077 0.210169875 0.332112543 0.362726517 0.454261132 0.49462144 0.564882666 0.608094965

0 0 0.014117513 0.032958503 0.126105146 0.190276355 0.279402301 0.351959518 0.427391041 0.496863936 0.559371768

0 0 0.025262917 0.044172707 0.171339287 0.2254654 0.347230449 0.406561319 0.510543877 0.566202436 0.653431257

0 0.053497942 0.078424275 0.327155006 0.340994554 0.528544251 0.55775814 0.698414274 0.734692137 0.843220941 0.881063978

0 0.054183813 0.553324358 0.567999931 0.823398458 0.829672534 0.992205994 1.006396017 1.12095436 1.141723286 1.227259674

0 0.998628258 1.014896527 1.278130971 1.282626605 1.41736477 1.419012654 1.504773858 1.511204256 1.571586216 1.581802138

];

Untitled

G = [0 0.521341463 0.533350684 0.665883284 0.67172425 0.737582734 0.742300916 0.783402078 0.792263221 0.821044626 0.832741575

0 0.025745257 0.309867432 0.322389241 0.463576058 0.473690852 0.561803148 0.580799585 0.642500899 0.667576709 0.713758941

0 0.026422764 0.039844834 0.19439517 0.209551035 0.324838828 0.360289954 0.446628071 0.490481735 0.557319327 0.602655924

0 0 0.013775283 0.032271323 0.121627209 0.186322238 0.272934655 0.345716426 0.420041954 0.48920823 0.551702501

0 0 0.024650506 0.043478108 0.165231812 0.222059656 0.338466704 0.401309548 0.500574659 0.559848467 0.642988268

0 0.052845528 0.077812066 0.318154496 0.340113167 0.517359322 0.555279122 0.686738426 0.730739367 0.831670226 0.876005585

0 0.053523035 0.541216373 0.567248555 0.811149572 0.828928219 0.980647477 1.004740171 1.110171466 1.138970004 1.217228937

0 0.986449864 1.014353642 1.2686084 1.282180056 1.409335769 1.418604515 1.497702055 1.510262576 1.565228281 1.580242153

];

Untitled

G=[0 0.521341463 0.660134004 0.733388289 0.778623536 0.809629255 0.833499546 0.853146128 0.874456636 0.89662983 0.91821302

0 0.297542179 0.454584044 0.551558953 0.618028729 0.669201641 0.711319803 0.757005075 0.804539766 0.850809611 0.893830377

0 0.181543839 0.307565326 0.399505581 0.474556151 0.538170121 0.614151846 0.692238655 0.766649719 0.834754854 0.895645599

0 0.094646331 0.183363288 0.269966263 0.349148056 0.464963741 0.581305519 0.68770027 0.781990078 0.864445067 0.936059393

0 0.049343057 0.132244966 0.218114121 0.395300761 0.559792448 0.700607048 0.81957937 0.920228118 1.005699893 1.078532032

0 0.07857011 0.165864058 0.457217856 0.680412569 0.852260576 0.98840057 1.098787708 1.18986574 1.265963073 1.330108814

0 0.094484891 0.623115089 0.903565748 1.088128282 1.22260293 1.326280803 1.409043276 1.476652765 1.53276719 1.579855307

0 1.035708756 1.311305597 1.457516154 1.553736255 1.623843465 1.677895039 1.721042548 1.756290178 1.785544954 1.810093942

];

Untitled

A=[0 1.055555556 0.0231672 2.130617275 -2.133375281 7.587033497 -15.29437708 40.69993331 -98.29592331 249.7991406 -623.9663625

0 0.0521262 1.163436299 -1.064864888 4.601040065 -9.080009972 25.38250066 -61.03758593 156.7276962 -390.764009 982.8207234

0 0.053497942 0.053701163 1.279875298 -2.292981936 8.489256729 -20.05032533 54.17983316 -133.9007416 337.4250601 -831.3427046

0 0 0.05647005 0.094257876 1.301830529 -1.927588543 7.187612337 -14.83207386 36.77277081 -77.47045291 161.69261

0 0 0.101051669 0.050226648 1.934211993 -3.968950328 14.50296283 -37.71037163 107.5234862 -289.9710957 793.3065277

0 0.106995885 0.099705329 1.997136402 -3.558848253 13.20426056 -32.22660186 89.51090002 -231.4677294 611.9470788 -1600.263657

0 0.108367627 1.996562177 -1.770789963 7.844691609 -15.47349723 43.48895371 -105.9334807 276.5224458 -706.4821786 1830.86492

0 1.997256516 0.065073075 4.038058996 -4.009674825 14.38644347 -29.07754374 77.69777099 -189.437652 487.9971853 -1243.78043

];

Untitled

A=[0 0.527777778 0.533569578 0.670848193 0.671894953 0.741743113 0.742505961 0.787077087 0.792993299 0.824360471 0.834059659

0 0.0260631 0.316922175 0.322727826 0.470668944 0.474110483 0.568516339 0.582210048 0.648789319 0.670148832 0.719618943

0 0.026748971 0.040174262 0.200184077 0.210169875 0.332112543 0.362726517 0.454261132 0.49462144 0.564882666 0.608094965

0 0 0.014117513 0.032958503 0.126105146 0.190276355 0.279402301 0.351959518 0.427391041 0.496863936 0.559371768

0 0 0.025262917 0.044172707 0.171339287 0.2254654 0.347230449 0.406561319 0.510543877 0.566202436 0.653431257

0 0.053497942 0.078424275 0.327155006 0.340994554 0.528544251 0.55775814 0.698414274 0.734692137 0.843220941 0.881063978

0 0.054183813 0.553324358 0.567999931 0.823398458 0.829672534 0.992205994 1.006396017 1.12095436 1.141723286 1.227259674

0 0.998628258 1.014896527 1.278130971 1.282626605 1.41736477 1.419012654 1.504773858 1.511204256 1.571586216 1.581802138

];

Untitled

A = [0 0.521341463 0.533350684 0.665883284 0.67172425 0.737582734 0.742300916 0.783402078 0.792263221 0.821044626 0.832741575

0 0.025745257 0.309867432 0.322389241 0.463576058 0.473690852 0.561803148 0.580799585 0.642500899 0.667576709 0.713758941

0 0.026422764 0.039844834 0.19439517 0.209551035 0.324838828 0.360289954 0.446628071 0.490481735 0.557319327 0.602655924

0 0 0.013775283 0.032271323 0.121627209 0.186322238 0.272934655 0.345716426 0.420041954 0.48920823 0.551702501

0 0 0.024650506 0.043478108 0.165231812 0.222059656 0.338466704 0.401309548 0.500574659 0.559848467 0.642988268

0 0.052845528 0.077812066 0.318154496 0.340113167 0.517359322 0.555279122 0.686738426 0.730739367 0.831670226 0.876005585

0 0.053523035 0.541216373 0.567248555 0.811149572 0.828928219 0.980647477 1.004740171 1.110171466 1.138970004 1.217228937

0 0.986449864 1.014353642 1.2686084 1.282180056 1.409335769 1.418604515 1.497702055 1.510262576 1.565228281 1.580242153

];

Untitled

A = [0 0.521341463 0.660134004 0.733388289 0.778623536 0.809629255 0.833499546 0.853146128 0.874456636 0.89662983 0.91821302

0 0.297542179 0.454584044 0.551558953 0.618028729 0.669201641 0.711319803 0.757005075 0.804539766 0.850809611 0.893830377

0 0.181543839 0.307565326 0.399505581 0.474556151 0.538170121 0.614151846 0.692238655 0.766649719 0.834754854 0.895645599

0 0.094646331 0.183363288 0.269966263 0.349148056 0.464963741 0.581305519 0.68770027 0.781990078 0.864445067 0.936059393

0 0.049343057 0.132244966 0.218114121 0.395300761 0.559792448 0.700607048 0.81957937 0.920228118 1.005699893 1.078532032

0 0.07857011 0.165864058 0.457217856 0.680412569 0.852260576 0.98840057 1.098787708 1.18986574 1.265963073 1.330108814

0 0.094484891 0.623115089 0.903565748 1.088128282 1.22260293 1.326280803 1.409043276 1.476652765 1.53276719 1.579855307

0 1.035708756 1.311305597 1.457516154 1.553736255 1.623843465 1.677895039 1.721042548 1.756290178 1.785544954 1.810093942

];

Untitled

HW2\_Q4\_main

8.59\*10^-16

HW2\_Q4\_main

HW2\_Q5\_main

HW2\_Q2

HW2\_Q4

%-- 28/11/2020 15:23 --%

HW3\_Q1

cos(0)\*(88\_400sin(0)+40\*sin(0)+405\*cos(0))

cos(0)\*(88+400sin(0)+40\*sin(0)+405\*cos(0))

cos(0)\*(88+400\*sin(0)+40\*sin(0)+405\*cos(0))

ForwardKinematics([0,0,0])

P(p,:)

HW3\_Q1

20^3

HW3\_Q1

untitled

HW3\_Q1

%-- 28/11/2020 17:59 --%

HW3\_Q1

unique(final\_solution\_P1,'rows')

unique(round(final\_solution\_P1,4),'rows')

9.1226-2\*PI

9.1226-2\*pi

HW3\_Q1

f = unique(final\_solution\_P1,'rows');

f = unique(round(final\_solution\_P1,4),'rows');

HW3\_Q1

mod(-1,2\*pi)

mod([-1,-1,-1],2\*pi)

mod([-1,-1,-pi],2\*pi)

mod([-1,-1,-0.5\*pi],2\*pi)

HW3\_Q1

Call

HW3\_Q1

warning('no solution found for P% with n=%d',i,n\_max)

warning('no solution found for P% with n=%d',[i,n\_max])

warning('no solution found for P% with n',i)

warning(strcat('no solution found for P',string(i),' with n=',string(n\_max)));

HW3\_Q1

HW3\_Q2

%-- 01/12/2020 14:43 --%

A = [ 1 1 1;1 1 1;1 1 1];

eig(A)

[1 i]\*[1;-i]

[1 i]'\*[1;-i]

[-1 -i]\*[1;-i]

[-1 i]\*[1;-i]

[1 -i]\*[1;-i]

[1,i]\*[1;i]

syms a b

f = a\*i-(3-a)\*i-4=0;

f = a\*i-(3-a)\*i-4

solve(f,a)

4/(2\*i)

3-(3/2 - 2i)

clc

a=1.5-2i;

b=1.5+2i;

syms t

a\*exp(ti)+b\*exp(-ti)\*i

a\*exp(t\*i)+b\*exp(-t\*i)\*i

a\*exp(t\*i)+b\*exp(-t\*i)

expand(ans)

expand(a\*exp(t\*i)+b\*exp(-t\*i))

expand(a\*exp(t\*i)\*i+b\*exp(-t\*i)\*i)

(20+sqrt(4\*64))/2

(20-sqrt(4\*64))/2

(20+sqrt(400-4\*64))/2

Q=[1 -1;1 1];

L=[16 0;0 4];

Q\*L\*Q'

Q'\*L\*Q

Q\*L\*inv(Q)

Q=[1 1;-1 1];

Q\*L\*inv(Q)

Q\*L\*Q'

V = [4 0;0 2];

Q\*V\*inv(Q)

inv(Q)

Q\*L\*inv(Q)

sqrt(Q\*V\*inv(Q))

Q=[1 -1;1 1];

L = [4 0;0 16];

Q\*L\*Q'

[1 -1]\*[1;1]

[sqrt(2) -sqrt(2)]\*[sqrt(2);sqrt(2)]

Q = [sqrt(2) -sqrt(2);sqrt(2) sqrt(2)]

Q\*L\*Q'

Q=[ 1.4142135623731 1.4142135623731

-1.4142135623731 1.4142135623731]

Q\*L\*Q'

Q = [1/sqrt(2) 1/sqrt(2);-1/sqrt(2) 1/sqrt(2)];

Q\*L\*Q'

Q\*L\*inv(Q)

Q'

inv(Q)

Q

Q\*V\*inv(Q)

ans^2

A=[-1;-1];

A'\*5\*A

%-- 12/12/2020 16:11 --%

HW4\_Q1

HW4\_Q2a

numel()

numel(0)

numel(2)

numel(0)

%-- 12/12/2020 21:14 --%

order\_reduction

%-- 19/12/2020 18:43 --%

HW4\_Q2aa

x1(1)=-l1\*sin(y1(1,1));

x2(1)=-l1\*sin(y1(1,1))-l2\*sin(y1(1,2))

sin(pi/4)

-cos(pi/4)

sin(1)+sin(pi/4)

-cos(pi/4)-cos(1)

-l1\*sin(y1(1,1));

-l1\*sin(y1(1,1))

-l1\*cos(y1(1,1))

l1\*sin(y1(1,1))+l2\*sin(y1(1,2))

-l1\*cos(y1(1,1))-l2\*cos(y1(1,2))

HW4\_Q2aa

order\_reduction

rkrun

HW4\_Q2aa

main\_q3

HW4\_Q3

HW4\_Q4

acos(0.5)

ans\*180/(2\*pi)

HW4\_Q4

y(end,1)\*180/(2\*pi)

HW4\_Q4

y(end,1)\*180/(2\*pi)

HW4\_Q4

y(end,1)\*180/(2\*pi)

HW4\_Q4

y(end,1)\*180/(2\*pi)

HW4\_Q4

y(end,1)\*180/(2\*pi)

HW4\_Q4

y(end,1)\*180/(2\*pi)

HW4\_Q4

x(19,1)

x(19,1)+0.5-dx

min(x(:,1))

ans-dx

HW4\_Q4

degrees(pi)

deg(pi)

rad2deg(pi)

rad2deg(acos(0.5))

HW4\_Q4

x(end,1)-dx

x(end,2)-dx

HW4\_Q3

x(end,2)+0.5-dx

x(end,2)+0.5+dx

x(end,2)

x(end,2)-dx

x(end-1,2)-dx

HW4\_Q3

-0.5+dx-x(end,2)

HW4\_Q4

-0.5+dx-x(end,2)

HW4\_Q4

ttemp\_x1+0.5-threshold

temp\_x1+0.5-threshold

dist1>0

HW4\_Q4

pi/4

HW4\_Q4

HW4\_Q3

(90580202/688068385)\*pi

rad2deg(y(end,1))

HW4\_Q4

rad2deg(y(end,1))

rad2deg(y(end,2))

x1

x2

t(45)

%-- 20/12/2020 20:25 --%

HW4\_Q4

%-- 21/12/2020 19:34 --%

HW4\_Q5

part\_7

HW4\_Q5

90580202\*pi/688068385

HW4\_Q5

HW4\_Q4

HW4\_Q5

HW4\_Q4

main\_fzero

order\_reduction

main\_fzero

t

t\_optimal

HW4\_Q4

deg2rad(-30)

main\_fzero

t\_optimal

solver

main\_fzero

t\_optimal

HW4\_Q4

main\_fzero

TutODE\_BVP\_NonLinShooting\_secant

main\_fzero

solver

main\_fzero

x1(254)

x1(253)

x2(253)

main\_fzero

HW4\_Q3

main\_fzero

HW4\_Q4

deg2rad(-30)

main\_fzero

90580202\*pi/688068385

main\_fzero

HW4\_Q4

main\_fzero

fzero

main\_fzero

HW4\_Q4

fzero\_cases

main\_fzero

y\_b(1)

rad2deg(y(end,1))

(y(end,1))

main\_fzero

fzero\_cases

main\_fzero

solver

main\_fzero

90580202\*pi/688068385

main\_fzero

x2(end,2)

x2(end,1)

main\_fzero

HW4\_Q5

HW4\_Q3

HW4\_Q5

%-- 24/12/2020 11:22 --%

pdetoolbax

pdetool

femlab

pdepe

2+cos(1)

-sin(1)

2.54030230586814^2

2.54030230586814\*2

syms

sym x

f=6.45\*x^2-5.08\*x+1+cos(1+0.84\*x)-2.54\*x\*cos(1+0.84\*x)

syms x

f=6.45\*x^2-5.08\*x+1+cos(1+0.84\*x)-2.54\*x\*cos(1+0.84\*x)

diff(f)

fsolve(diff(f)==0,1)

6.45\*2

6.45313580519899\*2

g=cos(1+0.84\*x)

diff(g)

g=2.54\*x\*cos(1+0.84\*x)

diff(g)

127/50

2667/1250

S = solve((129\*x)/10 - (127\*cos((21\*x)/25 + 1))/50 - (21\*sin((21\*x)/25 + 1))/25 + (2667\*x\*sin((21\*x)/25 + 1))/1250 - 127/25==0)

0.84\*0.43+1

1-2.54\*0.43

TutPDE\_Elliptic\_Leibmann

TutPDE\_Elliptic\_SecondaryVariables

pdetool

A=[-1 1 0;-1 0 1; 0 -1 1; 0 -1 0;-1 0 0;0 0 -1];

A'\*A

-A\*[-1 -1 -1]'

thermalmodel = createpde('thermal','transient');

geometryFromEdges(thermalmodel,@crackg);

pdegplot(thermalmodel,'EdgeLabels','on')

ylim([-1,1])

axis equal

thermalProperties(thermalmodel,'ThermalConductivity',1,...

'MassDensity',1,...

'SpecificHeat',1);

thermalBC(thermalmodel,'Edge',6,'Temperature',20);

thermalBC(thermalmodel,'Edge',1,'HeatFlux',-10);

thermalIC(thermalmodel,0);

generateMesh(thermalmodel);

figure

pdemesh(thermalmodel)

title('Mesh with Quadratic Triangular Elements')

tlist = 0:0.5:5;

thermalresults = solve(thermalmodel,tlist)

[qx,qy] = evaluateHeatFlux(thermalresults);

pdeplot(thermalmodel,'XYData',thermalresults.Temperature(:,end), ...

'Contour','on',...

'FlowData',[qx(:,end),qy(:,end)], ...

'ColorMap','hot')

pde\_poisson()

%nm09e01.m

f=@(x,y)0; g=@(x,y)0; % Eq.(E9.1.1) w.r.t. Eq.(9.1.1)

% (Rectangular) Domain and BC types

x0=0; xf=pi; y0=0; yf=pi; % 0/1 for Dirichlet/Neumann conditions

D=[x0 xf y0 yf; 0 0 0 0];

% BCs

bx0=@(y)0; bxf=@(y)0; % Eq.(E9.1.2a,b)

by0=@(x)sin(2\*x); byf=@(x)0; % Eq.(E9.1.2c,d)

% Numbers of grid points along x-/y-axes, Error tolerance,

Mx=50; My=50; tol=1e-5; MaxIter=600;

[un,xx,yy]=pde\_poisson(f,g,bx0,bxf,by0,byf,D,Mx,My,tol,MaxIter);

mesh(xx,yy,un) % Plot the solution graph

pdetool

mat

pdetool

cos(0)

createfigure

movie2avi(M,'myavifile.avi','Compression','Cinepak')

movie(M)

close all

ex2

%-- 06/01/2021 11:21 --%

explicit\_full

explicit

1-2

explicit

%-- 07/01/2021 21:31 --%

PDE\_Parabolic\_ADI

BC\_d(2,2)

PDE\_Parabolic\_ADI

ADIconv

q3

pdetool

PDE\_Parabolic\_ADI

HW5\_Q3

%-- 08/01/2021 11:50 --%

PDE\_Parabolic\_ADI

PDE\_Parabolic\_Explicit

HW5\_Q3

%-- 08/01/2021 13:19 --%

HW5\_Q3

PDE\_Parabolic\_ADI

PDE\_Parabolic\_Explicit

explicit\_full

PDE\_Parabolic\_Explicit

dr/dx^2

dt/dx^2

PDE\_Parabolic\_Explicit

dt/dx^2

PDE\_Parabolic\_Explicit

dt/dx^2

PDE\_Parabolic\_Explicit

dt/dx^2

PDE\_Parabolic\_Explicit

dt/dx^2

PDE\_Parabolic\_Explicit

dt/dx^2

PDE\_Parabolic\_Explicit

min(U)

min(min(U))

round(ans,1)

round(-0.34,1)

floor(-0.34,1)

floor(-0.34)

floor(-0.34,1)

ceil(0.34,1)

ceil(0.34)

floor(1.2,5)

PDE\_Parabolic\_Explicit

HW5\_Q2

HW5\_Q3

HW5\_Q2

t(end)

HW5\_Q2

0.0005/(0.05^2)

HW5\_Q2

M = (U-U2);

HW5\_Q2

N = (U2-U);

max(max(N))

HW5\_Q2

norm(rand(10))

HW5\_Q2

max(max(U\_CN))

max(max(U\_cn))

PDE\_Parabolic\_ADI

pdetool

HW5\_Q3

PDE\_Parabolic\_ADI

HW5\_Q3

cv = linspace(1,m,5);

cv

HW5\_Q3

Copy\_of\_HW5\_Q3

HW5\_Q3

Copy\_of\_HW5\_Q3

HW5\_Q3

Copy\_of\_HW5\_Q3

HW5\_Q3

Copy\_of\_HW5\_Q3

HW5\_Q3

Copy\_of\_HW5\_Q3

la\*U12(j-1,1)+2\*(1-la)\*U12(j,1)+la\*U12(j+1,1)+dt\*f(x(1),y(j)).\*U(l,2:n,1);

la\*U12(j-1,1)+2\*(1-la)\*U12(j,1)+la\*U12(j+1,1)+dt\*f(x(1),y(j)).\*U(l,2:n,1)

B(1)=la\*U12(j-1,1)+2\*(1-la)\*U12(j,1)+la\*U12(j+1,1)+dt\*f(x(1),y(j)).\*U(l,j,1);

B(1)=la\*U12(j-1,1)+2\*(1-la)\*U12(j,1)+la\*U12(j+1,1)+dt\*f(x(1),y(j)).\*U(l,j,1)

la\*U12(j-1,1)

2\*(1-la)\*U12(j,1)

la\*U12(j+1,1)

dt\*f(x(1),y(j)).\*U(l,j,1)

Copy\_of\_HW5\_Q3

la\*U12(j-1,1)+2\*(1-la)\*U12(j,1)+la\*U12(j+1,1)+dt\*f(x(1),y(j)).\*U(l,j,1)

B(1)=ans

B(1)

Copy\_of\_HW5\_Q3

HW5\_Q3

Copy\_2\_of\_HW5\_Q3

max(U(:,:,1))

max(max(U(:,:,1)))

max(max(U(:,:,10)))

Copy\_2\_of\_HW5\_Q3

HW5\_Q3

%-- 11/01/2021 17:22 --%

linspace(-5,5,10)

x=linspace(-5,5,10);

y = (1+x.^2).^-1

syms x

0.038+0.0212\*(x+5)+0.0118\*(x+5)\*(x+3.89)+0.0092\*(x+5)\*(x+3.89)\*(x+2.78)+0.0062\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)-0.0087\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)+0.0036\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)-0.0008\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)+0.0001\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)\*(x-2.78)

double(ans)

expamd(0.038+0.0212\*(x+5)+0.0118\*(x+5)\*(x+3.89)+0.0092\*(x+5)\*(x+3.89)\*(x+2.78)+0.0062\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)-0.0087\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)+0.0036\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)-0.0008\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)+0.0001\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)\*(x-2.78)

expand(0.038+0.0212\*(x+5)+0.0118\*(x+5)\*(x+3.89)+0.0092\*(x+5)\*(x+3.89)\*(x+2.78)+0.0062\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)-0.0087\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)+0.0036\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)-0.0008\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)+0.0001\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)\*(x-2.78)))

expand(0.038+0.0212\*(x+5)+0.0118\*(x+5)\*(x+3.89)+0.0092\*(x+5)\*(x+3.89)\*(x+2.78)+0.0062\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)-0.0087\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)+0.0036\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)-0.0008\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)+0.0001\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)\*(x-2.78))

vpa(ans,4)

f=ans;

fplot(f)

fplot((@x),(1+x^2)^-1)

fplot((@x),((1+x^2)^-1))

fplot((@x)((1+x^2)^-1))

fplot((@x)(1+x^2)^-1)

fun = ((@x),(1+x^2)^-1)

fun = ((@x)(1+x^2)^-1)

fun = ((1+x^2)^-1))

fun = (1+x^2)^-1)

fun = (1+x^2)^-1

fplot(fun)

f2 = 0.0001\*x^8 + 8.9e-5\*x^7 - 0.004873\*x^6 - 0.003153\*x^5 + 0.08045\*x^4 + 0.05048\*x^3 - 0.4738\*x^2 - 0.2471\*x + 0.8512;

fplot(f2)

fplot(fun)

hold on

fplot(f2)

legend('Original Function','Newton polynomial interpolation')

u = linspace(-5,5,10);

v = (1+u.^2).^-1;

for i=1:9

b1(i)=(v(i+1)-v(i))/(u(i+1)-u(i));

end

for i=1:8

b2(i)=(b1(i+1)-b1(i))/(u(i+2)-u(i));

end

for i=1:7

b3(i)=(b2(i+1)-b2(i))/(u(i+3)-u(i));

end

for i=1:6

b4(i)=(b3(i+1)-b3(i))/(u(i+4)-u(i));

end

for i=1:5

b5(i)=(b4(i+1)-b4(i))/(u(i+5)-u(i));

end

for i=1:4

b6(i)=(b5(i+1)-b5(i))/(u(i+6)-u(i));

end

for i=1:3

b7(i)=(b6(i+1)-b6(i))/(u(i+7)-u(i));

end

for i=1:2

b8(i)=(b7(i+1)-b7(i))/(u(i+8)-u(i));

end

for i=1:1

b9(i)=(b8(i+1)-b8(i))/(u(i+9)-u(i));

end

b9

finter = 0.038+0.0212\*(x+5)+0.0118\*(x+5)\*(x+3.89)+0.0092\*(x+5)\*(x+3.89)\*(x+2.78)+0.0062\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)-0.0087\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)+0.0036\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)-0.0008\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)+0.0001\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)\*(x-2.78)

finter = v(1)+b1(1)\*(x+5)+b2(1)\*(x+5)\*(x+3.89)+b3(1)\*(x+5)\*(x+3.89)\*(x+2.78)+b4(1)\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)+b5(1)\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)+b6(1)\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)+b7(1)\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)+b8(1)\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)\*(x-2.78)+b9(1)\*(x+5)\*(x+3.89)\*(x+2.78)\*(x+1.67)\*(x+0.55)\*(x-0.55)\*(x-1.67)\*(x-2.78)\*(x-3.89);

finter = vpa(finter,4)

expand(finter)

vpa(ans,4)

finter=ans;

fplot(finter)

fplot(fun)

hold on

fplot(finter)

legend('Original Function','Newton polynomial interpolation')

plot(u,v)

plot(u,v,'o')

fplot(fun)

hold on

fplot(finter)

plot(u,v,'o')

plot(u,v,'ok')

legend('Original Function','Newton polynomial interpolation')

axis equal

axis not equal

fplot(fun)

hold on

fplot(finter)

plot(u,v,'ok')

legend('Original Function','Newton polynomial interpolation')

grid on

legend('polynomial points')

legend('Original Function','Newton polynomial interpolation','polynomial points')

%-- 18/01/2021 13:44 --%

x = 0:0.1:1;

y = exp(x);

x = x';

y = y';

s = sum(y(2:10));

I1 = (0.1/2)\*(1+2\*s+y(11));

1.7183-1.7197

-ans/1.7183

ans\*100

s1 = sum(y(1:2:9));

s2 = sum(y(2:2:8));

I2 = (1/30)\*(1+4\*s1+2\*s2+y(11));

s1 = sum(y(2:2:10));

s2 = sum(y(3:2:9));

I2 = (1/30)\*(1+4\*s1+2\*s2+y(11));

steepestdescend

OptStoc\_Cont\_GA

norm(minc(iga+1)-minc(iga))

iga+1

OptStoc\_Cont\_GA

norm(minc(iga+2)-minc(iga+1))

OptStoc\_Cont\_GA

norm(minc(iga+2)-minc(iga+1))<=conv\_crit

OptStoc\_Cont\_GA

norm(minc(iga+2)-minc(iga+1))<=conv\_crit

norm(minc(338)-minc(337))<=conv\_crit

norm(minc(4781)-minc(4780))

norm(minc(iga+2)-minc(iga+1))

norm(minc(iga+1)-minc(iga))

OptStoc\_Cont\_GA

iga~=0

OptStoc\_Cont\_GA

OptStoc\_Cont\_SA

OptStoc\_Cont\_GA

norm(meanc(iga+1)-meanc(iga))

OptStoc\_Cont\_GA

minc(iga-25:iga+1)

minc(iga-50:iga-26)

norm(minc(iga-24:iga+1)-minc(iga-50:iga-25))

OptStoc\_Cont\_GA

iga-24

iga+1

OptStoc\_Cont\_GA

iga-itr\_for\_convergence\_check/2-1

iga+1

OptStoc\_Cont\_GA

N = minc(iga-half-1:iga+1);

M = minc(iga-itr\_for\_convergence\_check:iga-half);

N = minc(iga-half+1:iga+1);

iga-half

iga+1

52-26

N=minc(iga-half-1:iga+1);

OptStoc\_Cont\_GA

52-25

51-25-1

N = minc(iga-half+1:iga+1);

iga-half+2

iga+1

52-28

iga-half+1:iga+1

iga-half+2:iga+1

iga-itr\_for\_convergence\_check:iga-half+2

iga-itr\_for\_convergence\_check:iga-half+1

OptStoc\_Cont\_GA

norm(minc(iga-half+2:iga+1)-minc(iga-itr\_for\_convergence\_check+2:iga-half+1))

iga-half+2:iga+1

iga-itr\_for\_convergence\_check+2:iga-half+1

OptStoc\_Cont\_GA

iga-itr\_for\_convergence\_check+2:iga-half+1

OptStoc\_Cont\_GA

iga-itr\_for\_convergence\_check+2:iga-half+1

OptStoc\_Cont\_GA

GA\_Rosenbrock

GA\_Easom

GA\_Rosenbrock

steepestdescend

GA\_Rosenbrock

Z2-Z

GA\_Rosenbrock

OptStoc\_Cont\_SA

GA\_Rosenbrock

cost

prev\_cost

norm(cost-prev\_cost)

GA\_Rosenbrock

K

GA\_Rosenbrock

min(K)

GA\_Rosenbrock

cost(1)-cost(2)

GA\_Rosenbrock

GA\_Easom

GA\_Eggholder

GA\_Easom

GA\_Eggholder

OptStoc\_Cont\_SA

SA\_Rosenbrock

E-E\_prev

E-Eprev

SA\_Rosenbrock

OptDet\_SimplexNelderMead

GA\_Rosenbrock

SA\_Rosenbrock

Enew=Rosenbrock(XNew(1),XNew(2));

SA\_Rosenbrock

E\_accu(Niter-half+1:Niter)

E\_accu(Niter-itr\_for\_convergence\_check+1:Niter-half)

Niter-half+1

Niter

Niter-itr\_for\_convergence\_check+1

Niter-half

Niter-itr\_for\_convergence\_check:Niter-half-1

SA\_Rosenbrock

1-1e-5

1-1e-2

SA\_Rosenbrock

SA\_Eggholder

SA\_Easom

SA\_Eggholder

%-- 20/01/2021 23:42 --%

GA

OptDet\_SimplexNelderMead

f(n\_dim+1)-f(1)

OptDet\_SimplexNelderMead

Rosenbrock\_f(ans)

Rosenbrock\_f([2 4])

Rosenbrock(2,4)

Rosenbrock(1.9,3.9)

Rosenbrock(1.99,4)

OptDet\_SimplexNelderMead

OptDet\_QuasiNewton

norm(dx)

SimplexNelderMead

SimplexNelderMead\_Rosenbrock

OptDet\_SteepestDescend

SimplexNelderMead\_Easom

SimplexNelderMead\_Eggholder

min\_val

SimplexNelderMead\_Eggholder

GA\_Easom

SA\_Easom

SimplexNelderMead\_Easom

GA\_Eggholder

SA\_Eggholder

SimplexNelderMead\_Eggholder

GA\_Rosenbrock

SA\_Rosenbrock

SimplexNelderMead\_Rosenbrock

%-- 24/01/2021 22:54 --%

q1

S.a

q1

S2.a

S2.b

S2.x

q1

S.a

S.b

S.c

S.x

S.y

S.z

(5/9)\*(sqrt(15)/5)^2+(5/9)\*(-sqrt(15)/5)^2

lgwt(6,-1,1)

[x,y] = lgwt(6,-1,1);

y

x

[x,y] = lgwt(3,-1,1);

x

y

sqrt(15)/5

[x,y] = lgwt(3,0,2);

x

y

%-- 26/01/2021 17:03 --%

AdaptQuad

AdaptiveIntegral

Integr\_AdaptiveIntegral

AdaptQuad

HW7\_Q3

results(1)

results\_1(1)

results\_2(1)

results\_1(2)

HW7\_Q3

fun1([1,2])

HW7\_Q3

diff1 = results\_1 - Q1;

diff2 = results\_2 - Q2;

HW7\_Q3

HW7

HW7\_Q3

HW7

HW7\_Q3

HW7

HW7\_Q3

HW7

AdaptQuad

HW7

HW7\_Q3

HW7

HW7\_Q3

1e-6

10^-6-1e-6

HW7\_Q3

%-- 26/01/2021 23:54 --%

HW7\_Q3

%-- 22/02/2021 07:41 --%

1/240-1/720-1/144

1/240

log(exp(2))

log(1)

fun = @(x) 1-abs(x)/pi;

k = -2:0.1:2;

fun = @(x) 1-abs(x)./pi;

y = fun(k);

plot(k,y)

axis

grid

ylim([0.3 1.1])

integral((1-x/pi)\*cos(k\*x))

syms x k

integral((1-x/pi)\*cos(k\*x))

integral((1-x/pi)\*cos(k\*x),0,pi)

%-- 12/03/2021 21:20 --%

R = [-0.443289 -0.573841 0.688623; -0.310608 0.818967 0.482511; 0.840843 0.00000 0.541278];

V = [-834.535;398.818;348.406];

V\*R

V'\*R

%-- 28/05/2021 18:29 --%

0.28/0.57

0.23/0.43

(112-101)/7

1-0.9418

(95-101)/7

4/7

0.7157-(1-0.8051)

-10/7

1-0.9236

101-2.1

(98.9-101)/7

(103.1-101)/7

0.6179-(1-0.6179)

1.96/2

2.1/2.055

2.1/1.02

0.9803-(1-0.9803)

176.2-173.5

2.7/2

176.2-1.35

(173.5-175.7)/((7.8^2)/50)

(176.2-175.7)/((7.8^2)/50)

0.6591-(1-0.9649)

(173.5-175.7)/(7.8/sqrt(50))

(176.2-175.7)/(7.8/sqrt(50))

176.2-173.5

0.6736-(1-0.9767)

200\*ans

factorial(3)

factorial(200)

130\*(1-0.6503)

(100.5-130)/sqrt(45.461)

(7.8^2)/(50\*200)+(7.5^2)/30

(7.5-8.7)/sqrt(1.88)

1-0.8078

400\*1.5

0.4^2

ans\*400

100/sqrt(64)

(4.5-1.5)/(2.5/sqrt(30))

0.4^2+2.5^2

1100\*7.5

1100\*6.41

(6430-8250)/sqrt(7051)

(80.5-86)/(15/sqrt(14))

1-0.9147

(82-86)/(15/sqrt(14))

1-(1-0.8413)

450/2100

(0.21-0.17)/sqrt((0.17\*0.83)/2100)

(0.18-0.21)/sqrt((0.17\*0.83)/2100)

(0.18-0.21)/sqrt((0.21\*0.79)/2100)

(0.062\*62+0.423\*58)/(62+58)

=sqrt(0.24\*0.76\*(1/62+1/58))

sqrt(0.24\*0.76\*(1/62+1/58))

1-0.236

sqrt(0.236\*0.764\*(1/62+1/58))

(0.062-0.423)/0.078

sqrt((4.190^2)/8+(12.444^2)/6)

4.875-15.833

ans/5.292

5.292\*5.292

(4.875-15.833)/5.292

sqrt(272.833/5)

sqrt((4.190^2)/8+(7.387^2)/6)

(4.875-15.833)/3.360

(4.875-15.833)

3.36^2

(4.875-15.833)-1.881\*3.60

(4.875-15.833)+1.881\*3.60

0.78\*10/7

67-1.14\*63

56+4.82

ans/1.14

77-1.14\*63

%-- 11/07/2021 13:37 --%

main

%-- 17/07/2021 21:02 --%

testgmm

untitled

train

load('obj1.mat')

%-- 20/08/2021 20:18 --%

ANOVA

size(pima\_table,2)

ANOVA

K=pima\_table(:,n\_variables+1)=='1';

K=pima\_table{:,n\_variables+1}=='1';

ANOVA

K=pima\_table{:,n\_variables+1}=='1';

temp1 = pima\_table(pima\_table{:,n\_variables+1}=='1',:);

temp1 = pima\_table{pima\_table{:,n\_variables+1}=='1',:};

temp1 = pima\_table{pima\_table{:,n\_variables+1}='1',:};

temp1 = pima\_table{pima\_table{:,n\_variables+1}=='1',:};

temp1 = pima\_table{pima\_table(:,n\_variables+1)=='1',:};

temp1 = pima\_table{pima\_table{:,9}=='1',:};

temp1 = pima\_table(pima\_table{:,9}=='1',:);

temp1 = pima\_table(pima\_table{:,n\_variables+1}==1,:);

ANOVA

K = anova1(temp1{:,1},temp2{:,1},'off');

temp1{:,1}

ANOVA

K=pima\_table{:,1:8};

ANOVA

tbl{1}.F

tbl{1}

tbl{1,2,5}

(tbl{1}){2,5}

tbl{1}

ans{2,5}

tbl{1}

a = table(ans)

tbl(1){2,6}

tbl{1}{2,6}

ANOVA

table(pima\_table.Properties.VariableNames')

table(pima\_table.Properties.VariableNames(1:end-1)')

ANOVA

close all

delete(findall(0))

ANOVA

Regression

Untitled2

B(:,1)./B(:,2)

log(B(:,1)./B(:,2))

ln(B(:,1)./B(:,2))

log10(B(:,1)./B(:,2))

Regression

768-516

Regression

repmat(B',252,1)

data = [ones(252,1);pima\_table{index+1:end,1:n\_variables}];

N = pima\_table{index+1:end,1:n\_variables};

Regression

N = pima\_table{index+1:end,1:n\_variables};

data = [ones(253,1),pima\_table{index+1:end,1:n\_variables}];

temp1 = repmat(B',253,1).\*data;

Regression

p = 1/(1+exp(-temp2));

p = 1/(1+exp(-temp2'));

p = (1/(1+exp(-temp2)))';

Regression

p = (1./(1+exp(-temp2)))';

p = (1./(1+exp(-temp2)));

p-p2

max(ans)

q = (p>=0.5,1,0);

q = (p>=0.5);

Z = Y(253:end);

q = (p>=0.5);

q = (p>=0.5);

Z = pima\_table{index+1:end,end};

M = (double(q)==Z);

sum(M)

47/253

ANOVA

Regression

sum(Z)/253

Regression

output=[p;Z];

output=[p,Z];

min(p)

mean(p)

median(p)

Regression

max(p)

min(p)

Regression

ANOVA

Regression

anova

GMM

anova

GMM

anova

GMM

Regression

ANOVA

anova

exp(7)/(1+exp(7))

1/(1+exp(-7))

exp(7)/(1+exp(7))

exp(2)/(1+exp(2))

1/(1+exp(-2))

anova

1-scoreLR

anova

Regression

stats.p

ANOVA

size(pima\_table{:,i})

size(pima\_table{:,9})

[p(i),tbl{i}] = anova1(pima\_table{:,i},pima\_table{:,end});

[p(i),tbl{i}] = anova1(pima\_table{:,i},pima\_table{:,end},'alpha',0.05,'display','off');

Regression

ANOVA

Regression

anova

%-- 27/12/2021 12:51 --%

Adjustment

%-- 27/12/2021 18:00 --%

hw1b

%-- 10/01/2022 17:36 --%

hw1\_final

-log(f\_k)-0.5\*log(det(Sk))-0.5\*VTPV

tempV'\*tempP\tempV

tempV'\*(tempP\tempV)

tempV'\*(tempP/tempV)

tempV'\*tempP\*tempV

tempV'/tempP\*tempV

tempV'/inv(Sk)\*tempV

tempV'\inv(Sk)\*tempV

tempV'\*inv(Sk)\*tempV

tempV'\(Sk)\*tempV

tempV'/(Sk)\*tempV

hw1\_final

unique(maxValues\_ML)

unique(clusterResults\_ML)

max(max(clusterResults\_ML))

max(max(maxValues\_ML))

max(tempStorage)

min(tempStorage)

hw1\_final

sum(sum(isnan(maxValues\_ML)))

hw1\_final

tempV'/Sk\*tempV

Sk = cov([sampled\_R{i},sampled\_G{i},sampled\_B{i}]);

tempV'/Sk\*tempV

VTPV = tempV'./Sk.\*tempV;

hw1\_final

tempV'/Sk\*tempV

hw1\_final

tempV(j,:)'

hw1\_final

%-- 11/01/2022 17:51 --%

script

96/8

157+48

script

hw1\_final

%-- 13/01/2022 20:45 --%

[1 -1]'\*[0 100]

[1 -1]\*[0 100]

[1 -1]\*[0 100]'

-100/sqrt(2)

untitled

-50/SQRT(2)

-50/sqrt(2)

-50\*sqrt(2)

(sqrt(2)/2)^-2

(sqrt(2)/2)^-3\*-50

reshape(s,8,2)

reshape(s,2,8)

reshape(s',8,2)

reshape(s,2,8)

16^1/3

16^(1/3)

16/2^3

16/2^5

untitled

temp = reshape(prevW,2,length(prevW)/2);

untitled

A=[1 2];

min(0,A)

min(1,2)

untitled

min(Ds{3})

untitled

min(Ds{i})

max(Ds{i})

min(Ds{i})

untitled

%-- 14/01/2022 21:56 --%

untitled

%-- 15/01/2022 16:19 --%

hw1\_final

%-- 17/01/2022 17:39 --%

appFile = fullfile(matlabroot,'extern','examples','compiler','magicsquare.m');

compiler.build.standaloneApplication(appFile);

%-- 17/01/2022 19:13 --%

appFile = fullfile(matlabroot,'extern','examples','compiler','magicsquare.m');

compiler.build.standaloneApplication(appFile);

%-- 17/01/2022 19:16 --%

!magicsquare 4

%-- 17/01/2022 19:18 --%

compiler.runtime.download

%-- 08/04/2022 19:26 --%

hw1

n1 = size(I1,3);

hw1

rangeI1 = table([1:n1]',zeros(n1,0),zeros(n1,0));

rangeI1 = table([1:n1]',zeros(n1,1),zeros(n1,1));

rangeI1 = table([1:n1]',zeros(n1,1),zeros(n1,1));

rangeI1.Properties.VariableNames = {'band\_number','min','max'};

rangeI1.min(2)

hw1

K = I1(:,:1);

K = I1(:,:,1);

min(min(K))

K = I1(:,:,15);

min(min(K))

max(max(K))

K=uint16(65535);

K=uint16(65536);

K=uint16(-2);

%-- 08/04/2022 22:43 --%

tut01

hw1

subplot(1,2,1), imshow(Im1);

subplot(1,2,2), imshow(Im2);

close all

K = Im4(:,:,1);

max(max(K))

min(min(K))

hw1

K = imadjust((I2(:,:,5)));

max(max(K))

min(min(K))

A = uint16(K);

max(max(A))

min(min(A))

hw1

close all

hw1

double(wavelengths1)

hdr1.wavelength

hdr1.wavelength(2:end-1)

wavelengths1 = strtrim(split(string(hdr1.wavelength(2:end-1)),','));

wavelengths1 = double(split(string(hdr1.wavelength(2:end-1)),','));

hw1

new\_wavelet

=1024\*4

1024\*2\*2

\*2

new\_wavelet

2048\*4

new\_wavelet

grid on

new\_wavelet

%-- 10/04/2022 03:44 --%

new\_wavelet

%-- 10/04/2022 13:21 --%

new\_wavelet

%-- 11/04/2022 05:37 --%

new\_wavelet

%-- 11/04/2022 10:19 --%

hw2

roi.Position

hw2

roi.Position

hw2

roi.Position

hw2

roi.Position

hw2

load('roi.mat')

load('roi.mat')

load('roi.mat')

%-- 11/04/2022 11:29 --%

load(roi)

load(roi.mat)

load('roi.mat')

hw2

roiStruct.roi = roi;

roiStruct.roi = roi1;

hw2

max(max(IM\_imadjust1(:,:,1))

hw2

load('poly1.mat', 'poly')

hw2

%-- 11/04/2022 13:58 --%

hw2

saveas(gcf,'Barchart.png')

hw2

load('polygons.mat')

774\*4

close all

%-- 13/04/2022 19:33 --%

load('stats1.mat')

hw2

min(wavelengths1-700)

min(abs(wavelengths1-700))

min(abs(wavelengths1-635))

min(abs(wavelengths1-520))

min(abs(wavelengths1-450))

min(abs(wavelengths1-average(635,700\_))

min(abs(wavelengths1-average(635,700))

average(1,2)

min(abs(wavelengths1-mean(635,700))

min(abs(wavelengths1-mean(635,700)))

mean(635,700)

mean(1,3)

min(3,1)

mean([635,700])

min(abs(wavelengths1-mean([635,700])))

hw2

h=imshow(IM\_imadjust1);

hw2

avgs1{:,2+i}

i

i=1

avgs1{:,2+i}

avgs2{:,2+i}

A = [avgs1{:,2+i},ones(8,1)]

l =avgs2{:,2+i}

[a b]=inv(A'\*A)\*(A'\*l)

inv(A'\*A)\*(A'\*l)

coeff{1}

coeffs{1}

a=1.59195908629819e-05;

b=-0.0693265047651795;

X=[ 7057.52364104214

3780.53414780693

2305.23988355167

65330.5361035981

65532.3720930233

65297.3519385027

49433.0758955677

18401.0601877416];

A =

7057.52364104214 1

3780.53414780693 1

2305.23988355167 1

65330.5361035981 1

65532.3720930233 1

65297.3519385027 1

49433.0758955677 1

18401.0601877416 1

A =

[

7057.52364104214 1

3780.53414780693 1

2305.23988355167 1

65330.5361035981 1

65532.3720930233 1

65297.3519385027 1

49433.0758955677 1

18401.0601877416 1]

A =[7057.52364104214 1

3780.53414780693 1

2305.23988355167 1

65330.5361035981 1

65532.3720930233 1

65297.3519385027 1

49433.0758955677 1

18401.0601877416 1]

X=[1.59195908629819e-05 -0.0693265047651795

];

A\*X-l

X=X'

A\*X-l

hw2

max(max(specImage2(:,:,1)))

for i = 1:15

m(i)=max(max(specImage2(:,:,1)));

end

m=m';

for i = 1:15

n(i)=max(max(imadjust(Reflectance\_L2(:,:,i))));

end

for i = 1:15

n(i)=max(max(imadjust(Reflectance\_L2(:,:,i))));

end

for i = 1:12

n(i)=max(max(imadjust(Reflectance\_L2(:,:,i))));

end

n=n';

for i = 1:12

n(i)=max(max((Reflectance\_L2(:,:,i))));

end

for i = 1:12

n(i)=sum(sum(imadjust(Reflectance\_L2(:,:,i))==1));

end

nRows2\*nColumns2

sum(sum(imadjust(Reflectance\_L2(:,:,5))-imadjust(Reflectance\_L2(:,:,6))))

A=imadjust(Reflectance\_L2(:,:,5));

B=imadjust(Reflectance\_L2(:,:,6));

sum(sum(A-B))

A=(Reflectance\_L2(:,:,5));

B=(Reflectance\_L2(:,:,6));

sum(sum(A-B))

hw2

for i = 9:12

roi = images.roi.Polygon;

draw(roi);

roiPositions{i} = roi.Position;

end

hw2

for i = 1:15

n(i)=sum(sum(DN\_image(:,:,i)>255));

end

n'

for i = 1:12

n(i)=sum(sum(Reflectance\_L2(:,:,i)>1));

end

n

hw2

untitled3

hw2

A=DN\_image(:,:,wl\_indices(j));

sum(sum(A-band\_from\_1))

mean(mean(A))

mean(mean(band\_from\_1))

max(pixels1)

min(pixels1)

hw\_2\_new

untitled3

load('roiPositions.mat')

roi1=[[311.593520570566,1560.68789153715;319.065587999065,1560.92510002694;325.826029958184,1567.32972925137;329.621365794882,1570.17623112889;340.177143590699,1570.88785659827;341.007373304977,1574.44598394518;341.956207264151,1584.05292778182;343.735270937603,1589.86453578176;341.007373304977,1596.86218623068;334.365535590755,1596.98079047557;328.672531835708,1593.77847586336;321.556277141899,1586.54361692465;315.151647917470,1582.15525986347;306.256329550209,1573.61575423090;304.833078611447,1568.39716745544;307.323767754281,1562.70416370039]];

roi7=[[95.6465045551682,990.457175776125;108.428694851351,1011.79603891521;130.084471799429,1000.38714179134;129.767557990433,997.218003701378;135.683282425030,994.999607038404;123.957471492168,977.146795798282]];

roi8=[[791.782209938343,1099.43389609457;800.926745245305,1112.05335481817;816.838236679417,1141.68164921273;797.268931122520,1147.35126110304;784.466581692774,1136.74360014697;781.906111806825,1127.59906484001;792.879554175179,1125.40437636634]];

hw\_2\_new

roi2=[538.896686703657,1199.16459951487;542.700226187475,1222.10109519001;556.300761311429,1232.35912591910;560.450077111957,1262.78744178964;564.253616595774,1266.59098127345;562.524735012221,1231.32179696896;583.040796470388,1199.85615214829;556.992313944850,1211.26677059974];

roi3=[563.422689380779,1364.12048894416;563.029272771543,1373.95590417504;594.896018119600,1374.87387626326;624.926819291226,1369.23490486422;611.419515707481,1368.18579390626;590.175018808777,1371.33312678014;569.323938519308,1366.08757199034];

hw\_2\_new

untitled3

hw\_2\_new

roi\_new=[159.216881384253,793.151528973671;150.745583958532,792.518160941842;147.974598819278,797.505934192500;155.812528213169,807.323138685859;162.304550539423,806.610599650050;180.355539446566,807.323138685859;183.522379605714,800.910287363584;179.801342418715,790.934740862268;166.738126762230,790.301372830438]

load('roiPositions.mat')

temp=roiPositions{2};

roiPositions{2}=roiiPositions{3};

roiPositions{2}=roiPositions{3};

roiPositions{3}=temp;

hw\_2\_new

untitled3

max(max(temp3))

sum(sum(temp3>10))

imshow(temp3>10)

imshow(temp3>9)

hw\_2\_new

untitled3

hw\_2\_new

r1 = [210.955823923358,369.801302205853;224.124746004478,365.928089829053;234.324205263385,372.512550869613;231.871170758078,386.327008346866;219.735105310772,385.681472950733;210.955823923358,379.871654385533;210.955823923358,373.287193344973];

r2=[[249.994466568523,375.286759382401;245.865560578568,383.006018407100;249.635431265049,390.007206824849;256.756298117290,385.339747879683;259.269545241610,374.688367209944]]

r3=[1093.31376816819,3336.23694701851;1051.50686050015,3347.97504032531;1067.10405297631,3366.78814877592;1079.64612527672,3365.50178238614;1091.22342278479,3348.77901931892]

t1=roiPositions{1}

t2=roiPositions{2}

t3=roiPositions{3}

untitled3

hw\_2\_new

untitled3

hw\_2\_new

t5=[1223.76379462968,2075.25494246090;1226.69563293007,2080.57568604308;1229.95323104161,2078.72971377988;1230.82192387135,2073.40897019770;1235.27397462378,2082.09589849513;1229.51888462673,2083.83328415462;1234.62245500148,2091;1233.21082915314,2092.84597226320;1227.56432575981,2086.65653585128;1226.15269991148,2091.54293301859;1220.83195632930,2085.67925641782;1219.74609029212,2078.72971377988];

load('roiPositions.mat')

clear all

load('roiPositions.mat')

load('t3.mat')

roiPositions{3}=t1;

load('t4.mat')

roiPositions{4}=t2;

load('t5.mat')

roiPositions{5}=t5;

load('t6.mat')

roiPositions{6}=t3;

hw\_2\_new

sampSoil=sampSoil\*2;

hw\_2\_new

for i = 1:nBands

tempCoeff = coeffs{i};

tempBand = (double((DN\_image(:,:,i)))\*tempCoeff(1)+tempCoeff(2));

tempBand(tempBand<0)=0;

max(max(tempBand))

end

for i = 1:nBands

tempCoeff = coeffs{i};

tempBand = (double((DN\_image(:,:,i)))\*tempCoeff(1)+tempCoeff(2));

tempBand(tempBand<0)=0;

sum(sum(tempBand>1))

end

iscell

iscell(roiPositions)

hw\_2\_new

K=456;

K{1}

[newROIS] = multiplicateROIS(roiPositions{1},factor)

[newROIS] = multiplicateROIS(roiPositions{1},2)

TT = avgs;

FF=stds;

hw\_2\_new

%-- 17/04/2022 12:09 --%

hw2\_main

close all

%-- 17/04/2022 13:53 --%

hw2\_main

X=zeros(12,2)

for i=1:12

X(i,:)=coeffs{i};

end

X = X';

%-- 17/04/2022 20:43 --%

hw2\_main

%-- 22/04/2022 00:51 --%

untitled

%-- 02/05/2022 19:15 --%

nanstd( 1 2)

hw3\_main

title(string(MATERIALS(i)))

hw3\_main

close all

%-- 03/05/2022 18:43 --%

hw3\_main

124-111

110-91

90-76

hw3\_main

wavelengthsFromSrm(121)

wavelengthsFromSRM(121)

wavelengthsFromSRM(201)

wavelengthsFromSRM(301)

wavelengthsFromSRM(511)

wavelengthsFromHSC(26)

wavelengthsFromHSC(53)

wavelengthsFromHSC(87)

wavelengthsFromHSC(157)

%-- 14/05/2022 20:29 --%

hw4\_main

h=imshow(coloredImg2)

hw4\_main

%-- 16/05/2022 11:23 --%

fieName(end-3:end)

fileName(5)

fileName(6:end)

fileName((end-3):end)

fileName((end-2):end)

strsplit(fileName,'.')

strsplit(fileName,'.')(end)

(strsplit(fileName,'.'))(end)

(strsplit(fileName,'.'))\*(end)

strsplit(fileName,'.')

strsplit(fileName,'.'){end}

{strsplit(fileName,'.')}{end}

{strsplit(fileName,'.')}

strsplit(fileName,'.')

ans{end}

hw5\_test

wavelenghts(wavelengths>0)

hw5\_test

%-- 21/05/2022 17:56 --%

hw5\_test

imshow(app.ImageAxes,app.Img)

imagesc(app.ImageAxes,app.Img)

daspect(app.ImageAxes,[1 1 0]);

daspect(app.ImageAxes,[1 1 1]);

hw4\_main

imshow(maskedimage)

imshow(maskedImage)

imshow(maskedImage2)

rgbImg = imrotate(rgbImg,-90);

imshow(maskedImage\_1)

imshow(maskedImage\_2)

imshow(maskedImage\_1)

numEndmembers = countEndmembersHFC(hcube,'PFA',10^-7);

numEndmembers = countEndmembersHFC(hcube)

hw4\_main

hcube.DataCube(1,1,10)

load('BW\_2.mat')

load('BW\_1.mat')

imshow(BW\_1)

imshow(imrotate(BW\_2,90))

hw4\_main

load('BW\_1.mat')

imshow(rgbImg2)

load('BW\_2.mat')

hw4\_main

samplings{1}=[col2{1};row2{1}]

samplings{1}=[col2{1},row2{1}]

tempcol = col2{2};

temprow = row2{2};

samplings{2}=[tempcol(1:25),temprow(1:25)]

samplings{3}=[tempcol(26:45),temprow(26:45)]

samplings{4}=[tempcol(46:60),temprow(46:60)]

samplings{5}=[col2{3},row2{3}]

tempcol = col2{4};

temprow = row2{4};

samplings{6}=[tempcol(1:10),temprow(1:10)]

samplings{7}=[tempcol(11:end),temprow(11:end)]

samplings{8}=[col2{5},row2{5}]

tempcol = col2{5};

temprow = row2{5};

samplings{8}=[tempcol(1:35),temprow(1:35)]

50+25+20+15+15+10+12+35

close all

num=8;

load('samplings.mat')

close all

num=7

close all

temp = samplings;

load('samplings.mat')

samplings{6}=temp;

num=7

samplings{6}=temp{6};

close all

MATERIALS = ["Trees";...

"Buildings";...

"Road";...

"Land";...

"Sidewalk";...

"Car";...

"Sky"];

h=imshow(rgbImg)

h=imshow(rgbImg);

close all

MATERIALS = ["Trees";...

"Buildings";...

"Road";...

"Soil";...

"Sidewalk";...

"Car";...

"Sky"];

close all

ylim([0 1.2])

ylim([0 1.3])

A=rand(5);

B = [1,1;1,2;3,3;1,3;5,4];

A(B)

A(1,1)

A(1,2)

xlabel('470 [nm]')

ylabel('530 [nm]')

zlabel('620 [nm]')

legend(string(MATERIALS));

%-- 09/06/2022 19:51 --%

project\_main

%-- 17/06/2022 12:58 --%

project\_main

%-- 29/06/2022 17:47 --%

project\_main

Copy\_of\_project

max(max(K(:,:,1))

max(max(K(:,:,1)))

max(max(S(:,:,1)))

Copy\_of\_project

max(max(K(:,:,11)))

min(min(K(:,:,11)))

max(max(K(:,:,1)))

Copy\_of\_project

R = K(:,:,1);

R = (R - min(min(R)))/((max(max(R))-min(min(R))));

max(max(R))

K(:,:,1)=R;

Copy\_of\_project

wavelength = [

490.000000, 560.000000, 665.000000, 705.000000, 740.000000, 783.000000,

842.000000, 865.000000, 945.000000, 1375.000000, 1610.000000, 2190.000000]

wavelength = {

490.000000, 560.000000, 665.000000, 705.000000, 740.000000, 783.000000,...

842.000000, 865.000000, 945.000000, 1375.000000, 1610.000000, 2190.000000}

wavelength = [

490.000000, 560.000000, 665.000000, 705.000000, 740.000000, 783.000000,...

842.000000, 865.000000, 945.000000, 1375.000000, 1610.000000, 2190.000000]

fwhm = [

65.000000, 35.000000, 30.000000, 15.000000, 15.000000, 20.000000,...

115.000000, 20.000000, 20.000000, 30.000000, 90.000000, 180.000000]

fwhm = fwhm'

wavelength = wavelength'

Copy\_of\_project

M=S(:,:,1)-Q(:,:,1);

max(max(M))

min(min(M))

M=S(:,:,10)-Q(:,:,10);

max(max(M))

min(min(M))

max(max(K))

resampled = hypercube('data\FenixSpectralResample');

R = resampled.DataCube;

max(max(I(:,:,1)))

min(min(I(:,:,1)))

Copy\_of\_project

max(max(K(:,:,1)))

max(max(R(:,:,1)))

788/46

V = K(:,:,1);

V = V./12;

max(max(V))

for i=1:12

Y(i) = max(max(R(:,:,i)))/max(max(K(:,:,i)));

end

Copy\_of\_project

factor = A(W-299);

G = factor.\*I(:,:,1:202);

G = factor'.\*I(:,:,1:202);

Copy\_of\_project

max(max(K(:,:,1)))

factor2 = A(W-299);

Copy\_of\_project

A(W(i)-299)

factor2 = A(W-299);

Copy\_of\_project

max(max(K(:,:,1)))

Copy\_of\_project

max(max(K(:,:,1)))

for l=1:202

N(l,1)=I(1,1,l);

end

factor2 = A(W-299);

A = T{:,2};

factor2 = A(W-299);

R(1,1,1)

project\_final

hcube.Metadata.MapInfo.PixelSize

hcube.Metadata.MapInfo.PixelSize = [10 10];

project\_final

enviwrite(hcube,'data\FenixSpatialReduce');

K=zeros(size(I));

K=single(zeros(size(I)));

REDUCE\_FACTOR\*[1 1]

REDUCE\_FACTOR=10;

REDUCE\_FACTOR\*[1 1]

project\_final

max(max(K(:,:,1))

max(max(K(:,:,1)))

max(max(I(:,:,1))

max(max(I(:,:,1)))

project\_displaying\_images

SentinelCropped = hypercube('data\SentinelACropped','data\SentinelACropped.hdr');

rgb\_sentinel = colorize(SentinelOriginal,[3 2 1],'ContrastStretching',true);

rgb\_sentinel = colorize(SentinelCropped,[3 2 1],'ContrastStretching',true);

SentinelOriginal = hypercube('data\20170411\_istm\_clbr\_8bit\_subset');

rgb = colorize(SentinelOriginal,[3 2 1]);

rgb = colorize(SentinelOriginal,[3 2 1],'ContrastStretching',true);

J = imcrop(rgbImg\_2,[636 780 39 208]);

J = imcrop(rgb,[636 780 39 208]);

M = FenixOriginal.cropData(636:780,39:208,:);

M = FenixOriginal.cropData(636:780,39:208,':');

rgb = colorize(M,[3 2 1],'ContrastStretching',true);

I = FenixOriginal.DataCube;

P = fix(I/10)\*10;

max(max(P(:,:,1)))

temp = FenixRadiometric.DataCube;

max(max(temp(:,:,1)))

A = P(:,:,5);

sum(sum(A=20))

sum(sum(A==20))

sum(sum(A==30))

sum(sum(A==40))

sum(sum(A=10))

sum(sum(A==10))

sum(sum(A==15))

tempC = hypercube('data\tempcrop','data\tempcrop.hdr');

img = colorize(tempC,[3 2 1],'ContrastStretching',true);

tempC = hypercube('data\tempcrop2','data\tempcrop2.hdr');

img = colorize(tempC,[3 2 1],'ContrastStretching',true);

SentinelCropped = hypercube('data\SentinelACropped','data\SentinelACropped.hdr');

rgb\_sentinel = colorize(SentinelCropped,[3 2 1],'ContrastStretching',true);

close all

project\_displaying\_images

imshow(rgb\_original(80:330,775:1025,:))

imshow(rgb\_original(775:1025,80:330,:))

A = rgb\_original(775:1025,80:330,:);

imshow(rgb\_original(725:1024,30:329,:))

B = (rgb\_spatial(72:102,3:32,:));

C = rgb\_spectral(725:1024,30:329,:));

C = rgb\_spectral(725:1024,30:329,:);

A = rgb\_original(725:1024,30:329,:));

A = rgb\_original(725:1024,30:329,:);

U = (A(:,:,1)-C(:,:,1));

max(max(U))

close all

entropy)I(

entropy(I)

entropy(double(I))

entropy(double(J))

entropy(double(I(:,:,1))

entropy(double(I(:,:,1)))

entropy(double(K))

entropy(double(K(:,:,1)))

entropy(double(L(:,:,1)))

entropy(double(L(:,:,3)))

histogram(double(I(:,:,5)))

A = entropy(double(I(:,:,i)));

A = entropy(double(I(:,:,1)));

W = FenixOriginal.Wavelength;

V = SentinelCropped.Wavelength;

plot(W,A,W,B,W,C)

plot(W,A)

W = round(FenixOriginal.Wavelength\*1000);

V = SentinelCropped.Wavelength;

plot(W,A)

plot(W,A)

ylim=([0 0.2]);

plot(W,A)

ylim([0 0.2]);

ylim([0 0.2]);

project\_displaying\_images

mean(A)

mean(B)

mean(C)

mean(D)

entropy(double(rgb\_original))

entropy(double(rgb\_spatial))

entropy(double(rgb\_radiometric))

entropy(double(rgb\_spectral))

F = imajust(I(:,:,1));

F = imadjust(I(:,:,1));

sum(sum(F==1))

sum(sum(F<1))

mean(A)

mean(B)

mean(C)

mean(D)

G = entropy(imadjust(I));

max(max(I(:,:,1))

max(max(I(:,:,1)))

N = double(I(:,:,1));

N(1,1,1)

I(1,1,1)

entropy(I(:,:,1))

entropy(double(I(:,:,1)))

entropy(double(round(I(:,:,1))))

N = round(I(:,:,1));

N(1,1)

[counts,binLocations] = imhist(I(:,:,1))

[counts,binLocations] = imhist(I(:,:,1)/100)

mean(A)

mean(B)

mean(C)

mean(D)

plot(V,D,'-o')

plot(W,A,'-o')

plot(W,B,'-.')

plot(V,D,'\*')

mean(D)

H = [D(1:9),D(11:12)]

H = [D(1:9);D(11:12)]

mean(H)

mean(A)

mean(D)

2086\*397

ans/(209\*40)

Y=[mean(A);mean(B);mean(C);mean(D);mean(H)]

close all

M = SentinelCropped.DataCube;

W2=SentinelCropped.Wavelength;

W-W2

V-W2

V2 = FenixSpectral.Wavelength;

V-V2

mean(D)

mean(E)

max(max(M(:,:,1)))

P = double(M/100);

max(max(P(:,:,1)))

P = double(M)/100;

max(max(P(:,:,1)))

mean(E)

max(max(L(:,:,2)))

mean(D)

mean(H)

mean(E)

Y = [mean(D);mean(H);mean(E)]

AA = rand(10);

BB = mean(AA,2);

mean(AA(1,:))

CC = stdev(AA,0,2);

CC = std(AA,0,2);

std(AA(1,:))

I = FenixOriginal.CropData(':',300:1300,':');

I = FenixOriginal.cropData(':',300:1300,':');

I = FenixOriginal.cropData(300:1300,':',':');

project\_MTF

a=avg-2\*stdv;

b = min\_value;

c=[a,b];

d=max(c);

d=max(c');

d=d';

e=[c,d];

min(I(1,:,1))

min(I(2,:,1))

project\_MTF

A = (top-bot)./(top+bot);

S = mean(M0,2);

T = std(M0,0,2);

plot(S)

plot(S+T)

plot(S,'-b')

hold on

plot(S+T,'-r')

plot(S-T,'-r')

max(M0-MM)

min(M0-MM)

project\_MTF

min(M0-MM)

B = max(max(M(:,:,:)));

B = max(max(L(:,:,:)));

B = max(max(single(M(:,:,:))));

M = single(SentinelCropped.DataCube);

plot(MTF\_3);

plot(M0);

A = reshape(M0,[100 10]);

A = reshape(M0',[100 10]);

A = reshape(M0,[10 100]);

FenixOriginal = hypercube('data\Fenix\_sub2\_sent\_band','data\Fenix\_sub2\_sent\_band.hdr');

FenixOriginal = FenixOriginal.cropData(TOP\_ROW:BOTTOM\_ROW,':',':').DataCube;

clc

project\_MTF

n\_rows = BOTTOM\_ROW - TOP\_ROW +1;

project\_MTF

close all

W = FenixOriginal.Wavelengths;

project\_MTF

W = FenixOriginal.Wavelengths;

W = FenixOriginal.Wavelength;

project\_displaying\_images

fix\_sentinel\_A

mDataSentinel.MapInfo

mDataSpatial.MapInfo

mDataSentinel.MapInfo = mDataSpatial.MapInfo;

CroppedSentinelUTM = hypercube(I,SentinelCropped.Wavelength,mDataSentinel);

enviwrite(CroppedSentinelUTM,'data\CroppedSentinelUTM');

mDataSentinel.ProjectInfo = mDataSpatial.ProjectInfo;

mDataSentinel.ProjectionInfo = mDataSpatial.ProjectionInfo;

mDataSentinel.CoordinateSystemString = mDataSpatial.CoordinateSystemString;

CroppedSentinelUTM = hypercube(I,SentinelCropped.Wavelength,mDataSentinel);

enviwrite(CroppedSentinelUTM,'data\CroppedSentinelUTM');

%-- 02/07/2022 23:13 --%

fix\_sentinel\_A

project\_displaying\_classification

I = FenixOriginal.DataCube;

project\_displaying\_classification

I = imopen('svm\_results\Sentinel.tiff');

I = imshow('svm\_results\Sentinel.tiff');

project\_displaying\_classification

I = imshow('svm\_results\Sentinel.tiff');

I = imshow('svm\_results\Sentinel.tif');

I = imshow('svm\_results\Sentinel.tif')

imshow(Sentinel)

project\_displaying\_classification

plot(rand(5,1),[1 0 0])

plot(1:5,rand(5,1),[1 0 0])

plot(1:5,rand(5,1)',[1 0 0])

plot(1:5,'LineColor','b')

plot(1:5,'Color','b')

plot(1:5,'Color',[1 0 0])

project\_displaying\_classification

project\_displaying\_images

project\_displaying\_classification

sum(sum(Spatial-Original))

sum(sum(Spectral-Original))

sum(sum(Radiometric-Original))

sum(sum(Radiometric-Spectral))

%-- 05/07/2022 18:30 --%

project\_displaying\_classification

project\_displaying\_classification\_SVM

S = imagesc(Spatial);

project\_displaying\_classification\_SVM

project\_displaying\_classification\_KM

daspect([1 1 1])

project\_displaying\_classification\_KM

project\_displaying\_plots

project\_displaying\_images

max(max(B))

min(min(B))

F = imresize(Original,0.1,'nearest');;

compare\_classification\_KM

I = FenixOriginal.DataCube;

project\_displaying\_classification\_SVM

project\_displaying\_classification\_KM

compare\_classification\_SVM

A = [1 1 0;1 1 1; 1 2 3];

B = [0 1 2;1 2 2; 0 2 2];

sum(sum(A-B==0))

sum(sum(A(A==1 & B==1)))

sum(sum(A(A==2 & B==2)))

A(A==2 & B==2)

A(A==1 & B==1)

M = (A==1 & B==1)

A(M)

M = (A==2 & B==2);

M

size(A==2 & B==2)

M = (A==1 & B==1);

size(M)

sum(sum((A==2 & B==2)))

sum(sum(A(A==2 & B==2)))

(A(A==2 & B==2)))

(A(A==2 & B==2))

compare\_classification\_SVM

tot = [AA, BB, CC,DD]

max(max(Original))

min(min(Original))

A = ((Original==i & Radiometric==i)));

A = ((Original==i & Radiometric==i));

A = sum(sum(((Original==Radiometric)));

A = sum(sum(((Original==Radiometric))));

T=A;

B = Radiometric - Original;

BB = sum(sum(B==0));

compare\_classification\_SVM

Original = hypercube('svm\_results\_new\Original\_SVM\_new',~).DataCube;

Original = hypercube('svm\_results\_new\Original\_SVM\_new').DataCube;

Original = hypercube('svm\_results\_new\Original\_SVM\_new',[]).DataCube;

sum(sum(Original~=0));

sum(sum(Original~=0))

m\*n

compare\_classification\_SVM

compare\_classification\_KM

compare\_classification\_KM\_SVM

compare\_classification\_SVM

A-B

A-C

A-D

A-E

compare\_classification\_KM

A-B

A-C

project\_displaying\_classification\_KM

A-B

A-C

A-D

A-E

A-C

project\_displaying\_classification\_KM

compare\_classification\_SVM

A-B

A-C

A-D

A-E

compare\_classification\_KM

sum(sum(Radiometric==3))

compare\_classification\_KM

FenixOriginal.Metadata.Names

FenixOriginal.Metadata.ClassNames

FenixRadiometric.Metadata.ClassNames

compare\_classification\_KM

project\_displaying\_classification\_KM

compare\_classification\_KM

compare\_classification\_SVM

compare\_classification\_KM

[A B= = sort(Names)

[A B] = sort(Names)

R = results(B,:);

compare\_classification\_KM

%-- 07/07/2022 11:40 --%

compare\_classification\_KM

C-A

compare\_classification\_KM

project\_displaying\_classification\_KM

A-C

project\_displaying\_classification\_KM

A-C

compare\_classification\_KM\_SVM

results(:,2)./results(:,1)

%-- 07/07/2022 12:02 --%

project\_displaying\_images

close all

project\_MTF

avgs = [mean(MTF\_Spatial);mean(MTF\_Radiometric);mean(MTF\_Spectral);mean(MTF\_Sentinel)];

st = [std(MTF\_Spatial);std(MTF\_Radiometric);std(MTF\_Spectral);std(MTF\_Sentinel)];

compare\_classification\_SVM

compare\_classification\_KM

plot\_SVM\_KM

compare\_classification\_KM

%-- 12/07/2022 16:26 --%

copyfile(fullfile(matlabroot,'help','toolbox','coder','examples','euclidean'))

load('euclidean\_data.mat')

test

%-- 12/07/2022 16:33 --%

mex-setup

mex -setup

%-- 12/07/2022 16:52 --%

copyfile(fullfile(matlabroot,'help','toolbox','coder','examples','euclidean'))

%-- 11/11/2022 15:36 --%

AdjustCoordinates

opts.VariableTypes(3)

AdjustCoordinates

sum(rows)

pntA = cell('WTF');

pntA = "WTF";

string(5,1)

o = table();

o.Types

A="blabla";

repmat(A,3,1)

output1 = table(zeros(n,1),repmat("",n,1),zeros(n,1),repmat("",n,1));

m = size(optionsA,1);

n = size(optionsB,1);

pntA = "bla";

m = size(optionsA,1);

n=2558

temp1.codeA{counter} = optionsA.code(i);

temp1.codeA(counter) = optionsA.code(i);

temp1.typeA(counter) = optionsA.type(i);

NaN

temp1 = table(NaN,NaN,NaN,NaN,NaN,NaN)

temp1 = table(NaN,NaN,NaN,NaN,NaN,NaN,'VariableTypes',{'double','string','double','double','string','double'},'VariableNames',["codeA","typeA","groupA","codeB","typeB","groupB"]);

temp1 = table('Size',[1 6],'VariableTypes',{'double','string','double','double','string','double'},'VariableNames',["codeA","typeA","groupA","codeB","typeB","groupB"]);

temp1 = table(-1,"",-1,-1,"",-1,'VariableNames',["codeA","typeA","groupA","codeB","typeB","groupB"]);

temp2 = table(NaN,dH\_file/100000,NaN,NaN,dist\_file,NaN,'VariableNames',["dH\_from\_DB","dH\_from\_file","dH\_diff","dist\_from\_DB","dist\_from\_file","dist\_diff"]);

temp2.dist\_from\_DB(counter) = round(sqrt((x2-x1)^2+(y2-y1)^2));

temp2.dist\_from\_file(counter) = dist\_file;

optionsA = sortrows(optionsA,["R","H"]);

tempTable = [];

tempTable = [tempTable;allOptions];

tempTable = [];

tempTable = [tempTable;allOptions];

fullOutput = [];

rows = (isnan(allOptions.dH\_from\_file));

by\_dist\_diff = allOptions(rows, :);

best\_by\_dist\_diff = by\_dist\_diff(1,:);

output\_point\_idn(i,:) = [bestOption(:,1:2),bestOption(:,4:5)];

output\_idn\_quality(i,:) = bestOption(:,7:12);

bestOptionsTable = [output\_point\_idn,output\_idn\_quality];

writetable(fullOutput,strcat('output\full\_',fileName));

writetable(bestOptionsTable,strcat('output\best\_',fileName));

inputFile{i,"num"}

output\_point\_names(i,:) = [inputFile{i,"num"},pntA,pntB];

tempTable = table(1:height(fullOutput)','VariableNames',"num");

tempTable = table((1:height(fullOutput))','VariableNames',"num");

AdjustCoordinates

tempTable = table((1:height(fullOutput))','VariableNames',"i");

fullOptionsTable = [tempTable,fullOutput];

bestOptionsTable = [output\_point\_names,output\_point\_idn,output\_idn\_quality];

writetable(fullOptionsTable,strcat('output\full\_',fileName));

writetable(bestOptionsTable,strcat('output\best\_',fileName));

AdjustCoordinates

files = {'before\_86.csv','dar2\_additional.csv','dar2\_main.csv','dar3.csv','izun\_b.csv'};

files{3}

AdjustCoordinates

writetable(fullOptionsTable,strcat('output\full\_',fileName));

writetable(bestOptionsTable,strcat('output\best\_',fileName));

AdjustCoordinates

bestOptionsTable = [output\_point\_names,output\_point\_idn,output\_coordinates,output\_idn\_quality];

writetable(bestOptionsTable,strcat('output\best\_',fileName));

AdjustCoordinates

%-- 17/11/2022 12:30 --%

AdjustCoordinates

coordinates.group(1)

A=coordinates.group;

A==8;

coordinates.group==8;

coordinates(coordinates.group==8,:)=[];

unique([1,2,3,1])

unique(coordinates.group)

unique([1,2,3,1,NaN,NaN])

AdjustCoordinates

rows = strcmpi(string(coordinates.full\_name),('36R'));

optionsA = coordinates(rows, :);

tmppnt = optionsA;

rows = strcmpi(string(coordinates.full\_name),('37R'));

optionsA = coordinates(rows, :);

tmppnt = [tmppnt;optionsA];

rows = strcmpi(string(coordinates.full\_name),('338A'));

optionsA = coordinates(rows, :);

%-- 22/11/2022 10:08 --%

AdjustCoordinates

%-- 22/11/2022 13:49 --%

AdjustCoordinates

%-- 02/12/2022 21:42 --%

AdjustCoordinates

%-- 03/12/2022 02:07 --%

AdjustCoordinates

clc

clear all

AdjustCoordinates

%-- 06/12/2022 18:59 --%

AdjustCoordinates

%-- 09/12/2022 17:27 --%

AdjustCoordinates

Adjustment

t1 = string([deltaH\_table{:,1};deltaH\_table{:,2}]);

t1 = string([deltaH\_table{:,1}]);

t2 = string([deltaH\_table{:,2}]);

Adjustment

max(v)

Adjustment

max(v)

AdjustCoordinates

mergedTable = [inputFile,output\_point\_idn,output\_coordinates,output\_idn\_quality];

rows = mergedTable.dH\_diff < 0.2;

writetable(mergedTable,strcat('output\',fileName,'\_merged\_all.csv'));

rows\_unknown = isnan(mergedTable.dist\_diff);

mergedTable\_unknown = mergedTable(~rows,:);

mergedTable\_unknown = mergedTable(~rows\_unknown,:);

mergedTable\_unknown = mergedTable(rows\_unknown,:);

rows\_good = (tempMergedTable.dH\_diff<0.2 & tempMergedTable.dist\_diff<2000) | ((isnan(tempMergedTable.dH\_diff) & tempMergedTable.dist\_diff<2000));

mergedTable\_good = tempMergedTable(rows\_good,:);

mergedTable\_bad = tempMergedTable(~rows\_good,:);

writetable(mergedTable\_bad,strcat('output\',fileName,'\_merged\_bad.csv'));

if ~REGULAR\_DH\_MODE

mergedTable\_good.dH = mergedTable\_good.dH\_from\_file;

mergedTable\_good.diff = mergedTable\_good.diff/100;

end

writetable(mergedTable\_good,strcat('output\',fileName,'\_merged\_good.csv'));

[mergedTableGood, mergedTableBad, mergedTableUnknown] = SplitMergedTable(mergedTable);

AdjustCoordinates

n=5

WKT\_column = repmat('',n,1);

WKT\_column = repmat("",n,1);

[outputMergedTable] = AddWKTField(mergedTableGood)

A = table(WKT\_column,'VariableNames',["WKT"]);

rowsA = isnan(mergedTable.codeA);

rowsB = isnan(mergedTable.codeB);

pointsNotFound = [mergedTable.A(rowsA);mergedTable.B(rowsB)];

pointsNotFound = string([mergedTable.A(rowsA);mergedTable.B(rowsB)]);

unique(pointsNotFound)

pointsNotFound = unique(string([mergedTable.A(rowsA);mergedTable.B(rowsB)]));

sort(pointsNotFound)

rowsA = isnan(mergedTable.codeA);

rowsB = isnan(mergedTable.codeB);

pointsNotFound = sort(unique(string([mergedTable.A(rowsA);mergedTable.B(rowsB)])));

notFoundPoints = table(pointsNotFound,'VariableNames',"pointName");

writetable(coordinatesUnknown,strcat('output\',fileName,'\_not\_found\_points.csv'));

pointsNotFound = sort(unique(string([mergedTable.A(rowsA);mergedTable.B(rowsB)])));

notFoundPoints = table(pointsNotFound,'VariableNames',"pointName");

writetable(notFoundPoints,strcat('output\',fileName,'\_not\_found\_points.csv'));

AdjustCoordinates

%-- 10/12/2022 20:01 --%

AdjustCoordinates

%-- 15/12/2022 09:18 --%

AdjustCoordinates

%-- 17/12/2022 18:51 --%

FindPoints

AdjustCoordinates

FindPoints

AdjustCoordinates

rows=(coordinates.code=370110);

rows=(coordinates.code==370110);

find(rows)

coordinates.group(311588) = 18;

rows=(coordinates.code==370110);

find(rows)

coordinates.group(180982) = 8;

AdjustCoordinates

rows=(coordinates.code==370110);

find(rows)

coordinates.group(311588) = 18;

coordinates.group(311588)

coordinates.group(180982) = 8;

coordinates.code(180982)

AdjustCoordinates

coordinates.group(311588) = 18;

coordinates.group(180982) = 8;

coordinates.code(180982)

AdjustCoordinates

coordinates.group(311588) = 18;

coordinates.group(180982) = 8;

coordinates.code(180982)

AdjustCoordinates

coordinates.group(311588) = 18;

coordinates.group(180982) = 8;

AdjustCoordinates

inputFile = inputFile(1:453,:);

AdjustCoordinates

inputFile = inputFile(1:453,:);

FindPoints

AdjustCoordinates

%-- 29/12/2022 22:11 --%

AdjustCoordinates

%-- 01/01/2023 09:06 --%

FindUnknownPoints

%-- 01/01/2023 09:17 --%

FindUnknownPoints

tempPoint = wantedPoints{1};

tempPoint = wantedPoints(1);

wantedPoints(1,1);

wantedPoints(1,1)

wantedPoints.pointName(1)

wantedPoints.pointName{1}

A = [connectedFromA;connectedFromA];

A = [A;connectedFromB];

[C,ia,ic] = unique(A.A,'rows');

[C,ia,ic] = unique(string(A.A),'rows');

connectedFromA = [connectedFromA;connectedFromA;connectedFromB,connectedFromA];

connectedFromA = [connectedFromA;connectedFromA;connectedFromB;connectedFromA];

[~,ia,~] = unique(string(connectedFromA.A),'rows','stable');

connectedFromA = connectedFromA(ia,:);

inputFile = sortrows(inputFile,"dH\_from\_DB","descend");

round(5.2)

meannan(1,2,NaN)

mean(1,2,NaN)

mean([1,2,NaN])

nanmean([1,2,NaN])

round(1.12345,3)

FindUnknownPoints

sum(mod(connectedFromA.dH,1)

sum(mod(connectedFromA.dH,1))

sum(mod(connectedFromB.dH,1))

sum(sum(mod(connectedFromA.dH,1)),sum(mod(connectedFromB.dH,1)))==0

(sum(mod(connectedFromA.dH,1))+sum(mod(connectedFromB.dH,1)))==0

5+NaN

FindUnknownPoints

isnan(inputFile)

height(NaN)

width(NaN)

isnan(inputFile)

width(inputFile)

A = inputFile(1,1:5)

repmat(A,5,1)

RADIUS\_XY = 2000;

approximatedCoordinates = table('Size',[n 4],'VariableTypes',{'string','double','double','double'},'VariableNames',["pointName","Y","X","H"]);

FindUnknownPoints

A = counter:counter+h-1;

tempIndexesTable.index = [counter:counter+h-1];

tempIndexesTable.index = [counter:counter+h-1]';

tempIndexesTable.pointNum = repmat(i,h,1);

FindUnknownPoints

SwapDuplicateNames

num2str(1.123000,4)

num2str(1.123000,5)

num2str(1.123000,2)

SwapDuplicateNames

namesA(18,1) = "NOT FOUND";

namesB(11,1) = "FOUND MORE THAN 1";

namesB(17,1) = "FOUND MORE THAN 1";

sum(namesA=="NOT FOUND")

WARNING("There are points for which the main name was not found",namesA);

warning("There are points for which the main name was not found",namesA);

warning("There are points for which the main name was not found",0);

CSV2REZ

AdjustCoordinates

wanted\_points\_names = {'4848','3'};

wantedPoints = table(wantedPoints');

wantedPoints = table(wantedPoints')

SwapDuplicateNames

CSV2WKT

inputFile.WKT = strrep(inputFile.WKT,':',',');

CSV2WKT

length(fileName)

CSV2WKT

iscell(fileName)

iscell({fileName})

iscell({fileName,'bbb'})

CSV2WKT

%-- 12/01/2023 11:01 --%

untitled

newStr3 = strsplit(char(newStr2),' ');

%-- 12/01/2023 11:21 --%

ReadDAT

K=strfind(newStr,'####');

K=strfind(newStr,'#####');

M = contains(newStr,'#####');

ReadDAT

find(rows\_tot)

length(14)

ReadDAT

tempRow(7)

ReadDAT

split(inputData(startLineNum-3))

tempRow = split(inputData(startLineNum-2))

length(char(tempRow(6)))

tempDate = tempRow(6);

tempDate(1:3)

tempRow = split(inputData(startLineNum-1));

measData = inputData(startLine+1:endLine-1);

measData = inputData(startLineNum+1:endLineNum-1);

coreData = inputData(startLineNum+1:endLineNum-1);

A = 1:3:10

ReadDAT

measData(split(measData)

measData(split(measData))

measData=(split(measData))

stationSumRows = split(stationSumRows);

totalSumRows = split(totalSumRows);

rb\_MeasData = measData(:,measData(11)=="Rb");

rb\_MeasData = measData(:,measData(:,11)=="Rb");

measData(:,11)

rb\_MeasData = measData(measData(:,11)=="Rb",:);

rf\_MeasData = measData(measData(:,11)=="Rf",:);

measData = measData(:,[6,7,11,12,15]);

rb\_MeasData = measData(measData(:,3)=="Rb",[1 2 4 5]);

rf\_MeasData = measData(measData(:,3)=="Rf",[1 2 4 5]);

rb\_MeasData = table(measData(measData(:,3)=="Rb",[1 2 4 5]));

rb\_MeasData = array2table(measData(measData(:,3)=="Rb",[1 2 4 5]));

rb\_MeasData = array2table(measData(measData(:,3)=="Rb",[1 2 4 5]),'VariableTypes',{'string','double','double','double'},'VariableNames',["pointName","temp","readVal","neasDist"]);

rb\_MeasData = array2table(measData(measData(:,3)=="Rb",[1 2 4 5]),'VariableNames',["pointName","temp","readVal","neasDist"]);

rb\_MeasData.temp = double(rb\_MeasData.temp);

rb\_MeasData.readVal = double(rb\_MeasData.readVal);

stationSumRows = split(stationSumRows)

tempRow=split(totalSumRows(1))

meas\_dH = tempRow(9);

H0 = tempRow(15);

tempRow=split(totalSumRows(2));

tempRow=split(totalSumRows(2))

tempRow=split(totalSumRows(1))

tempRow=split(totalSumRows(2))

tempRow=split(totalSumRows(1));

dH\_tot = tempRow(9);

H0 = tempRow(15);

tempRow=split(totalSumRows(2));

n\_stations = tempRow(7);

dist\_back = tempRow(10);

dist\_fore = tempRow(13);

H1 = tempRow(16);

startPoint = measData(1,1);

startPoint = measData(1,1);

endPoint = measData(end,1);

lenfth(unique(measDataBack.pointName))

length(unique(measDataBack.pointName))

unique(measDataBack.pointName)

measDataBack = array2table(measData(measData(:,3)=="Rb",[1 2 4 5]),'VariableNames',["pointName","temp","readVal","measDist"]);

measDataFore = array2table(measData(measData(:,3)=="Rf",[1 2 4 5]),'VariableNames',["pointName","temp","readVal","measDist"]);

length(unique(measDataBack.pointName))

dH\_calc = sum(measDataBack.readVal-measDataFore.readVal)/measMode;

measDataBack.temp = double(measDataBack.temp);

measDataBack.readVal = double(measDataBack.readVal);

measDataBack.measDist = double(measDataBack.measDist);

measDataFore.temp = double(measDataFore.temp);

measDataFore.readVal = double(measDataFore.readVal);

measDataFore.measDist = double(measDataFore.measDist);

dH\_calc = sum(measDataBack.readVal-measDataFore.readVal)/measMode;

A = measDataBack.readVal-measDataFore.readVal;

abs(dH\_calc-dH\_tot)

ReadDAT

abs(dH\_calc-dH\_tot)

double('a')

double('b')

AdjustCoordinates

TestDHBetweenPointsWihoutH

Adjustment

TestDHBetweenPointsWihoutH

A = [1 1;2 2;1 1; 3 2;];

[~,ia] = unique(A);

A(ia,:)

[~,ia] = unique(inputFile(:,2:7));

inputFile = inputFile(ia,:);

[~,ia] = unique([abs(inputFile(:,4)),inputFile(:,5:7)]);

A=[abs(inputFile(:,4)),inputFile(:,5:7)];

A=[inputFile(:,4),inputFile(:,5:7)];

A=[abs(inputFile{:,4}),inputFile(:,5:7)];

A=table(abs(inputFile{:,4}),inputFile(:,5:7));

A=table(abs(inputFile(:,4)),inputFile(:,5:7));

abs(-1)

abs([-1;-1])

K =abs(inputFile(:,4));

K =abs(inputFile(:,5));

abs(inputFile.dH)

K =abs(inputFile.dH);

A=table(abs(inputFile.dH),inputFile(:,5:7));

A=[abs(inputFile.dH),inputFile(:,5:7)];

TestDHBetweenPointsWihoutH

SDF = sum(A);

SDF = sum(A');

sum(fromPoint)

variables(i)

K = sum(abs(A)');

min(K)

X = N\_mat\U\_mat;

X2 = N\_mat\*inv(U\_mat);

X2 = inv(N\_mat)\*(U\_mat);

X2 = pinv(N\_mat)\*(U\_mat);

X=pinv(N\_mat)\*(U\_mat);

TestDHBetweenPointsWihoutH

%-- 14/01/2023 20:50 --%

TestDHBetweenPointsWihoutH

%-- 16/06/2023 19:56 --%

AdjustCoordinates

%-- 17/06/2023 19:03 --%

untitled3

addpath(fullfile(matlabroot,'toolbox','matlab','graphfun'))

untitled3

CreateGraph

CreateGraph

CreateGraph

untitled4

CreateGraph

untitled4

untitled4

untitled4

p = plot(G,'EdgeLabel',1:numedges(G));

CreateGraph

%-- 30/06/2023 15:01 --%

UnifySegments

%-- 30/06/2023 16:06 --%

AdjustCoordinates

AdjustCoordinates

AdjustCoordinates

SwapDuplicateNames

CSV2WKT

CSV2REZ

%-- 30/06/2023 18:41 --%

AdjustCoordinates

SwapDuplicateNames

CSV2WKT

CSV2REZ

CreateGraph

AdjustCoordinates

CreateGraph

%-- 30/06/2023 19:24 --%

CreateGraph

%-- 30/06/2023 19:27 --%

CreateGraph

%-- 01/07/2023 15:32 --%

CreateGraph

%-- 01/07/2023 15:39 --%

untitled

%-- 01/07/2023 15:47 --%

untitled

CreatePhiLambdaCoordinatesLists

if path(end)=='\'

A=1;

else

A=0;

end

CreatePhiLambdaCoordinatesLists

T = (coordinates.type=='duplicate');

T = (coordinates.type='duplicate');

T = (coordinates.type{2});

T = (coordinates.type=='main');

T = (string(coordinates.type)=='duplicate');

T = coordinates(string(coordinates.type)=='duplicate');

T = coordinates{string(coordinates.type)=='duplicate'};

T = coordinates(string(coordinates.type)=='duplicate',:);

CreatePhiLambdaCoordinatesLists

IG12toIGD12

clc

ReadCoordinatesFiles

CreatePhiLambdaCoordinatesLists

output\_coordinates = {coordinates,temp\_table};

output\_coordinates = [coordinates,temp\_table];

output\_coordinates = [coordinates;temp\_table];

output\_coordinates = [coordinates(1:2000,:);temp\_table];

T = coordinates(1:2000,:);

temp\_table = table(phi,lambda,'VariableNames',["phi","lambda"]);

output\_coordinates = [T,temp\_table];

writetable(output\_coordinates,strcat('E:\OneDrive\Research\matlab\coordinates\','\_GG\_coordinates.csv'));

CreatePhiLambdaCoordinatesLists

%-- 03/08/2023 12:20 --%

AdjustCoordinates

coordinates(coordinates.group==11,:)=[];

coordinates(coordinates.group==8 && (coordinates.letter~='h') && (coordinates.letter~='pt') && (coordinates.letter~='ra') && (coordinates.letter~='hsh'),:)=[];

coordinates(coordinates.group==8 & (coordinates.letter~='h') & (coordinates.letter~='pt') & (coordinates.letter~='ra') & (coordinates.letter~='hsh'),:)=[];

coordinates(coordinates.group==8 & (string(coordinates.letter)~='h') & (string(coordinates.letter)~='pt') & (string(coordinates.letter)~='ra') & (string(coordinates.letter)~='hsh'),:)=[];

SwapDuplicateNames

CSV2WKT

CSV2REZ

CreateGraph

%-- 04/08/2023 22:03 --%

CreateGraph

Copy\_of\_ComputeLoops

R = incidence(G);

Copy\_of\_ComputeLoops

R = incidence(T);

A=rand(12,4);

R(6,5)

R(1,1)

R(1,2)

S=sparse(6,5)

A=[1 0 0 0; 0 0 -1 1; -1 1 1 0; 0 0 0 -1];

A=[1 0 0 0; 0 0 -1 1; -1 1 1 0];

S=sparse(A);

Y = [ 1 2 3 4]';

S\*Y

A = zeros(13,6);

A=full(R)

R = R';

RR=R(:,1:5)

R1 = incidence(T)';

R2 = incidence(G)';

R = [R1;R2];

S = unique(R,'rows');

A = full(R);

B=full(S);

C = full(R1);

S = unique(R,'rows','stable','first');

S = unique(R,'rows','stable');

Copy\_of\_ComputeLoops

TREE = full(R1);

ALL = full(S);

ALL = full(S(1:5,:));

Copy\_of\_ComputeLoops

TREE==ALL

Copy\_of\_ComputeLoops

TREE==ALL

Copy\_of\_ComputeLoops

Af = Af(:,1:end-1);

Copy\_of\_ComputeLoops

T.Edges.EndNodes{1,1} = '2';

T.addedge(1,5,1)

T=T.addedge(1,5,1)

T = T.remedge(1)

T = T.rmedge(1)

T=T.addedge(2,3,2);

T=T.addedge(2,4,2);

T=T.addedge(6,1,2);

T=T.addedge(5,1,2);

T=T.addedge(1,4,2);

T = T.rmedge(2);

T = T.rmedge(3);

T = T.rmedge(4);

T = T.rmedge(5);

T = T.rmedge(6);

B = Bf';

B = full(B);

size(At,1)

B = full(Bf);

Bf\*At+Ac

full(Bf\*At+Ac)

full(Bf)

S = full(Af);

Copy\_of\_ComputeLoops

GetCoordinatesFromMergedTable

Copy\_of\_ComputeLoops

p.XData

p.NodeLabel

C = p.EdgeCData

C = p.EdgeCData;

doc ismember

p.XData = inputFileCoordinates.Y(ind);

p.YData = inputFileCoordinates.X(ind);

A = full(A);

A = full(Af);

B = full(Bf);

R=incidence(T);

F = T.edges.EndNotes;

F = T.edges;

F = T.Edges.EndNotes;

F = T.Edges.EndNodes;

H = G.Edges.EndNode;

H = G.Edges.EndNodes;

F

A = inputFile.A,inputFile.B;

A = inputFile.A;inputFile.B;

A = [inputFile.A;inputFile.B];

A = inputFile.A;

B = inputFile.B;

EdgeTable = table([A;B],'VariableNames',{'EndNodes'});

EdgeTable = table([string(A);string(B)],'VariableNames',{'EndNodes'});

A = string(inputFile.A);

B = string(inputFile.B);

EdgeTable = table([A,B],'VariableNames',{'EndNodes'});

EdgeTable = table([string(inputFile.A),string(inputFile.B)],inputFile.dist,'VariableNames',{'EndNodes','Weight'});

EdgeTable = [EdgeTable;inputFile];

EdgeTable = [EdgeTable,inputFile];

inputFileCoordinates.Properties.VariableNames(4) = "Name";

inputFileCoordinates.Properties.VariableNames(4) = "Group";

pointNamesOrder = unique([string(inputFile.A);string(inputFile.B)]);

pointNamesOrder = unique([string(inputFile.A);string(inputFile.B)],'stable');

M = incidence(S)';

M = M(:,1:end-1);

M=full(M);

A=full(Ac);

A-M

sum(sum(abs(ans)))

BB = full(B);

B(1,:).\*G.dH

B(1,:).\*G.Edges.dH

BB = full(B(1,:));

CC = G.Edges.dH;

sum(BB.\*CC)

sum(BB\*.CC)

BB\*CC

M = full(Bf);

N = full(B);

U = full(Af);

deltas = G.Edges.dH(ind);

CC = G.Edges.dH(ind);

B(1,:)\*CC

B(2,:)\*CC

Copy\_of\_ComputeLoops

Af=full(Af);

Ac=full(Ac);

At=full(At);

Bf=full(Bf);

B(1,:)\*G.Edges.dH

Copy\_of\_ComputeLoops

B(1,:)\*G.Edges.dH

Af=full(Af);

Bf=full(Bf);

Ac=full(Ac);

At=full(At);

B\*G.Edges.dH

Copy\_of\_ComputeLoops

B\*G.Edges.dH

Af=full(Af);

Copy\_of\_ComputeLoops

B\*G.Edges.dH

Copy\_of\_ComputeLoops

B\*G.Edges.dH

full(incidence(G)')

Copy\_of\_ComputeLoops

[T,pred] = minspantree(graph(G));

Copy\_of\_ComputeLoops

G.Endnotes(6)

G.Edges.Endnotes(6)

G.Edges.EndNotes(6)

G.Edges.EndNodes(6)

G.Edges.EndNodes(6,:)

G.Edges.EndNodes(6,:)={'1240b','5KFS'}

B\*G.Edges.dH

edgesOrderInGraph = G.Edges.EndNodes;

edgesOrderInGraph = G.Edges.EndNodes

edgesOrderInTable = [G.A,G.B]

edgesOrderInTable = [G.Edges.A,G.Edges.B]

edgesOrderInGraph==edgesOrderInTable

string(edgesOrderInGraph(:,1))==string(edgesOrderInTable(:,1))

idx = (string(edgesOrderInGraph(:,1))==string(edgesOrderInTable(:,1)) & string(edgesOrderInGraph(:,2))==string(edgesOrderInTable(:,2)));

Copy\_of\_ComputeLoops

ComputeLoops

w

ComputeLoops

B=full(B);

ComputeLoops

[a,b]=ismultigraph(G)

a=ismultigraph(G)

help subgraph

ComputeLoops

c=B(i,:);

ComputeLoops\_bad

CreateGraph

ComputeLoops\_bad

ComputeLoops

YX = [coordinatesY,coordinatesX];

YX = unique(YX,'rows');

UV = [p.XData',p.YData'];

YX-UV

YX = [coordinatesY,coordinatesX];

YX = unique(YX,'stable');

UV = [p.XData',p.YData'];

YX-UV

P.NodeLabel

p.NodeLabel

P.NodeLabel

p.NodeLabel

p.EdgeLabel

L=string(p.NodeLabel);

L=string(p.NodeLabel)';

ComputeLoops

[cycles,edgecycles] = cyclebasis(G);

V = subgraph(G,cycles{2});

plot(V)

V = subgraph(G,cycles{1});

plot(V)

[cyclesV,edgecyclesV] = cyclebasis(V);

figure;plot(S)

degree(S)

S = rmnode(S,degree(S)==0);

S = rmnode(S,find(degree(S)==0));

plot(S)

%-- 25/11/2023 20:48 --%

UnifySegments

%-- 29/11/2023 16:09 --%