

```

%% ----- Created by Dedik Romahadi
clc; clearvars; close all;

%% Spectrum Identification
% Sesuaikan RPM mesin
RPM1=1500; RPM2=1800;

prompt = {'Masukkan nilai RMS? '};
dlgtitle = 'Input RMS';
dims = [1 35];
answer = inputdlg(prompt,dlgtitle,dims);
RMS = str2double(cell2mat(answer));

cond1 = 'Kondisi mesin sangat baik.';
cond2 = 'Operasi jangka panjang masih diijinkan.';
cond3 = 'Operasi jangka pendek masih diijinkan.';
cond4 = 'Vibrasi akan menyebabkan kerusakan.';

if RMS>=0&&RMS<=2.3
    disp(cond1);
elseif RMS>2.3&&RMS<=4.5
    disp(cond2);
elseif RMS>4.5&&RMS<=7.1
    disp(cond3);
else
    disp(cond4);
end

if RMS>2.3&&RMS<=4.5; eB=1; else; eB=2; end
if RMS>4.5&&RMS<=7.1; eC=1; else; eC=2; end
if RMS>7.1; eD=1; else; eD=2; end

[file, pathname]=uigetfile({'*.txt'},'File selector');
fullpathname = strcat(pathname, file);
Text = fileread(fullpathname);

if file == 0
    disp('File not input yet!')
return;
end

fid=split(convertCharsToStrings(Text));
Fid=rmmissing(double(fid(35:end,:)));
Data = reshape(Fid,2,[]);
freq = Data(:,1); Value = Data(:,2);

```

```
plot(freq,Value, 'r'); title('Test Input Spectrum'); ylabel('Amplitude (mm/s)');  
xlabel('Frequency (Hz)');
```

```
s1=round(0.5*RPM1/60); s2=round(0.5*RPM2/60); s3=RPM1/60;  
s4=RPM2/60; s5=round(2*s3); s6=round(2*s4); s7=round(3*s3);  
s8=round(3*s4); s9=round(4*s3); s10=round(4*s4); s11=round(5*s3);  
s12=round(5*s4);
```

```
% Belt Frequency
```

```
Pulley_1=250; Pulley_2=208.34; CenterDistance=500;  
s13=round(((pi*RPM1*260)/BL(Pulley_1,Pulley_2,CenterDistance))/60);
```

```
% Bearing SKF 16008 Frequency
```

```
% BPFI
```

```
s14=23.50; s15=268.25;
```

```
% BPFO
```

```
s16=176.50; s17=211.75;
```

```
% FTF
```

```
s18=11; s19=13.25;
```

```
% BSF
```

```
s20=209.75; s21=251.50;
```

```
smat=round([s1 s2 s3 s4 s5 s6 s7 s8 s9 s10 s11 s12 s13 s14 ...  
            s15 s16 s17 s18 s19 s20 s21]);
```

```
for i=1:numel(smat)
```

```
    bm(i)=(smat(:,i)-0.5);
```

```
    bn(i)=(smat(:,i)+0.5);
```

```
    bs(:,i)=linspace(bm(i),bn(i),5);
```

```
end
```

```
bs1=freq==transpose(bs(:,1)); bs2=freq==transpose(bs(:,2));  
bs3=freq==transpose(bs(:,3)); bs4=freq==transpose(bs(:,4));  
bs5=freq==transpose(bs(:,5)); bs6=freq==transpose(bs(:,6));  
bs7=freq==transpose(bs(:,7)); bs8=freq==transpose(bs(:,8));  
bs9=freq==transpose(bs(:,9)); bs10=freq==transpose(bs(:,10));  
bs11=freq==transpose(bs(:,11)); bs12=freq==transpose(bs(:,12));  
bs13=freq==transpose(bs(:,13)); bs14=freq==transpose(bs(:,14));  
bs15=freq==transpose(bs(:,15)); bs16=freq==transpose(bs(:,16));  
bs17=freq==transpose(bs(:,17)); bs18=freq==transpose(bs(:,18));  
bs19=freq==transpose(bs(:,19)); bs20=freq==transpose(bs(:,20));  
bs21=freq==transpose(bs(:,21));
```

```
cs1=logical(sum(bs1,2)); cs2=logical(sum(bs2,2));
```

```
cs3=logical(sum(bs3,2)); cs4=logical(sum(bs4,2));
```

```
cs5=logical(sum(bs5,2)); cs6=logical(sum(bs6,2));
```

```

cs7=logical(sum(bs7,2)); cs8=logical(sum(bs8,2));
cs9=logical(sum(bs9,2)); cs10=logical(sum(bs10,2));
cs11=logical(sum(bs11,2)); cs12=logical(sum(bs12,2));
cs13=logical(sum(bs13,2)); cs14=logical(sum(bs14,2));
cs15=logical(sum(bs15,2)); cs16=logical(sum(bs16,2));
cs17=logical(sum(bs17,2)); cs18=logical(sum(bs18,2));
cs19=logical(sum(bs19,2)); cs20=logical(sum(bs20,2));
cs21=logical(sum(bs20,2));

```

```

es1=Value(cs1); es2=Value(cs2); es3=Value(cs3); es4=Value(cs4);
es5=Value(cs5); es6=Value(cs6); es7=Value(cs7); es8=Value(cs8);
es9=Value(cs9); es10=Value(cs10); es11=Value(cs11); es12=Value(cs12);
es13=Value(cs13);

```

```

% Bearing Frequency

```

```

es14=[Value(cs14); Value(cs15)]; es15=[Value(cs16); Value(cs17)];
es16=[Value(cs18); Value(cs19)]; es17=[Value(cs20); Value(cs21)];

```

```

Value = [max([es1; es2]); max([es3; es4]); max([es5; es6]);...
        max([es7; es8]); max([es9; es10]); max([es11; es12]);...
        max(es13); max([es14; es15; es16; es17])];

```

```

Freq = {'0.5X'; '1X'; '2X'; '3X'; '4X'; '5X'; 'Pulley'; 'Bearing'};

```

```

T = table(Freq,Value)

```

```

%% Testing Neural Networks

```

```

% load My_Net; Result = My_Net(Value)

```

```

% out1 = 'Vibrasi disebabkan oleh Unbalance.';
% out2 = 'Vibrasi disebabkan oleh Misalignment.';
% out3 = 'Vibrasi disebabkan oleh Looseness.';
% out4 = 'Terjadi kerusakan pada bearing.';
% out5 = 'Terjadi kerusakan pada pulley.';

```

```

% if Result(1,:) == max(Result)
%     disp(out1);
% elseif Result(2,:) == max(Result)
%     disp(out2);
% elseif Result(3,:) == max(Result)
%     disp(out3);
% elseif Result(4,:) == max(Result)
%     disp(out4);
% else
%     disp(out5);
% end

```