In [1]:
```
import pandas as d
import matplotlib.pyplot as m
import seaborn as sns
import numpy as n
from scipy.interpolate import make_interp_spline
from scipy import interpolate
```

In [2]:
```
from matplotlib import style
m.style.use('ggplot')
```

In [3]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [4]:
```
df=d.read_excel('ritesh.xlsx',engine="openpyxl")
```

In [5]:
```
df.drop(['Sample No','WPSF','WRFF'],axis=1,inplace=True)
```

In [6]:
```
df_repeated=df.iloc[26:32]
df_repeated.describe()
```

Out[6]:

	V	I	WS	NPD	GFR	BW	RH	P	%D
count	6.0	6.0	6.0	6.0	6.0	6.000000	6.000000	6.000000	6.000000
mean	26.0	210.0	6.0	19.0	19.0	8.776667	3.380000	2.130000	32.601117
std	0.0	0.0	0.0	0.0	0.0	0.698847	0.177876	0.191207	1.743806
min	26.0	210.0	6.0	19.0	19.0	7.950000	3.250000	1.820000	29.756900
25%	26.0	210.0	6.0	19.0	19.0	8.407500	3.290000	2.035000	31.981350
50%	26.0	210.0	6.0	19.0	19.0	8.525000	3.310000	2.170000	32.671300
75%	26.0	210.0	6.0	19.0	19.0	9.362500	3.375000	2.282500	33.689450
max	26.0	210.0	6.0	19.0	19.0	9.650000	3.730000	2.310000	34.727000

In [7]:
```
df.drop(df.index[26:32], axis=0,inplace=True)
```

In [8]:
```
df2 = {'V':26, 'I': 210, 'WS': 6.0,'NPD':19,'GFR':19,'BW':8.52,'RH':3.31,'P':
df_new = df.append(df2, ignore_index = True)
display(df_new)
```

	V	I	WS	NPD	GFR	BW	RH	P	%D
0	25.0	200.0	5.5	18.0	20.0	7.20	3.25	2.38	35.7384
1	27.0	200.0	5.5	18.0	18.0	9.37	3.61	1.99	35.7579
2	25.0	220.0	5.5	18.0	18.0	8.55	3.43	2.62	35.7184
3	27.0	220.0	5.5	18.0	20.0	8.47	3.64	3.54	41.7022
4	25.0	200.0	6.5	18.0	18.0	7.74	2.65	1.57	29.1295
5	27.0	200.0	6.5	18.0	20.0	8.72	3.43	2.08	29.5118
6	25.0	220.0	6.5	18.0	20.0	6.47	3.25	2.48	35.4975
7	27.0	220.0	6.5	18.0	18.0	8.48	3.38	3.20	38.0469
8	25.0	200.0	5.5	20.0	18.0	7.24	3.40	1.62	29.2807
9	27.0	200.0	5.5	20.0	20.0	7.45	3.50	1.90	33.8893

	V	I	WS	NPD	GFR	BW	RH	P	%D
10	25.0	220.0	5.5	20.0	20.0	9.29	4.00	2.14	29.0041
11	27.0	220.0	5.5	20.0	18.0	9.09	3.60	2.08	28.7846
12	25.0	200.0	6.5	20.0	20.0	6.55	3.40	1.85	26.9854
13	27.0	200.0	6.5	20.0	18.0	7.10	3.04	2.02	27.6735
14	25.0	220.0	6.5	20.0	18.0	7.47	3.47	2.18	35.8673
15	27.0	220.0	6.5	20.0	20.0	8.55	3.42	2.20	33.6621
16	24.0	210.0	6.0	19.0	19.0	7.27	3.19	1.58	26.0334
17	28.0	210.0	6.0	19.0	19.0	10.31	3.49	3.07	38.4123
18	26.0	190.0	6.0	19.0	19.0	7.51	3.24	1.62	28.8026
19	26.0	230.0	6.0	19.0	19.0	8.05	3.46	2.28	33.1148
20	26.0	210.0	5.0	19.0	19.0	7.79	3.88	3.00	39.3728
21	26.0	210.0	7.0	19.0	19.0	7.48	3.08	2.68	38.3701
22	26.0	210.0	6.0	17.0	19.0	8.91	2.96	2.55	38.7384
23	26.0	210.0	6.0	21.0	19.0	9.30	3.38	1.50	29.0880
24	26.0	210.0	6.0	19.0	17.0	8.20	3.20	2.31	35.0581
25	26.0	210.0	6.0	19.0	21.0	8.28	3.81	2.40	37.2333
26	26.0	210.0	6.0	19.0	19.0	8.52	3.31	2.17	32.6700

In [9]:
```
df
```

Out[9]:

	V	I	WS	NPD	GFR	BW	RH	P	%D
0	25	200	5.5	18	20	7.20	3.25	2.38	35.7384
1	27	200	5.5	18	18	9.37	3.61	1.99	35.7579
2	25	220	5.5	18	18	8.55	3.43	2.62	35.7184
3	27	220	5.5	18	20	8.47	3.64	3.54	41.7022
4	25	200	6.5	18	18	7.74	2.65	1.57	29.1295
5	27	200	6.5	18	20	8.72	3.43	2.08	29.5118
6	25	220	6.5	18	20	6.47	3.25	2.48	35.4975
7	27	220	6.5	18	18	8.48	3.38	3.20	38.0469
8	25	200	5.5	20	18	7.24	3.40	1.62	29.2807
9	27	200	5.5	20	20	7.45	3.50	1.90	33.8893
10	25	220	5.5	20	20	9.29	4.00	2.14	29.0041
11	27	220	5.5	20	18	9.09	3.60	2.08	28.7846
12	25	200	6.5	20	20	6.55	3.40	1.85	26.9854
13	27	200	6.5	20	18	7.10	3.04	2.02	27.6735
14	25	220	6.5	20	18	7.47	3.47	2.18	35.8673
15	27	220	6.5	20	20	8.55	3.42	2.20	33.6621
16	24	210	6.0	19	19	7.27	3.19	1.58	26.0334

	V	I	WS	NPD	GFR	BW	RH	P	%D
17	28	210	6.0	19	19	10.31	3.49	3.07	38.4123
18	26	190	6.0	19	19	7.51	3.24	1.62	28.8026
19	26	230	6.0	19	19	8.05	3.46	2.28	33.1148
20	26	210	5.0	19	19	7.79	3.88	3.00	39.3728
21	26	210	7.0	19	19	7.48	3.08	2.68	38.3701
22	26	210	6.0	17	19	8.91	2.96	2.55	38.7384
23	26	210	6.0	21	19	9.30	3.38	1.50	29.0880
24	26	210	6.0	19	17	8.20	3.20	2.31	35.0581
25	26	210	6.0	19	21	8.28	3.81	2.40	37.2333

In [9]:
```
X=df_new.loc[:,['V','I','WS','NPD','GFR']]          # Features!
Y=df_new['BW']   # Target!

# Copy of original X and Y!!
X_copy=X
Y_copy=Y
```

In [10]:
```
from sklearn import preprocessing
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import GridSearchCV
from sklearn.svm import SVR
from sklearn.metrics import mean_absolute_error
from sklearn.metrics import mean_squared_error
from sklearn.metrics import mean_absolute_percentage_error
from math import sqrt
```

In [11]:
```
Sx = StandardScaler()
Sy= StandardScaler()
X_t = Sx.fit_transform(X)   # using subscript "te" for testing!
Y_t = Sy.fit_transform(Y.values.reshape(-1,1))
```

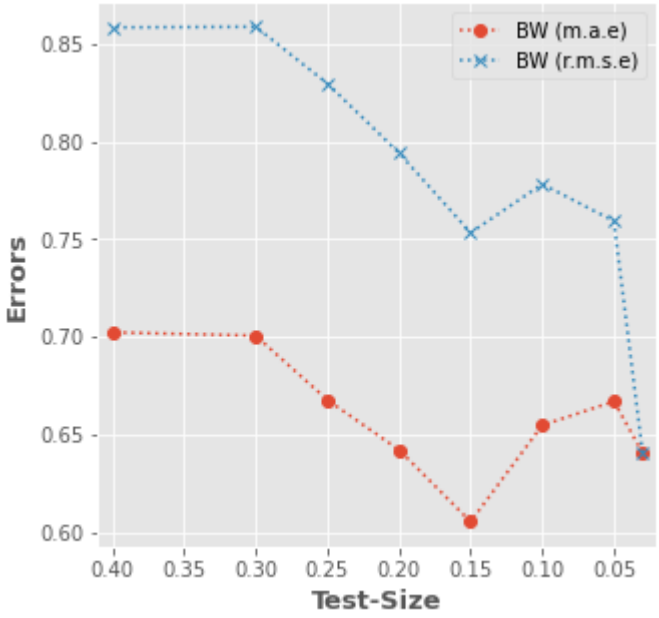
In [12]:
```
from sklearn.model_selection import train_test_split
```

In [13]:
```
testing_mae=[]
testing_rmse=[]


split_percent=[0.4,0.3,0.25,0.2,0.15,0.1,0.05,0.03]

split_percent.reverse()

for i in split_percent:

    sum_maeTe=0      # Intiate sum=0 beform summing in random shuffles..!!
    sum_rmseTe=0

    count=0

    for r_s in range(0,100,2):    # 50 random shuffles!!

        X_train, X_test, Y_train, Y_test = train_test_split(X_t, Y_t, test_si

        svr = SVR(kernel='rbf',C=1.5,epsilon=0.01)
        svr.fit(X_train, Y_train.ravel())
```

```
        y_predTe = svr.predict(X_test)
        y_predTe = Sy.inverse_transform(y_predTe.reshape(-1, 1))

        maeTe = mean_absolute_error(Sy.inverse_transform(Y_test.reshape(-1, 1
        rmseTe = sqrt(mean_squared_error(Sy.inverse_transform(Y_test.reshape(

        sum_maeTe+=maeTe        # Sum at all 50 random shuffles..!!
        sum_rmseTe+=rmseTe

        count+=1

    testing_mae.append(sum_maeTe/count)     # Averaging..!!
    testing_rmse.append(sum_rmseTe/count)
```

In [14]:
```
m.figure(figsize=(5, 5))
m.plot(split_percent,testing_mae,label='BW (m.a.e)',marker='o',linestyle=':')
m.plot(split_percent,testing_rmse,label='BW (r.m.s.e)',marker='x',linestyle='
m.xlim([0.41,0.02])
m.xlabel("Test-Size",fontsize=13,fontweight='bold')
m.ylabel("Errors",fontsize=13,fontweight='bold')
m.legend()
#m.savefig("16BW.jpg",dpi=2000)
```

Out[14]: <matplotlib.legend.Legend at 0x7fd4cfa576a0>



In [20]:
```
df_new
```

Out[20]:

	V	I	WS	NPD	GFR	BW	RH	P	%D
0	25.0	200.0	5.5	18.0	20.0	7.20	3.25	2.38	35.7384
1	27.0	200.0	5.5	18.0	18.0	9.37	3.61	1.99	35.7579
2	25.0	220.0	5.5	18.0	18.0	8.55	3.43	2.62	35.7184
3	27.0	220.0	5.5	18.0	20.0	8.47	3.64	3.54	41.7022
4	25.0	200.0	6.5	18.0	18.0	7.74	2.65	1.57	29.1295
5	27.0	200.0	6.5	18.0	20.0	8.72	3.43	2.08	29.5118
6	25.0	220.0	6.5	18.0	20.0	6.47	3.25	2.48	35.4975
7	27.0	220.0	6.5	18.0	18.0	8.48	3.38	3.20	38.0469

	V	I	WS	NPD	GFR	BW	RH	P	%D
8	25.0	200.0	5.5	20.0	18.0	7.24	3.40	1.62	29.2807
9	27.0	200.0	5.5	20.0	20.0	7.45	3.50	1.90	33.8893
10	25.0	220.0	5.5	20.0	20.0	9.29	4.00	2.14	29.0041
11	27.0	220.0	5.5	20.0	18.0	9.09	3.60	2.08	28.7846
12	25.0	200.0	6.5	20.0	20.0	6.55	3.40	1.85	26.9854
13	27.0	200.0	6.5	20.0	18.0	7.10	3.04	2.02	27.6735
14	25.0	220.0	6.5	20.0	18.0	7.47	3.47	2.18	35.8673
15	27.0	220.0	6.5	20.0	20.0	8.55	3.42	2.20	33.6621
16	24.0	210.0	6.0	19.0	19.0	7.27	3.19	1.58	26.0334
17	28.0	210.0	6.0	19.0	19.0	10.31	3.49	3.07	38.4123
18	26.0	190.0	6.0	19.0	19.0	7.51	3.24	1.62	28.8026
19	26.0	230.0	6.0	19.0	19.0	8.05	3.46	2.28	33.1148
20	26.0	210.0	5.0	19.0	19.0	7.79	3.88	3.00	39.3728
21	26.0	210.0	7.0	19.0	19.0	7.48	3.08	2.68	38.3701
22	26.0	210.0	6.0	17.0	19.0	8.91	2.96	2.55	38.7384
23	26.0	210.0	6.0	21.0	19.0	9.30	3.38	1.50	29.0880
24	26.0	210.0	6.0	19.0	17.0	8.20	3.20	2.31	35.0581
25	26.0	210.0	6.0	19.0	21.0	8.28	3.81	2.40	37.2333
26	26.0	210.0	6.0	19.0	19.0	8.52	3.31	2.17	32.6700

In [ ]: