Cheching the solubility of molecule in water

Load Data

```
import pandas as pd
df  = pd.read_csv('https://raw.githubusercontent.com/dataprofessor/data/master/delaney_solubility_with_descriptors.csv')
df.head()
```

	MolLogP	MolWt	NumRotatableBonds	AromaticProportion	logS
0	2.5954	167.850	0.0	0.0	-2.18
1	2.3765	133.405	0.0	0.0	-2.00
2	2.5938	167.850	1.0	0.0	-1.74
3	2.0289	133.405	1.0	0.0	-1.48
4	2.9189	187.375	1.0	0.0	-3.04

Data Preparation

Data Separation as X and y

```
y = df["logS"]
y
```

```
0      -2.180
1      -2.000
2      -1.740
3      -1.480
4      -3.040
        ...
1139    1.144
1140   -4.925
1141   -3.893
1142   -3.790
1143   -2.581
Name: logS, Length: 1144, dtype: float64
```

```
X = df.drop(["logS"], axis=1)
X
```

	MolLogP	MolWt	NumRotatableBonds	AromaticProportion
0	2.59540	167.850	0.0	0.000000
1	2.37650	133.405	0.0	0.000000
2	2.59380	167.850	1.0	0.000000
3	2.02890	133.405	1.0	0.000000
4	2.91890	187.375	1.0	0.000000
...	...	...	...	...
1139	1.98820	287.343	8.0	0.000000
1140	3.42130	286.114	2.0	0.333333
1141	3.60960	308.333	4.0	0.695652
1142	2.56214	354.815	3.0	0.521739
1143	2.02164	179.219	1.0	0.461538

1144 rows × 4 columns

Data Spliting

```
import sklearn
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
X_train.shape
```

```
(915, 4)
```

```
X_test.shape
```

```
(229, 4)
```

Model Building

Linear Regression

Training The Model

```
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
```

```
lr.fit(X_train, y_train)
```

```
LinearRegression
LinearRegression()
```

Applying a model to make a prediction

```
y_lr_train_pred = lr.predict(X_train)
y_lr_test_pred = lr.predict(X_test)
```

```
y_lr_train_pred
```

```
array([ -4.47026156,  -5.88839323,  -4.22029229,  -0.79352323,
        -1.73062942,  -1.26187649,  -1.43969557,  -2.36087206,
        -2.77481055,  -2.23755209,  -4.15424648,  -8.49226359,
        -4.26355722,  -1.5773174 ,  -2.61749814,  -0.72935686,
        -3.18513222,  -3.71048234,  -1.57995763,  -5.11539503,
        -1.90340063,  -4.18841892,  -4.85523085,  -2.13317387,
        -3.34701685,  -3.45842365,  -3.73715663,  -3.61882675,
        -3.02530867,  -2.82178231,  -1.54861269,  -2.73630968,
        -1.79353838,  -6.49056433,  -2.19068424,  -4.10802907,
        -2.7303035 ,  -2.28598248,  -2.60739155,  -3.64519507,
        -2.65218678,  -4.35285798,   1.91721982,  -3.70871993,
        -3.50138017,  -2.71898539,  -4.77720737,  -3.38184768,
        -1.71815824,  -2.79713356,  -5.26774517,  -2.96884167,
        -2.27709013,  -2.76017539,  -2.12748391,  -2.3140745 ,
        -7.74681081,  -3.21253144,  -1.76376134,  -4.17337487,
        -2.93249251,  -0.51506669,  -3.06834756,  -0.72788913,
        -4.05296417,  -2.03616338,  -2.67036427,  -6.08064174,
        -5.49773266,  -2.9647599 ,  -2.96100034,  -2.60084617,
        -0.6827446 ,  -4.3585946 ,  -4.0043512 ,  -0.79738924,
        -3.34044901,  -1.56377011,  -2.19761587,  -1.11416728,
        -1.82672415,  -4.54679498,  -5.39336962,  -3.35057255,
        -1.43769254,  -2.59386818,  -5.0338893 ,  -4.99765259,
        -4.8997667 ,  -1.16441496,  -2.64509058,   1.39936011,
         0.01242659,  -2.20985267,  -1.37887588,  -5.53667654,
        -1.44974826,  -0.7796894 ,  -6.55748455,  -3.43117797,
        -3.38057188,  -1.44085591,  -3.67783751,  -4.1593352 ,
        -2.6803058 ,  -0.58590302,  -4.77815138,  -2.54561346,
        -0.42618347,  -0.70287549,  -2.06797098,  -3.00792612,
        -2.83552267,  -1.54025749,  -6.01542152,  -1.81446383,
```

```
 -4.98860572,  -5.1403329 ,  -2.07607676,  -0.56359351,
 -0.79092247,  -2.38738366,  -2.87529252,  -4.25585109,
 -1.85688211,  -5.47997057,  -0.15237579,  -3.86722215,
 -3.40567511,  -1.16828096,  -3.0571832 ,  -2.7303035 ,
 -4.85523085,  -1.38125051,  -1.70910714,  -2.93481005,
 -2.27419885,  -3.94320804,  -2.91347828,  -3.79600415,
 -6.64215698,  -0.82000533,  -3.76867198,  -0.51613106,
 -1.81981733,  -3.96125236,  -3.54500416,  -2.37052336,
 -3.41259165,  -1.79353838,  -3.39363131,  -1.45279161,
 -4.28364627,  -2.44484282,  -1.41163293,  -2.24540864,
 -3.01887358,  -5.24422393,  -2.996998  ,  -2.37455166,
 -5.14660978,  -7.38690482,  -3.16310546,  -1.98006043,
 -4.95579009,   0.04116924,  -2.62183065,  -6.08064174,
 -1.93958904,  -1.71610939,  -2.46165126,  -4.32947161,
 -2.17947599,  -2.70709475,  -0.24806662,  -3.73405795,
 -1.35587169,  -2.04768625,  -1.62686618,  -1.2679672 ,
 -4.19146993,  -1.53917269,  -2.39596484,  -2.35071265,
 -6.49909096,  -3.43202403,  -1.481193  ,  -7.74681081,
 -0.96842783,  -2.74810203,  -1.85208473,  -7.74681081,
 -2.8401647 ,  -4.38425128,  -2.65184786,   0.33631985,
 -1.53329109,  -1.65450802,  -2.78712359,  -4.84716047,
 -1.72676342,  -0.75919502,  -1.481193  ,  -1.05034745,
 -1.90619841,  -2.23348727,  -6.62740455,  -0.06538507,
 -5.60498883,  -1.73449543,  -3.85457389,  -2.07999886,
 -6.03150164,  -2.24778827,  -2.1143171 ,  -1.55596252,
 -8.43136399,  -4.17536325,  -2.36563493,  -1.63779456,
  1.67007582,  -4.21620768,  -4.924424  ,  -2.50017466,
 -2.35141347,  -0.46419517,  -3.96774146,  -3.10324961,
 -3.51065394,  -3.69536693,  -1.12854214,  -1.30087645
```

```
y_lr_test_pred
```

```
array([-1.53917269, -6.08287944, -5.28375883, -3.27746087, -1.70320929,
       -5.63167711, -1.43239582, -2.74544339, -3.43274443, -3.79048137,
       -1.26884289, -1.54861269, -3.72258406, -3.25962057, -3.76344744,
       -2.13317387, -2.00365994, -4.80828381, -1.43582956, -2.81358533,
       -2.37870232, -1.45080608, -3.71048234, -3.06158669, -1.12854214,
       -1.11007232, -2.45566761, -5.0338893 , -3.29087978, -4.21832264,
       -2.45834377, -3.09200621, -3.89726792, -1.85208473, -2.17761301,
       -6.01542152, -1.06493784, -2.35071265, -1.72034021, -1.45695216,
       -1.85208473, -6.66998091, -1.23899725, -1.46015868, -1.65115215,
       -3.79048137, -4.73551134, -1.09859266, -0.69404611, -1.91048017,
       -3.16121056, -2.34753749, -2.01980335, -2.51436431, -7.04012802,
       -5.27488937, -4.18979204, -2.9647599 , -1.7364388 , -1.64388527,
       -4.60884756, -4.84900677, -0.18955927, -1.73839902, -1.63779456,
       -2.01616982, -1.46391856, -7.64117637, -1.34518182, -2.28204094,
       -4.61251682, -3.68259142, -3.22251183, -5.16075847, -2.1483811 ,
       -2.07607676, -1.81561364, -2.35312712, -5.03153948, -5.35648355,
       -0.81613933, -3.67501678, -5.63795589, -0.78415377, -3.13201883,
       -1.81174763, -5.14700571,  1.13125267, -0.47305289, -3.93153406,
       -4.8733322 , -1.17801962, -2.15050196, -2.58388566,  1.12994853,
       -8.66482607, -3.8983998 , -1.59441957, -2.20288627, -5.26282707,
       -2.76987509, -5.42806373, -1.27911139, -2.33457807, -4.6588233 ,
       -7.26140165, -5.34312066, -0.66718494, -6.38434339, -4.01980772,
       -2.46854687, -5.3318694 , -4.60015378, -6.37044332, -2.33428187,
       -1.91765773, -2.22777384, -2.07999886, -3.3320867 , -0.75919502,
       -1.01798361, -2.98085644, -4.37252867, -5.3111226 , -2.72015257,
       -2.70246249, -1.61121439,  1.40322611, -4.2837828 , -3.89106644,
       -0.7259909 , -2.29209364, -3.22644351, -0.39254905, -3.1498734 ,
       -5.70731435, -5.29651733, -1.75260761, -1.43582956, -2.72900687,
       -3.92961996, -3.74280639, -1.9653113 , -1.33821542, -3.89726792,
       -6.08562232, -4.4730887 , -4.54440128, -6.67351198, -3.12186165,
       -1.44588226, -1.44892561, -1.63121936, -2.09005156, -6.08562232,
       -4.29427126, -2.07752796, -3.58240751, -1.07499053, -4.92907534,
       -6.49056433, -1.06880384, -2.46094328, -3.05417721, -4.9802434 ,
       -5.77353535, -3.97932544, -2.35071265, -2.28890565, -0.84735767,
       -2.82178231, -1.43582956, -0.98497997, -4.19034629, -2.51659019,
       -6.02394669, -2.72733014, -2.13423693, -4.61167881, -2.54561346,
       -3.48722684, -2.72733014, -0.89204913, -2.16864249, -4.36489994,
       -4.19146993, -0.53725099, -0.96961412, -5.05314827, -1.0193366 ,
       -6.38434339, -3.46506424, -1.91048017, -1.31048759, -2.19960141,
       -2.36072985, -4.89274622, -1.70320929, -0.10079998, -2.09823418,
       -3.58726393, -1.50010279, -2.02346393, -2.37957801, -1.63988079,
       -2.36563493, -1.36979039, -4.74043162, -2.80897252, -1.54419903,
       -2.12829672, -2.59386818, -3.72312418, -3.71134074, -3.29215116,
       -2.72733014, -3.29183556, -3.2715436 , -3.79048137, -3.65309971,
       -2.08696527, -2.46165126, -6.49056433, -1.71743786,  0.22974901,
       -1.91509076, -3.00635454, -3.10567883, -1.48599039])
```

Evaluate Model Performance

```
from sklearn.metrics import mean_squared_error,r2_score
lr_train_mse = mean_squared_error(y_train, y_lr_train_pred)
lr_train_r2 = r2_score(y_train, y_lr_train_pred)

lr_test_mse = mean_squared_error(y_test, y_lr_test_pred)
lr_test_r2 = r2_score(y_test, y_lr_test_pred)



print("LR MSE (Train):",lr_train_mse)
print("LR R2 (Train):",lr_train_r2)
print("LR MSE (Test):",lr_test_mse)
print("LR R2 (Test):",lr_test_r2)
```

```
LR MSE (Train): 1.0139894491573003
LR R2 (Train): 0.7695127746587307
LR MSE (Test): 0.9990844407075306
LR R2 (Test): 0.7705650058569232
```

```
lr_results = pd.DataFrame(['Linear Regression',lr_train_mse,lr_train_r2,lr_test_mse,lr_test_r2]).transpose()
lr_results.columns = ['Model','Training MSE','Training R2','Testing MSE','Testing R2 ']
lr_results
```

	Model	Training MSE	Training R2	Testing MSE	Testing R2
0	Linear Regression	1.013989	0.769513	0.999084	0.770565

Random Forest

```
from sklearn.ensemble import RandomForestRegressor
rf = RandomForestRegressor(max_depth=2,random_state=100)
```

Training The Model

```
rf.fit(X_train, y_train)
```

```
RandomForestRegressor
RandomForestRegressor(max_depth=2, random_state=100)
```

Applying a model to make a prediction

```
y_rf_train_pred = rf.predict(X_train)
y_rf_test_pred = rf.predict(X_test)
```

```
y_rf_train_pred
```

```
array([-4.39406068, -7.01386075, -4.1790564 , -1.17794119, -2.2850991 ,
       -1.33950462, -1.25181203, -2.66705519, -2.65236412, -2.31264491,
       -4.1790564 , -7.01386075, -4.06327457, -1.33950462, -2.62909993,
       -1.17794119, -2.68110568, -4.39406068, -1.48786212, -5.44303134,
       -1.36708347, -4.39406068, -5.41614062, -1.48334237, -4.33728987,
       -4.09424671, -2.68110568, -2.48465432, -2.48465432, -4.3016489 ,
       -1.35145256, -2.68110568, -2.29709247, -7.01386075, -1.51301599,
       -4.33728987, -2.65236412, -2.66705519, -1.52706648, -4.09424671,
       -2.66705519, -4.39406068, -1.36550305, -4.39406068, -3.60378319,
       -2.65236412, -4.39406068, -4.39406068, -1.35145256, -2.68110568,
       -5.44303134, -3.50229491, -2.66705519, -2.66705519, -1.36550305,
       -2.65236412, -7.01386075, -2.68110568, -1.36550305, -4.09424671,
       -4.21302627, -1.17794119, -2.68110568, -1.35145256, -4.1790564 ,
       -1.51301599, -1.36550305, -7.01386075, -6.90556761, -3.70848106,
       -1.52706648, -2.47609271, -1.36550305, -3.92155071, -4.07753611,
       -1.17794119, -4.33728987, -1.35145256, -2.48465432, -1.17794119,
       -1.46929188, -4.39406068, -4.39406068, -2.68110568, -2.2850991 ,
       -2.66705519, -5.44303134, -4.39406068, -5.44303134, -1.17794119,
       -4.07753611, -1.36550305, -1.17794119, -1.39584443, -1.36550305,
       -5.44303134, -1.25181203, -1.17794119, -7.01386075, -2.68110568,
```

```
       -4.1790564 , -1.25181203, -4.33728987, -3.69041159, -1.49752668,
       -1.36550305, -4.1790564 , -3.58839215, -1.17794119, -1.36550305,
       -1.33950462, -2.66705519, -2.62909993, -1.36550305, -7.01386075,
       -1.36708347, -4.39406068, -5.44303134, -1.33950462, -1.17794119,
       -1.17794119, -2.61504944, -2.68110568, -4.39406068, -2.4439194 ,
       -4.39406068, -1.17794119, -3.52046149, -2.68110568, -1.17794119,
       -2.68110568, -2.65236412, -5.41614062, -1.17794119, -2.44153808,
       -2.62909993, -2.49354382, -2.68110568, -3.4357199 , -2.48465432,
       -7.01386075, -1.17794119, -3.44977039, -1.36550305, -2.34464946,
       -4.39406068, -3.60378319, -1.43937388, -3.70848106, -2.29709247,
       -4.39406068, -1.33950462, -4.19542023, -2.56814024, -1.35145256,
       -2.37168888, -4.12744416, -7.01386075, -2.66705519, -2.68157965,
       -7.01386075, -7.01386075, -2.68110568, -1.33950462, -4.39406068,
       -1.36550305, -1.39584443, -7.01386075, -1.33950462, -2.2850991 ,
       -2.50346976, -4.39406068, -2.49354382, -4.07753611, -1.33676149,
       -2.68110568, -1.25181203, -2.44153808, -2.29709247, -1.36550305,
       -4.39406068, -1.33950462, -2.60035838, -2.47146521, -2.68110568,
       -4.1790564 , -1.20828257, -7.01386075, -1.36550305, -1.83032288,
       -1.69041032, -7.01386075, -1.39584443, -4.39406068, -2.66705519,
       -1.17794119, -2.29709247, -1.36550305, -2.66705519, -5.44303134,
       -2.2850991 , -1.17794119, -1.20828257, -1.35145256, -2.29709247,
       -1.36550305, -7.01386075, -1.36550305, -5.44303134, -2.2850991 ,
       -4.39406068, -2.49354382, -5.31583946, -2.65236412, -2.60035838,
       -1.17794119, -7.01386075, -4.33728987, -2.45591277, -2.29709247,
       -1.36550305, -4.07753611, -4.39406068, -2.49354382, -2.49354382,
       -1.33676149, -4.39406068, -4.39406068, -4.37313972, -4.33728987,
       -1.17794119, -1.20828257, -1.25181203, -7.01386075, -4.39406068,
       -2.68110568, -1.17794119, -7.01386075, -2.68110568, -2.66705519,
       -4.12744416, -1.36550305, -4.39406068, -1.52706648, -1.41063233,
       -1.36550305, -5.44303134, -2.68110568, -1.1935721 , -4.39406068,
       -2.68110568, -1.36550305, -2.49354382, -1.41410104, -2.62909993,
       -2.62909993, -4.39406068, -4.37313972, -1.1935721 , -4.35335683,
       -2.68110568, -2.47266096, -2.66705519, -4.39406068, -2.49354382,
       -1.20828257, -1.36550305, -1.33950462, -4.33728987, -4.39406068,
       -7.01386075, -1.17794119, -1.17794119, -2.47060383, -1.33676149,
       -1.52706648, -1.17794119, -2.48465432, -7.01386075, -1.38179394,
       -2.47060383, -7.01386075, -2.65236412, -1.36550305, -1.17794119,
       -4.39406068, -4.06327457, -2.68110568, -1.17794119, -2.65236412,
```

y_rf_test_pred

```
array([-1.33950462, -6.98577329, -4.39406068, -2.62909993, -1.35145256,
       -5.44303134, -1.35145256, -4.1790564 , -2.47266096, -4.33728987,
       -1.36550305, -1.35145256, -2.68110568, -2.8547836 , -3.70848106,
       -1.48334237, -2.49354382, -4.39406068, -1.25181203, -2.66705519,
       -2.49354382, -1.35145256, -4.39406068, -3.52046149, -1.17794119,
       -1.17794119, -2.49354382, -5.44303134, -2.62909993, -4.1790564 ,
       -1.48334237, -2.68110568, -4.37313972, -1.69041032, -2.66705519,
       -7.01386075, -1.17794119, -2.47146521, -2.29709247, -1.64276102,
       -1.69041032, -7.01386075, -1.36550305, -1.30996483, -2.29709247,
       -4.33728987, -4.09424671, -1.36550305, -1.17794119, -1.38179394,
       -3.60378319, -2.66705519, -1.36550305, -1.52706648, -7.01386075,
       -5.46984259, -4.07753611, -3.70848106, -1.25253761, -1.36550305,
       -4.39406068, -4.39406068, -1.17794119, -1.33950462, -2.29709247,
       -1.48347619, -1.36550305, -7.01386075, -1.36550305, -1.39584443,
       -5.12506004, -4.39406068, -2.68110568, -5.41614062, -1.36550305,
       -1.33950462, -2.2850991 , -1.36550305, -5.44303134, -5.44303134,
       -1.17794119, -2.68110568, -5.77629399, -1.36550305, -2.68110568,
       -2.2850991 , -4.39406068, -1.33676149, -1.36550305, -4.39406068,
       -5.80374552, -1.1935721 , -1.69041032, -1.52706648, -1.35145256,
       -7.01386075, -4.06327457, -2.2850991 , -2.49354382, -4.39406068,
       -2.48465432, -4.39406068, -1.35145256, -2.47060383, -4.39406068,
       -7.01386075, -5.44303134, -1.33676149, -7.01386075, -4.39406068,
       -2.66705519, -4.39406068, -4.68874923, -7.01386075, -1.36550305,
       -1.41063233, -2.66705519, -2.49354382, -2.68110568, -1.17794119,
       -1.33676149, -2.66705519, -2.68110568, -5.44303134, -2.66705519,
       -1.39584443, -1.64276102, -1.36550305, -4.37313972, -4.39406068,
       -1.35145256, -1.39584443, -2.68110568, -1.17794119, -2.62909993,
       -7.01386075, -4.39406068, -1.49832493, -1.25181203, -2.65236412,
       -1.52706648, -4.09424671, -2.49354382, -1.33950462, -4.37313972,
       -7.01386075, -4.39406068, -4.39406068, -7.01386075, -2.66705519,
       -1.25181203, -1.33950462, -1.20828257, -2.49354382, -7.01386075,
       -4.39406068, -1.64276102, -2.50020677, -1.17794119, -4.39406068,
       -7.01386075, -1.17794119, -3.58839215, -1.52706648, -4.39406068,
       -6.90556761, -4.39406068, -2.47146521, -2.65236412, -1.35145256,
       -4.3016489 , -1.25181203, -1.17794119, -4.06327457, -1.36550305,
       -6.98577329, -2.66705519, -1.51301599, -4.39406068, -3.58839215,
       -4.39406068, -2.66705519, -1.36550305, -1.33950462, -4.39406068,
       -4.39406068, -1.36550305, -1.35145256, -4.39406068, -1.33676149,
       -7.01386075, -3.36592128, -1.38179394, -1.35145256, -2.49354382,
       -2.56814024, -5.44303134, -1.35145256, -1.17794119, -2.51816082,
       -4.33728987, -1.33950462, -1.33950462, -2.68157965, -1.36550305,
       -2.45591277, -1.25181203, -4.39406068, -2.68110568, -1.33950462,
       -2.51816082, -2.66705519, -2.68110568, -4.39406068, -4.39406068,
```

```
            -2.66705519, -2.68110568, -2.68110568, -4.33728987, -4.1790564 ,
            -1.38113396, -2.50346976, -7.01386075, -1.20828257, -1.17794119,
            -2.29709247, -2.66365457, -4.37313972, -1.25181203])
```

Evaluate Model Performance

```
from sklearn.metrics import mean_squared_error,r2_score
rf_train_mse = mean_squared_error(y_train, y_rf_train_pred)
rf_train_r2 = r2_score(y_train, y_rf_train_pred)

rf_test_mse = mean_squared_error(y_test, y_rf_test_pred)
rf_test_r2 = r2_score(y_test, y_rf_test_pred)


rf_results = pd.DataFrame(['Random Forest',rf_train_mse,rf_train_r2,rf_test_mse,rf_test_r2]).transpose()
rf_results.columns = ['Model','Training MSE','Training R2','Testing MSE','Testing R2 ']
rf_results
```

	Model	Training MSE	Training R2	Testing MSE	Testing R2
0	Random Forest	1.057186	0.759694	1.05209	0.758393

Start coding or generate with AI.

Model Compression

```
df_models = pd.concat([lr_results, rf_results], ignore_index=True , axis = 0)
df_models
```

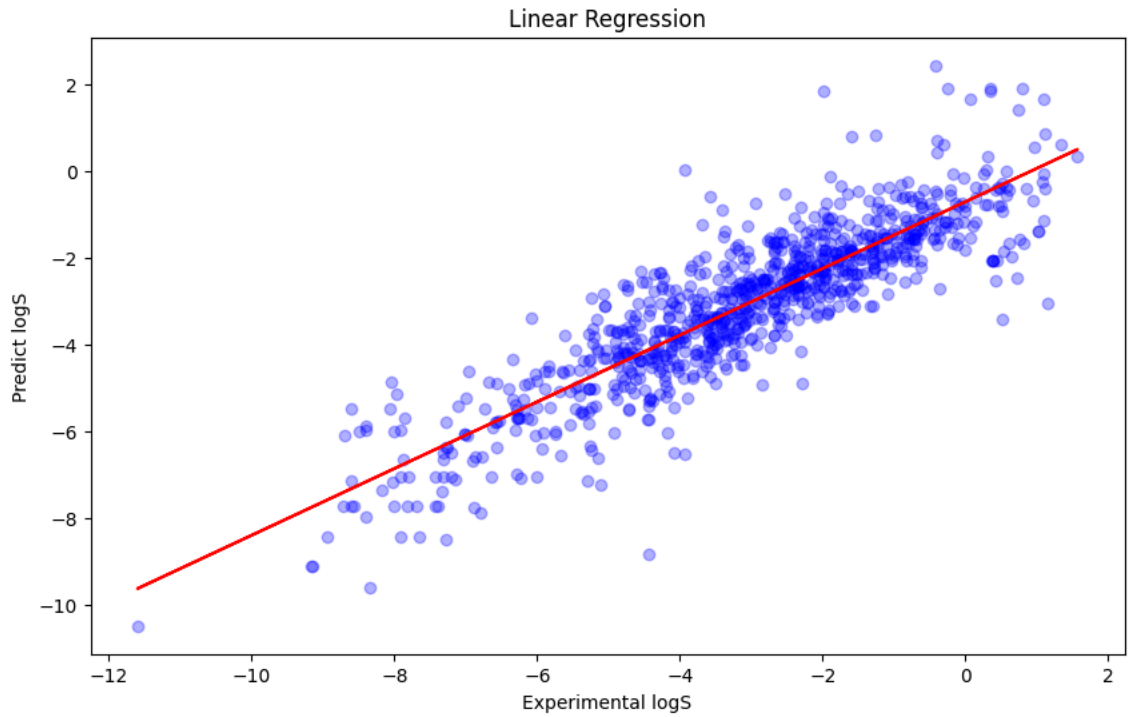
	Model	Training MSE	Training R2	Testing MSE	Testing R2
0	Linear Regression	1.013989	0.769513	0.999084	0.770565
1	Random Forest	1.057186	0.759694	1.05209	0.758393

Data Visualization of prediction Result

```
import matplotlib.pyplot as plt
import numpy as np

plt.figure(figsize=(10, 6))
plt.scatter(y_train, y_lr_train_pred, color='blue', alpha=0.3)
z = np.polyfit(y_train, y_lr_train_pred, 1)
p = np.poly1d(z)
plt.plot(y_train, p(y_train), color='red')
plt.xlabel('Experimental logS')
plt.ylabel('Predict logS')
plt.title('Linear Regression')

plt.show()
```



Start coding or generate with AI.