

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.svm import SVR
from sklearn.metrics import r2_score, mean_squared_error
from tqdm import tqdm

file_path = r'.xlsx'
data = pd.read_excel(file_path)

y = data.iloc[:, 0].values
X = data.iloc[:, 1:].values

kernel = 'rbf'  linear, poly, rbf, sigmoid
C = 3000
epsilon = 0.01 # Epsilon sigmoidin the epsilon-SVR model

num_random_states = 1000

results_df = pd.DataFrame(columns=['Random_State', 'Train_R2', 'Test_R2', 'Train_RMSE',
'Test_RMSE'])

for random_state in tqdm(range(1, num_random_states + 1)):
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=random_state)

    svr_model = SVR(kernel=kernel, C=C, epsilon=epsilon)

    svr_model.fit(X_train, y_train)

    y_train_pred = svr_model.predict(X_train)
    y_test_pred = svr_model.predict(X_test)

    train_r2 = r2_score(y_train, y_train_pred)
    test_r2 = r2_score(y_test, y_test_pred)
    train_rmse = np.sqrt(mean_squared_error(y_train, y_train_pred))
    test_rmse = np.sqrt(mean_squared_error(y_test, y_test_pred))
```

```
    results_df.loc[len(results_df)] = [random_state, train_r2, test_r2, train_rmse,
test_rmse]

output_path = r'.xlsx'
results_df.to_excel(output_path, index=False)

print(f'Results saved to {output_path}')
```