

Example 4: Dataset with more observations and features.

Data: [Advertising.csv](#)

Feature=3, Observations = 200

x.shape: (200,3) y.shape: (200,)

```
data = np.genfromtxt('Advertising.csv', delimiter=',')
data = data[1:,1:]
x = data[:,0:3]
y = data[:, -1]
```

Base Model with 1 linear regression model

Input

```
lr = LinearRegression(n_jobs=-1).fit(x,y)
```

Output

```
r2: 0.8972
MSE: 2.784
lr.coef_ = array([ 0.04576465,  0.18853002, -0.00103749 ])
lr.intercept_ = 2.938889369459403
Run Time: 0.002 seconds
```

SML Model with 2 linear regressions (1 run)

Input

```
left = LinearRegression(n_jobs=-1)
right = LinearRegression(n_jobs=-1)
result = sml(x,y, left, right)
```

Output:

```
r2: 0.973
MSE: 0.739
Run Time: 0.411 seconds

{'best_r2': 0.9727159532373776,
 'best_mse': 0.7390086990754077,
 'best_row': 89,
 'best_col': 0,
 'best_x': 135.2,
 'left': LinearRegression(copy_X=True, fit_intercept=True, n_jobs=-1, normalize=False),
 'right': LinearRegression(copy_X=True, fit_intercept=True, n_jobs=-1, normalize=False)}

result['left'].coef_ = array([0.06720927, 0.0996199 , 0.00717197])
result['left'].intercept_ = 3.3993734578784034
result['right'].coef_ = array([0.03328705, 0.26815454, 0.00196244])
result['right'].intercept_ = 3.510201394370972
```

Conclusion

Model increases accuracy from 0.89 to 0.97 but it takes 200 times more to achieve it. Still, for LinearRegression it takes less than half a second for a dataset that is 200 rows with 3 features.