

```
import numpy as np
from sklearn.preprocessing import StandardScaler as StandardScaler
from sklearn.decomposition import PCA as PCA
import pandas as pd

X_array = [0, 1, 2, 2, 3, 3, 4]
Y_array = [1, 1, 1, 3, 2, 3, 5]
raw_data = np.column_stack((X_array, Y_array))
```

(a) PCA after standardizing data

```
scaler = StandardScaler()
_ = scaler.fit(raw_data)
standardized_data = scaler.transform(raw_data)
```

```
standardized_pca = PCA(n_components=2)
_ = standardized_pca.fit(standardized_data)
print("First Two Components of Standardized PCA")
print(standardized_pca.components_)
```

```
First Two Components of Standardized PCA
[[ 0.70710678  0.70710678]
 [ 0.70710678 -0.70710678]]
```

```
standardized_pca_transformed_data = standardized_pca.fit_transform(standardized_data)
standardized_pca_dataframe = pd.DataFrame(data = standardized_pca_transformed_data
                                          , columns = ['PC 1', 'PC 2'])
standardized_pca_dataframe["Standardized X"] = standardized_data[:, 0]
standardized_pca_dataframe["Standardized Y"] = standardized_data[:, 1]
print(standardized_pca_dataframe)
```

	PC 1	PC 2	Standardized X	Standardized Y
0	-1.873053	-0.560268	-1.720618	-0.928279
1	-1.305278	0.007507	-0.917663	-0.928279
2	-0.737503	0.575282	-0.114708	-0.928279
3	0.283552	-0.445773	-0.114708	0.515711
4	0.340799	0.632529	0.688247	-0.206284
5	0.851327	0.122002	0.688247	0.515711
6	2.440157	-0.331278	1.491202	1.959700

```
print(np.matmul(standardized_data, standardized_pca.components_).round(2))
```

```
[[-1.87 -0.56]
 [-1.31  0.01]
 [-0.74  0.58]]
```

```
[ 0.28 -0.45]
[ 0.34  0.63]
[ 0.85  0.12]
[ 2.44 -0.33]]
```

(b) PCA without standardizing data

```
raw_data_pca = PCA(n_components=2)
_ = raw_data_pca.fit(raw_data)
print("First Two Components of Raw PCA")
print(raw_data_pca.components_)
```

```
First Two Components of Raw PCA
[[ 0.65908697  0.75206673]
 [ 0.75206673 -0.65908697]]
```

```
raw_pca_transformed_data = raw_data_pca.fit_transform(raw_data)
raw_pca_dataframe = pd.DataFrame(data = raw_pca_transformed_data
                                , columns = ['PC 1', 'PC 2'])
raw_pca_dataframe["Raw X"] = raw_data[:, 0]
raw_pca_dataframe["Raw Y"] = raw_data[:, 1]
print(raw_pca_dataframe)
```

	PC 1	PC 2	Raw X	Raw Y
0	-2.379272	-0.764174	0	1
1	-1.720185	-0.012107	1	1
2	-1.061098	0.739959	2	1
3	0.443035	-0.578215	2	3
4	0.350055	0.832939	3	2
5	1.102122	0.173852	3	3
6	3.265343	-0.392255	4	5

```
print(np.matmul(raw_data, raw_data_pca.components_).round(2))
```

```
[[ 0.75 -0.66]
 [ 1.41  0.09]
 [ 2.07  0.85]
 [ 3.57 -0.47]
 [ 3.48  0.94]
 [ 4.23  0.28]
 [ 6.4  -0.29]]
```

Since manually multiplying the Principal Components with the Raw Dataset does not yield the same result as the sklearn PCA model, we know that PCA is not scale invariant.

Therefore, it is best to standardize the data before the procedure.