Examine saved models

The purpose of this notebook is to examine the learned representations and accuracies of saved models.

Setup

Add NeuralODE code

```
train (generic function with 1 method)
```

Load model

```
loadModel (generic function with 1 method)
```

```
(Chain(
     layer_1 = Scale((100.)),            # 100 parameters
```

Load data

```
(ArrayAndFuncs([1.0, 1.0, 1.0, 1.0, 1.0,    more ,1.0], [Interpolate([   more], [   more
```

Add display method for `Interpolate`

```
function (itp::Interpolate)(t::Float64)
    i = searchsortedfirst(itp.locations, t)
    @inbounds itp.SET[i] - 1
end
```

Test model

Training data

```
1.0
```

Testing data

```
1.0
```

What is the testing SET number?

```
SET_num = 13
```

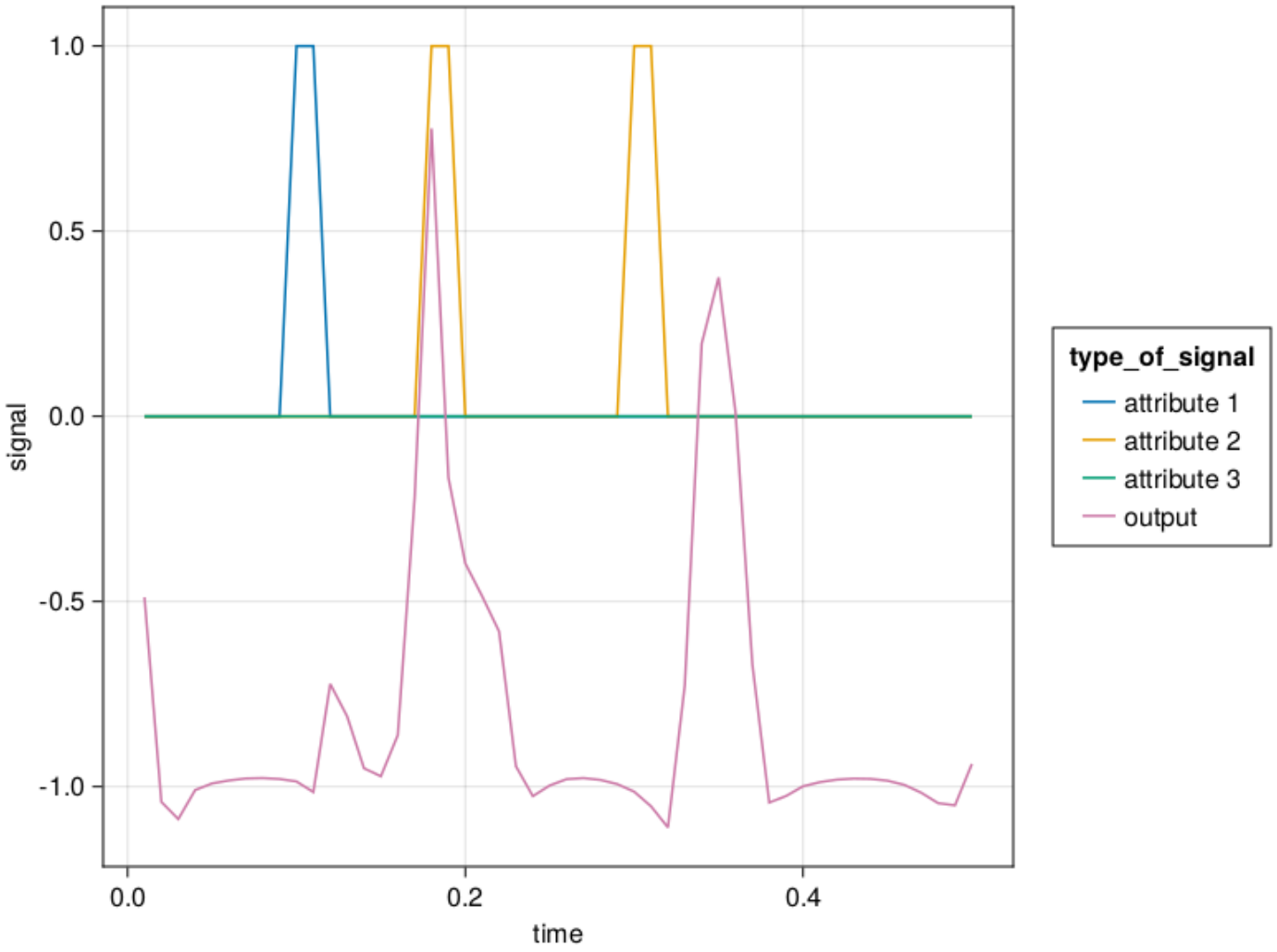
```
SET_num = 13
```

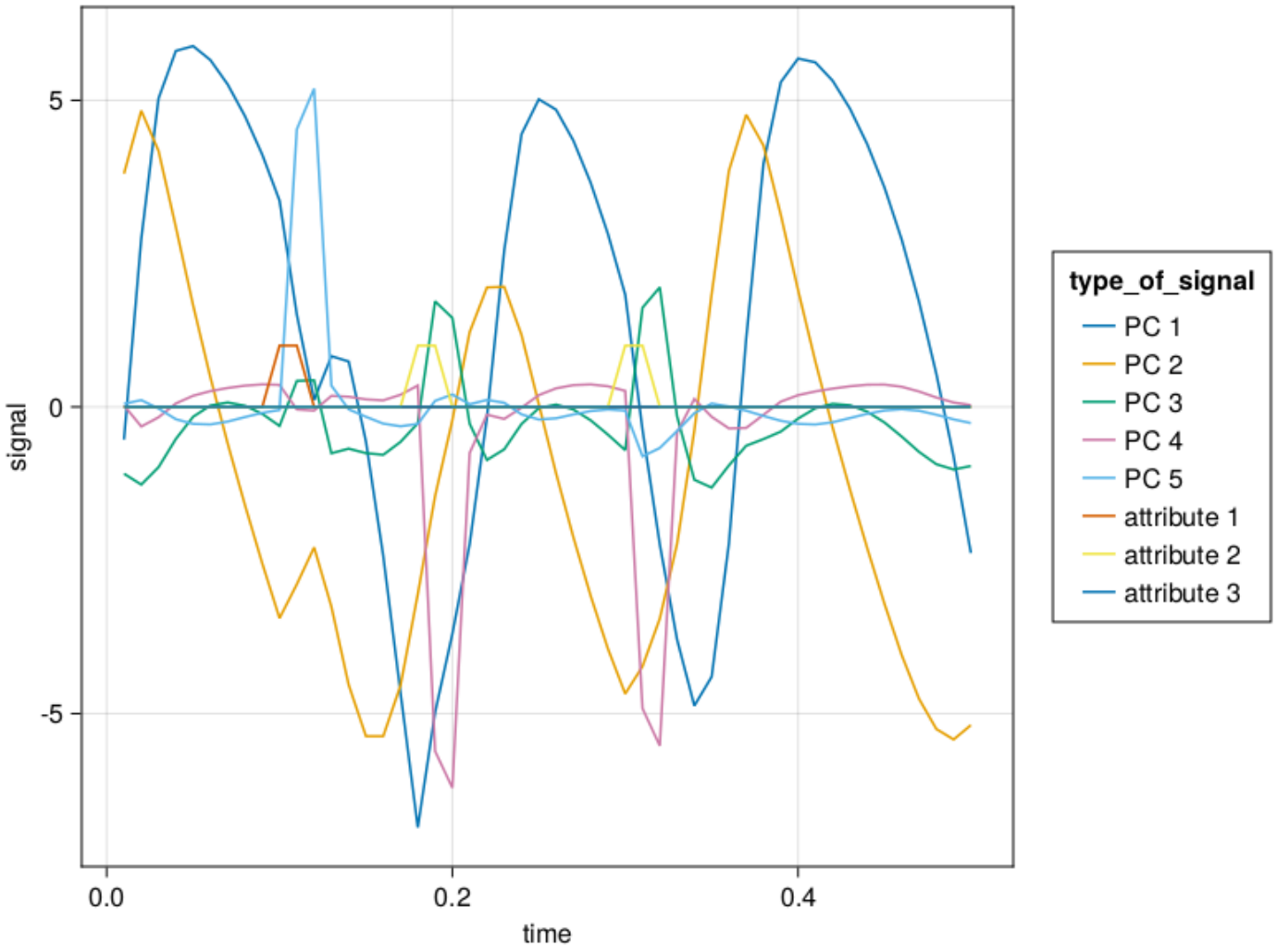
Visualize low-dimensional dynamics

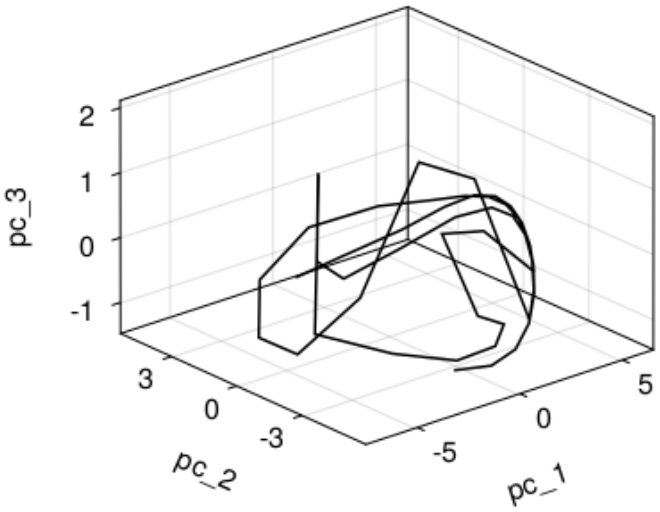
```
input_to_mat (generic function with 1 method)
```

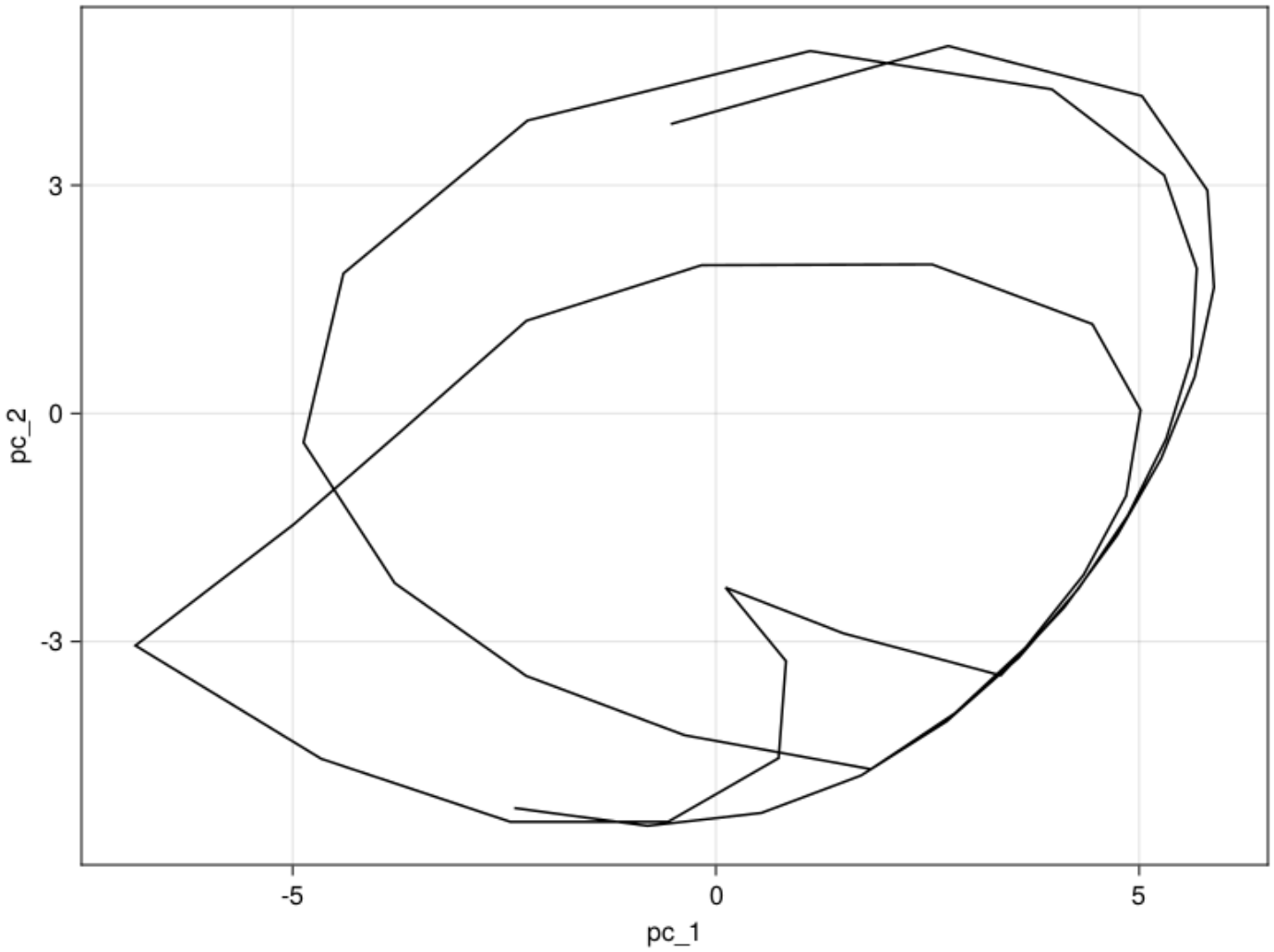
This is data.

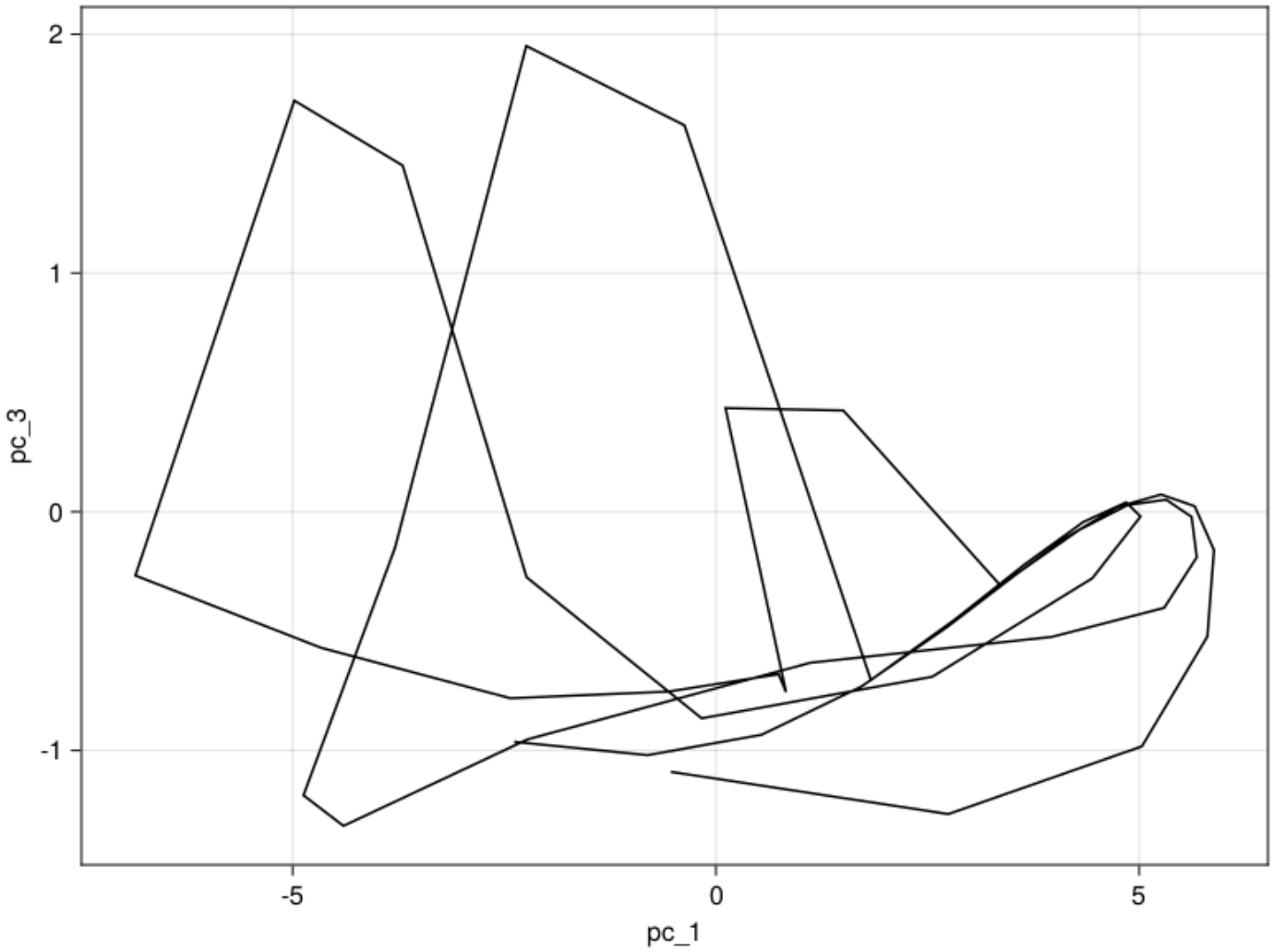
```
begin
    time_range = 0.01f0:0.01f0:0.50f0
    input = input_to_mat(reduce(vcat, testing_input_funcs[SET_num].
    (0.01:0.01:0.50)))
    (y_out_train, r_out_train), st_out = model(training_data[1], ps, st)
    (y_out_test, r_out_test), st_out = model(testing_data[1], ps, st)
    cat_r_out = reduce(hcat, r_out_train[:,:,i] for i in 1:size(r_out_train)[3])
    M = fit(PCA, cat_r_out;)
    pc_rates = predict(M, r_out_test[:,:,SET_num])
    y = y_out_test[1,:,SET_num]
    df_rates = DataFrame(
        time = time_range,
        out = y,
        pc_1= pc_rates[1,:],
        pc_2 = pc_rates[2,:],
        pc_3 = pc_rates[3,:],
        pc_4 = pc_rates[4,:],
        pc_5 = pc_rates[5,:],
    )
    md"This is data."
end
```

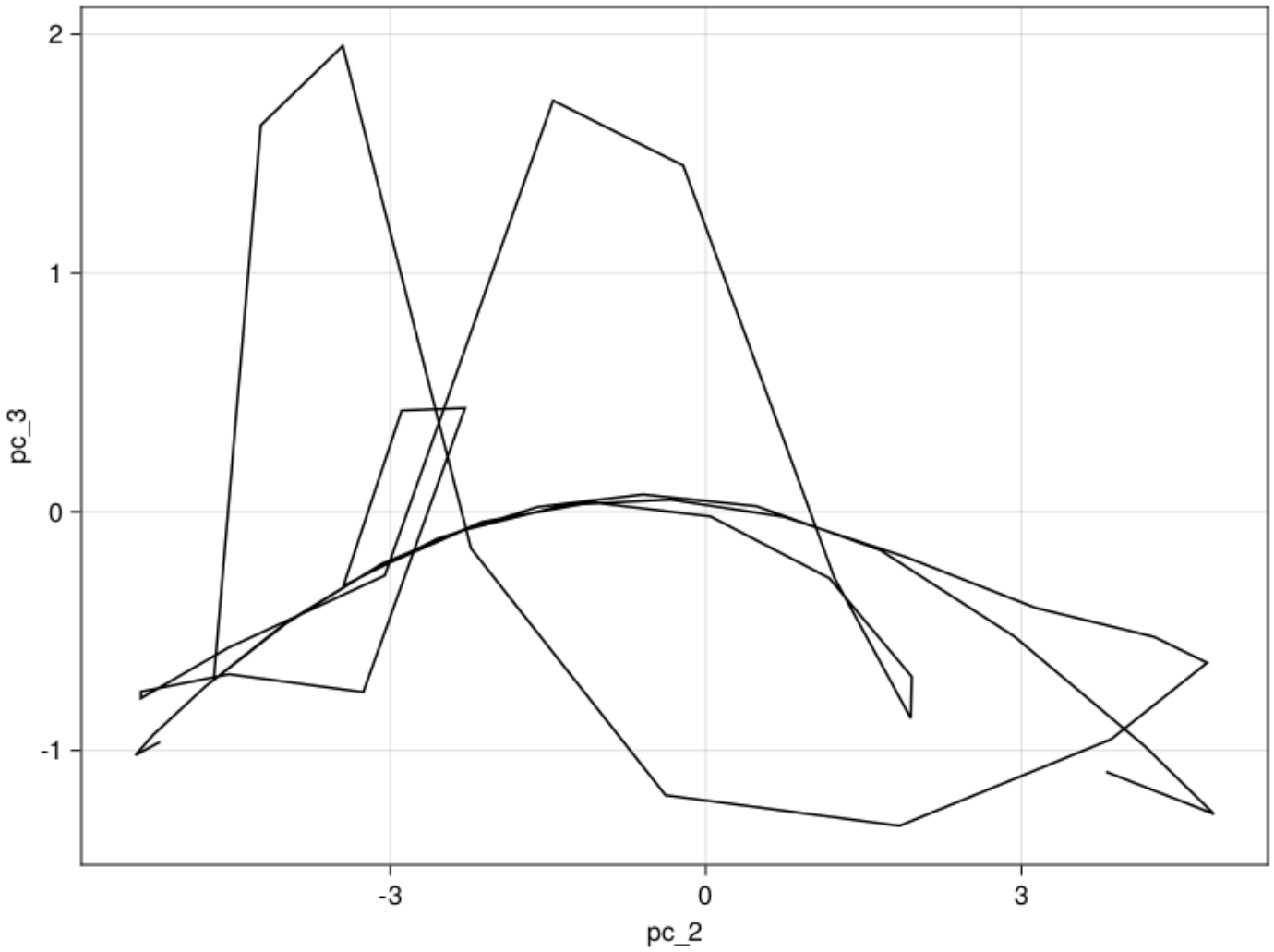

1.0
0.5
0.0
-0.5
-1.0
signal
0.0
0.2
0.4
time
type_of_signal
attribute 1
attribute 2
attribute 3
output


5
0
-5
signal
0.0
0.2
0.4
time
type_of_signal
PC 1
PC 2
PC 3
PC 4
PC 5
attribute 1
attribute 2
attribute 3


pc_3
2
1
0
-1
3
0
-3
pc_2
5
0
-5
pc_1


3
0
-3
pc_2
-5
0
5
pc_1


2
1
0
-1
pc_3
-5
0
5
pc_1


2
1
0
-1
pc_3
-3
0
3
pc_2