

# SHMTools Dataloader Demo

This notebook demonstrates the new simplified dataloader functions that make it easy to reproduce the conversion-plan.md examples.

## Three ways to load data:

1. **Quick setup:** `setup_notebook_environment()` - handles imports and path setup
2. **Example-specific:** `load_example_data('pca')` - preprocessed data for specific examples
3. **Direct loading:** `load_3story_data()` - raw data loading functions

## Method 1: Quick Setup (Recommended for new notebooks)

```
In [1]: # One-line setup for notebooks
from shmtools.utils.data_loading import setup_notebook_environment
nb = setup_notebook_environment()

# Now you have everything you need
np = nb['np']
plt = nb['plt']
load_3story_data = nb['load_3story_data']
check_data_availability = nb['check_data_availability']

print("Available functions:", list(nb.keys()))
```

```
/Users/eric/repo/shm/shmtools-python/shmtools/classification/nlpca.py:27: UserWarning: TensorFlow not available. NLPCA functions will not work. Install TensorFlow: pip install tensorflow
  warnings.warn(
Found shmtools at: /Users/eric/repo/shm/shmtools-python
Available functions: ['np', 'plt', 'Path', 'load_3story_data', 'load_sensor_diagnostic_data', 'load_cbm_data', 'load_active_sensing_data', 'load_modal_osp_data', 'check_data_availability', 'get_data_dir']
```

```
In [2]: # Check what data is available
check_data_availability()
```

Data directory: /Users/eric/repo/shm/shmtools-python/examples/data  
Directory exists: True

|                              |            |               |
|------------------------------|------------|---------------|
| data3SS.mat                  | ( 25.0 MB) | – ✓ Available |
| data_CBM.mat                 | ( 54.0 MB) | – ✓ Available |
| data_example_ActiveSense.mat | ( 32.0 MB) | – ✓ Available |
| dataSensorDiagnostic.mat     | ( 0.1 MB)  | – ✓ Available |
| data_OSPEXampleModal.mat     | ( 0.1 MB)  | – ✓ Available |

Available: 5/5 datasets (111.1 MB total)

## Method 2: Example-Specific Loading (Best for reproducing conversion-plan.md examples)

```
In [3]: from shmtools.utils.data_loading import load_example_data

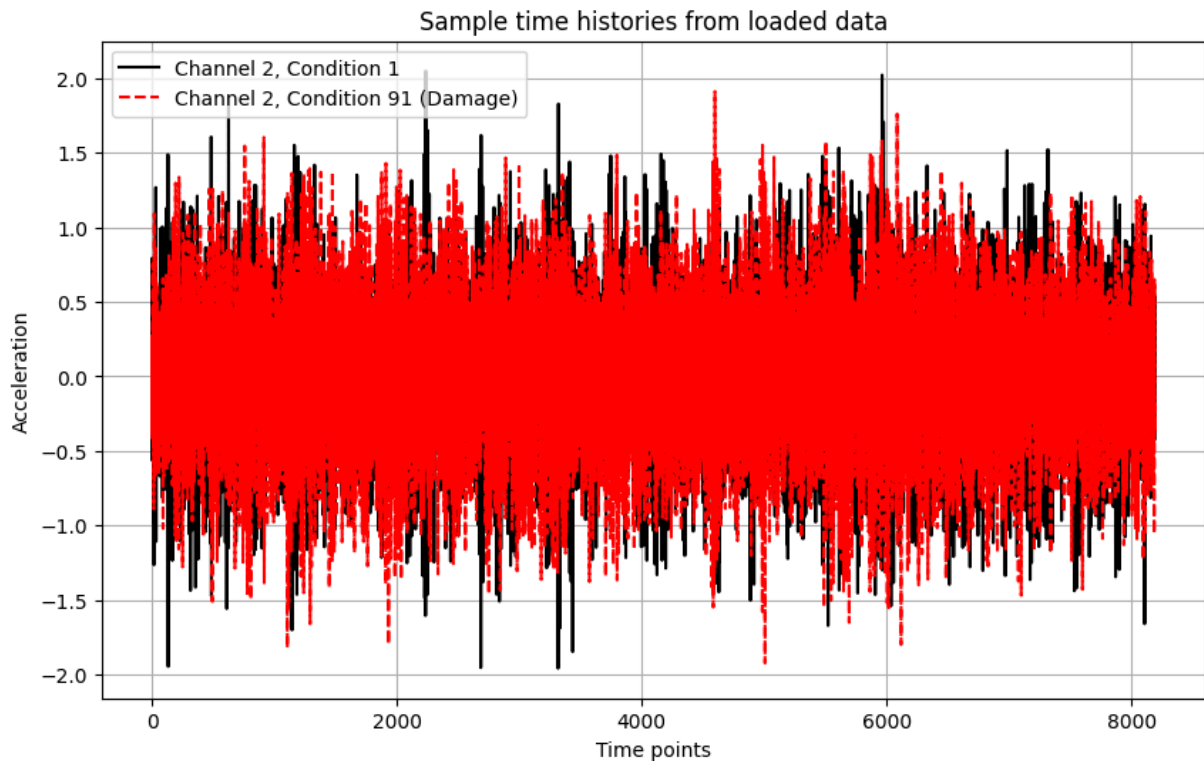
# Load preprocessed data for PCA/Mahalanobis/SVD examples
data = load_example_data('pca')

# Data is ready to use – channels 2–5 already extracted
signals = data['signals'] # Shape: (8192, 4, 170)
fs = data['fs']
damage_states = data['damage_states']
t, m, n = data['t'], data['m'], data['n'] # Same variables as in notebooks

print(f"Signals shape: {signals.shape}")
print(f"Channels: {data['channels']}")
print(f"Time points: {t}, Channels: {m}, Conditions: {n}")
```

Signals shape: (8192, 4, 170)  
Channels: ['Ch2', 'Ch3', 'Ch4', 'Ch5']  
Time points: 8192, Channels: 4, Conditions: 170

```
In [4]: # Quick visualization
plt.figure(figsize=(10, 6))
plt.plot(signals[:, 0, 0], 'k-', label='Channel 2, Condition 1')
plt.plot(signals[:, 0, 90], 'r--', label='Channel 2, Condition 91 (Damage)')
plt.title('Sample time histories from loaded data')
plt.xlabel('Time points')
plt.ylabel('Acceleration')
plt.legend()
plt.grid(True)
plt.show()
```



## Method 3: Direct Loading (For custom data processing)

```
In [5]: # Direct access to raw data loading functions
data_dict = load_3story_data()

# Full dataset with all channels
dataset = data_dict['dataset'] # Shape: (8192, 5, 170)
channels = data_dict['channels']

print(f"Full dataset shape: {dataset.shape}")
print(f"All channels: {channels}")

# Manual extraction of channels 2-5 (like in original notebooks)
data = dataset[:, 1:5, :]
print(f"Extracted channels 2-5 shape: {data.shape}")
```

```
Full dataset shape: (8192, 5, 170)
All channels: ['Force', 'Ch2', 'Ch3', 'Ch4', 'Ch5']
Extracted channels 2-5 shape: (8192, 4, 170)
```

## Loading Other Dataset Types

```
In [6]: # Load different dataset types
cbm_data = load_example_data('cbm')
sensor_data = load_example_data('sensor_diagnostic')
active_data = load_example_data('active_sensing')

print(f"CBM data keys: {list(cbm_data.keys())}")
```

```
print(f"Sensor diagnostic keys: {list(sensor_data.keys())}")
print(f"Active sensing keys: {list(active_data.keys())}")
```

CBM data keys: ['\_\_header\_\_', '\_\_version\_\_', '\_\_globals\_\_', 'Fs', 'damageStates', 'dataset', 'stateList']

Sensor diagnostic keys: ['\_\_header\_\_', '\_\_version\_\_', '\_\_globals\_\_', 'sd\_ex\_broken', 'sd\_ex']

Active sensing keys: ['\_\_header\_\_', '\_\_version\_\_', '\_\_globals\_\_', 'waveformBase', 'waveformTest', 'borderStruct', 'sampleRate', 'sensorLayout', 'pairList', 'actuationWaveform', 'damageLocation']

## Summary

These dataloader functions make it much easier to:

1. **Get started quickly** with `setup_notebook_environment()`
2. **Reproduce examples** with `load_example_data('pca')` etc.
3. **Handle different execution contexts** automatically (different working directories)
4. **Reduce repetitive code** in notebook imports

**For conversion-plan.md examples**, use Method 2 with the appropriate example type:

- Phase 1-3: `load_example_data('pca')`,  
`load_example_data('mahalanobis')`, `load_example_data('svd')`
- Phase 4-5: `load_example_data('factor_analysis')`,  
`load_example_data('nlpca')`
- Phase 6: `load_example_data('ar_model_order')`