CausalEngine: Making Machine Learning Smarter

(From Correlation to Causation)

(Heyang Gong)

August 13, 2025

? (What are we talking about today?)

(Limitations of Traditional ML)

? (What does traditional ML do?)

- (Learn correlations in data)
- : (Like: see clouds, predict rain)
- (But don't know WHY it rains)

Don't know what happens inside

: (Problems arise:)

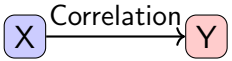
- (Fails with noisy data)
- "" (Individual differences = "noise")
- "", "" (Tells WHAT, not WHY)

(Core Problem)

, ! (Real-world data is always noisy, traditional methods can't handle it!)

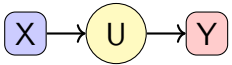
? (What kind of solution do we need?)

Traditional Method



- Simple & Direct
- Easy to overfit
- Noise sensitive

Causal Method



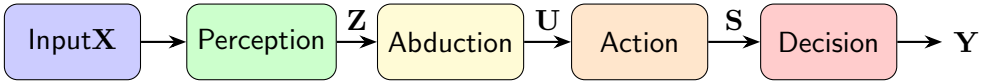
Individual

- Understand mechanism
- Noise robust
- Personalized prediction

(Key Insight)

U, , ! (Each individual has unique feature U - this is not noise, but useful information!)

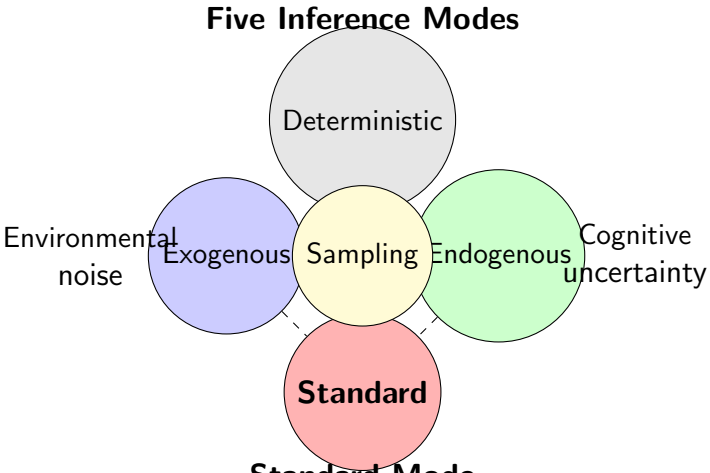
(CausalEngine's Four Steps)



(Core Idea)

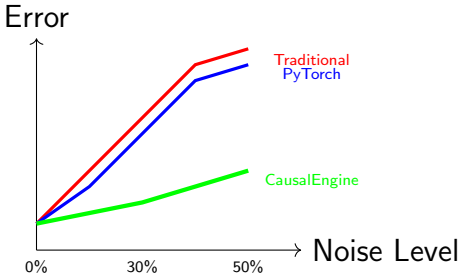
: , , ! (Like a doctor: observe symptoms, infer causes, then treat!)

(Five Inference Modes)



: (Noise Robustness Test: Regression)

, ? (How does performance change with noise?)



30%: (Performance at 30% label noise:)

- Traditional MLP: Error 47.60
- PyTorch: Error 45.32
- **CausalEngine: Error 11.41**

(Amazing improvement)

76%! (76% improvement!)

(Key Finding)

, ! (More noise = more advantage for CausalEngine!)

(Installation and Basic Usage)

(Installation is simple)

```
pip install causal-sklearn
```

(Basic Usage Example)

```
from causal_sklearn import MLPCausalRegressor
from sklearn.datasets import make_regression
from sklearn.model_selection import train_test_split

# Generate data
X, y = make_regression(n_samples=1000, n_features=10, noise
    =20)
X_train, X_test, y_train, y_test = train_test_split(X, y)
```

? (When to use CausalEngine?)

: (Particularly suitable:)

- ✓ (Noisy data)
- ✓ (Individual differences matter)
- ✓ (Medical diagnosis)
- ✓ (Financial risk)
- ✓ (Recommendation systems)
- ✓ (Anomaly detection)

: (Limited advantage:)

- × (Very clean data)
- × (Pure image classification)
- × (No interpretability needed)
- × (Very limited computation)

(Rule of thumb)

, (When data quality is uncertain or robustness is needed, CausalEngine is ideal)

(Core Contributions)

1. **(New ML Paradigm)**
   - (From correlation to causation)
   - , (Individual differences as features, not noise)
2. **(Practical Implementation)**
   - scikit-learn API (Full scikit-learn compatibility)
   - (Efficient analytical computation)
   - (Easy integration)
3. **(Exceptional Robustness)**
   - (Superior performance in noisy environments)
   - (Suitable for real-world messy data)

(One-sentence Summary)

"""", (CausalEngine brings new possibilities to ML by understanding WHY, not just WHAT)

! (Thank You!)

(Questions & Discussion)

(Making ML Smarter)