Spring '24

Capstone in Cyber

Security & Policy

Module [2] | Topic 02[1]

Machine Learning for Cyber

Anton Sobolev

ML: Typical Setting

Task: Classification

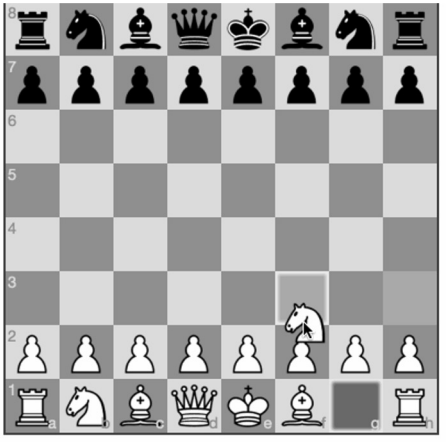
- **Pattern Inference ⇝ Prediction**
  - Prediction ≠ Forecast
  - Where does Q come from?

[Can Computers Solve Human Tasks?] ⇝ [how unique is human / life in general?]

- **Early Debate: CS'sts VS ST'sts**
  - Is human thinking reproducible?
  - Focus Inside the Box / Outside the Box
  - [Logic] Algorithms VS [Expectations] Pr() Models
  - Turing Test [gpt4 - strong AI?]

- **CS' Algorithm Approach**

```
[b/w] ⇝ [w] ⇝ [a2:a3] ⇝ [a7:a6]
              [....⇜]   [.....]
[game theory analogy]   [h2:h3]   [h7:h6] ⇝ result_i
[dominant approach
 most of the history]
[chess gamedev]
```

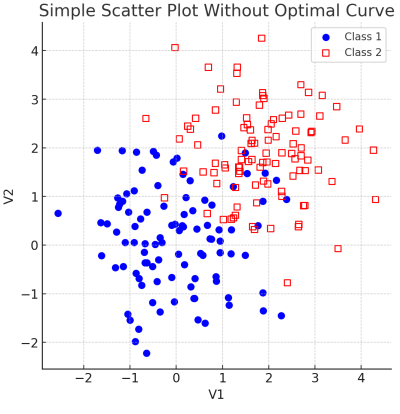


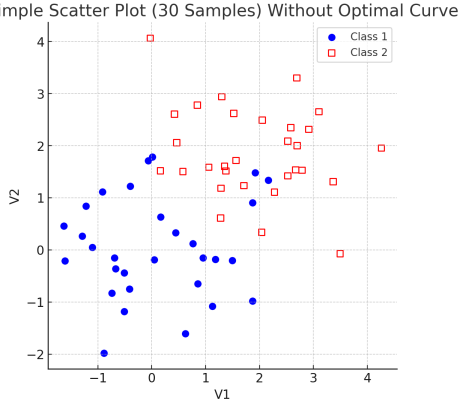
ML: Typical Setting

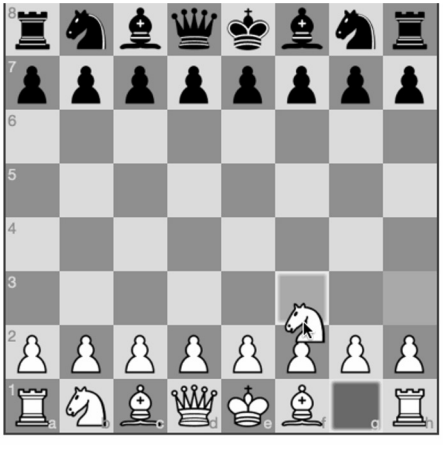
Task: Classification

- Stats' Approach
  - **Pr() Model**: Expected move~observed data
  - Prediction ≠ Forecast

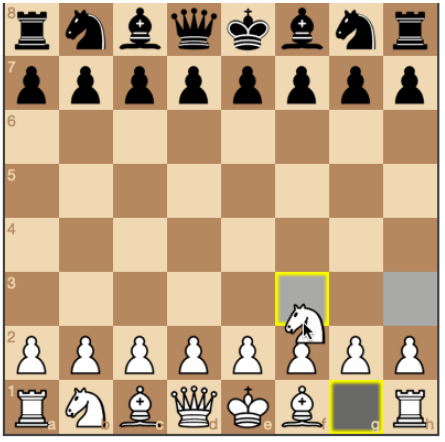
- **Pattern Inference ⇝ Prediction**
  - Prediction ≠ Forecast
  - Where does Q come from?







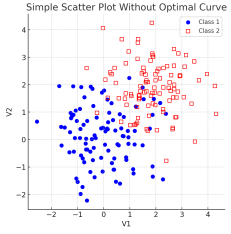
ML: Typical Task



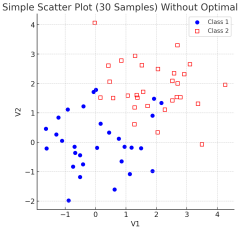
- **Algorithm**: [1] if [] ⇝ [] result$_i$

  [game theory analogy]
  [dominant approach most of the history]
  [chess gamedev]

- **Pr() Model**: Expected move [value] ~ observed games

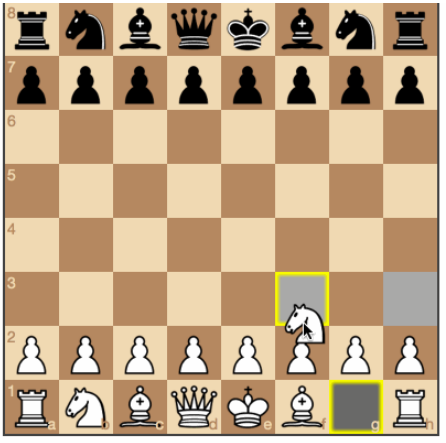


⇝

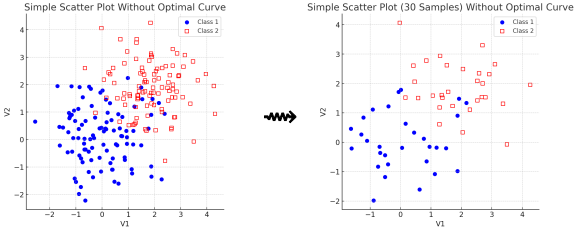


⇝ next step

ML: Typical Task



- **Pr() Model**: Expected move [value] ~ observed games



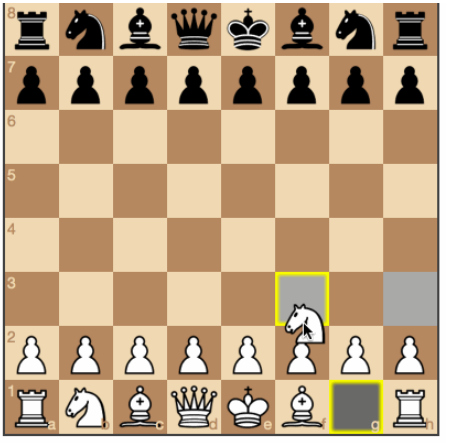
[queen with 5 choices example]

- **Model Structure**:

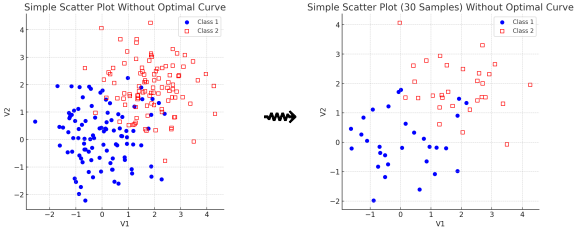
[Probability == Filtering]

	Black 1	White 1	Black 2	White 2	Black 3	White 3	Black 4	White 4	Black 5	White 5	White_1to_e6	White_1to_e8
1	e4	e5	f4	f5	g4	g5	h4	h5	a4	a5	0.59	0.80
2	d5	d4	c6	c7	b6	b7	a6	a7	h5	h6	0.46	0.65
3	c4	c5	b4	b5	a4	a5	h5	h6	g4	g5	0.44	0.71
4	e6	e7	f6	f7	g6	g7	h6	h7	f4	f5	0.43	0.69
5	g5	g6	h5	h6	f5	f6	e5	e6	d4	d5	0.42	0.62

ML: Typical Task



- **Pr() Model**: Expected move [value] ~ observed games

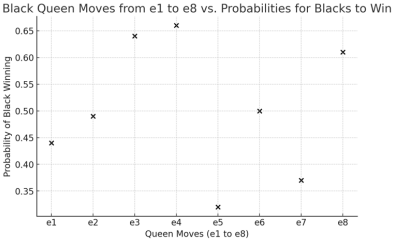


[queen with 5 choices example]

- **Model Structure**: $Y_{1\_to\_e6} \sim {}_aB1 + {}_aB2 + {}_aB3 + .. + {}_aW5$

[Probability == Filtering]

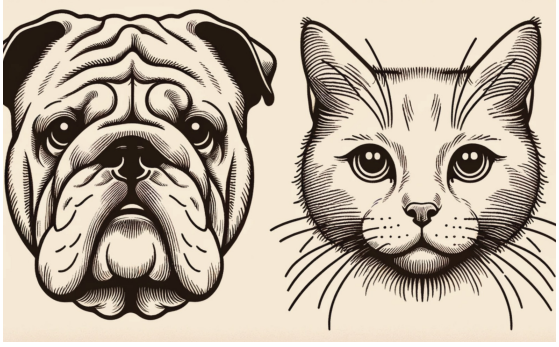
	Black 1	White 1	Black 2	White 2	Black 3	White 3	Black 4	White 4	Black 5	White 5	White_1to_e6	White_1to_e8
1	e4	e5	f4	f5	g4	g5	h4	h5	a4	a5	0.59	0.80
2	d5	d4	c6	c7	b6	b7	a6	a7	h5	h6	0.46	0.65
3	c4	c5	b4	b5	a4	a5	h5	h6	g4	g5	0.44	0.71
4	e6	e7	f6	f7	g6	g7	h6	h7	f4	f5	0.43	0.69
5	g5	g6	h5	h6	f5	f6	e5	e6	d4	d5	0.42	0.62

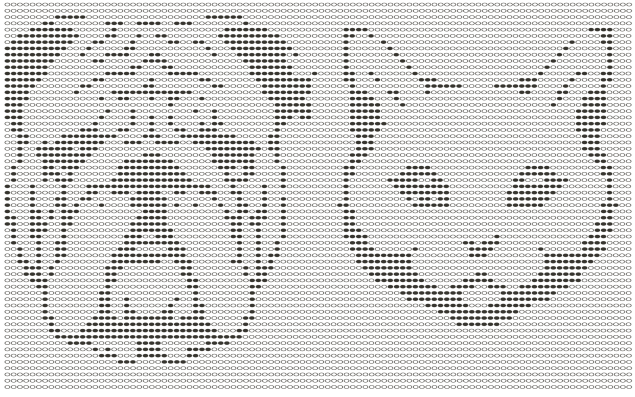


ML: 1st round debates

Where does Q come from?

Is it possible to reproduce human thinking?





Algorithmic Approach

[draughts ≠ chess]
[efficient ~ computationally]
[not efficient ~ human capital exp]

Statistical Approach

[not that task!]
[efficient ~ an undergrad can do!]
[not efficient ~ computationally!]
[data collection is biased]

Recap

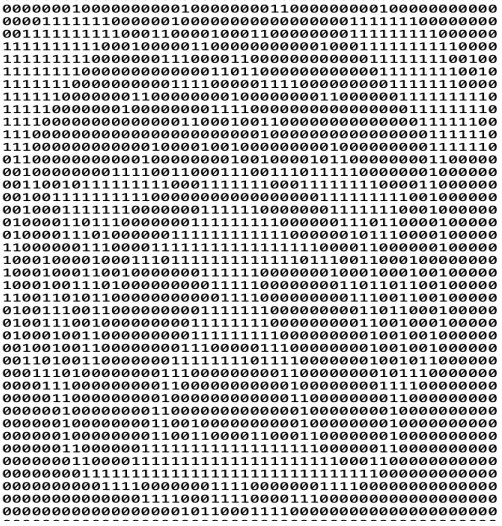
- Automated Data Collection
- **ML: Typical Task** [Classification]
  - Initial Motivation: AI & Turing Test
  - Algorithm ~><~ Statistical Model ⇝ **ML**
  - Cats VS Dogs [How?]
  - [3] Game-changers [Revolutions]

- **Unsupervised Learning**
  - Efficient Information Aggregation
    ⇝ Dimensionality Reduction

  - Clustering
  - Anomaly Detection

ML: Revolutions

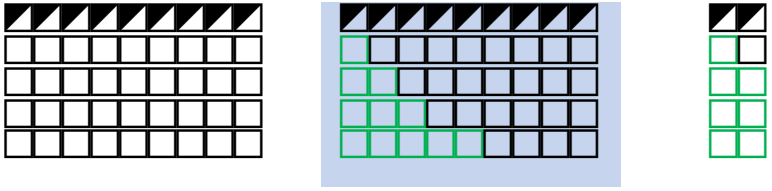
Digital Electronics
photo / video



- **Revolution [1]: Grid Revolution**
  Everything can be presented
  as table or vector

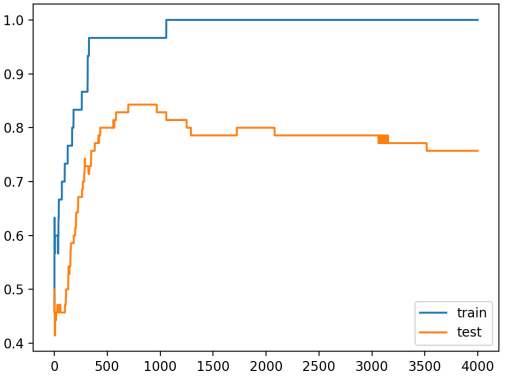
Media Storage / File Storage

- **Revolution [2]: Dimension Reduction [2]**
  Any table with useful information
  can be reduced



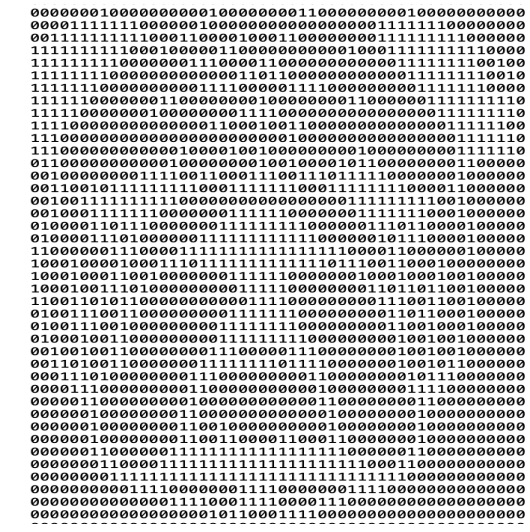
↝ Feature Extraction

- **Revolution [3]: Iterations and Batch**
  Piece-by-Piece Analysis ↝ Same resul

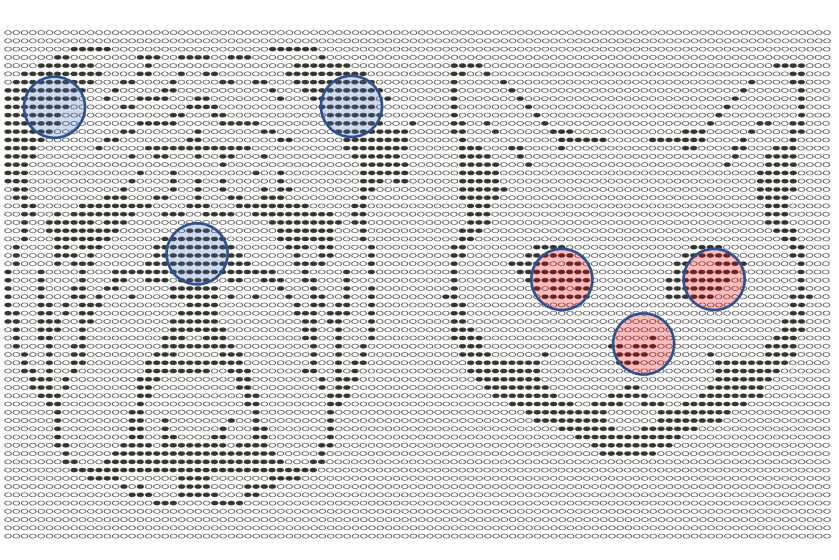


Naïve ML: Setting

Subheader ##
Subheader ##

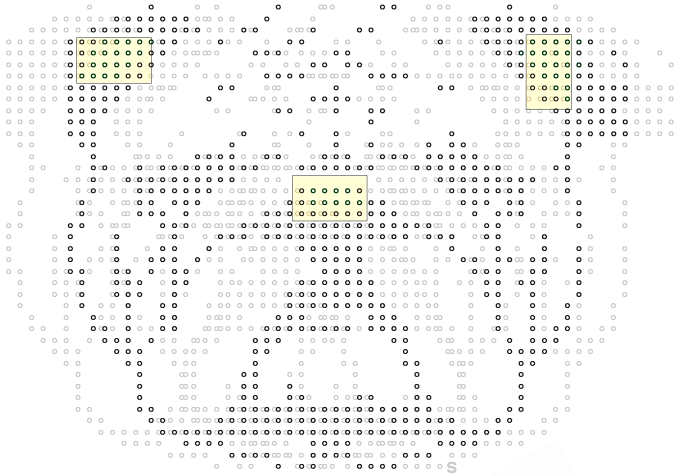


- What is learning in this case?



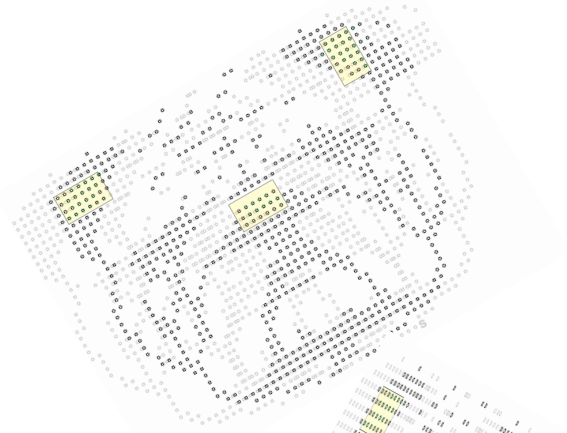
Naïve ML: Issues

- **Location Dependency**
  - Location = Feature
    [not what we want]
  - Non-Linear Visual Distortion



- **Cost**
  - Storage ◻ Memory [200k²]
  - Processing Speed

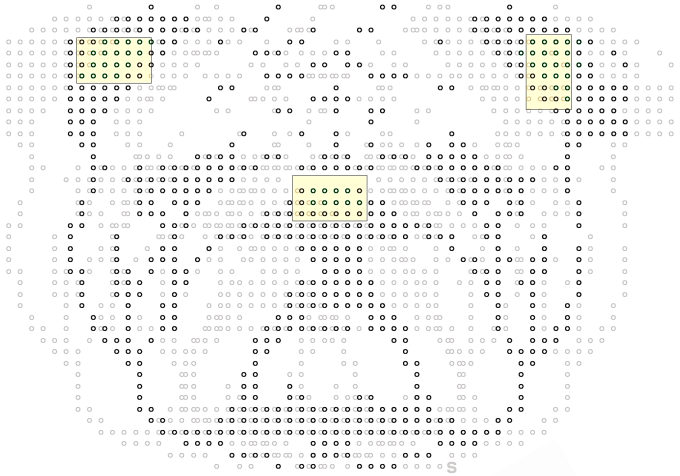
- **Inefficiency**
  - Only part of input data Is informative



- **Solution: Dimension Reduction**
  - Principal Components
  - Eigen Value Decomposition
  - Numeric Approximations

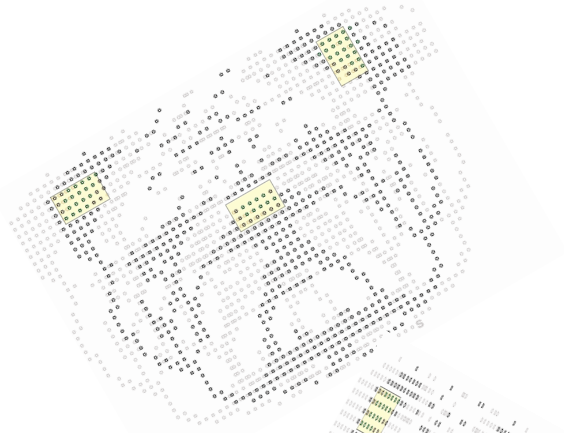
Naïve ML: Issues

- **Location Dependency**
  - Location = Feature
    [not what we want]
  - Non-Linear Visual Distortion

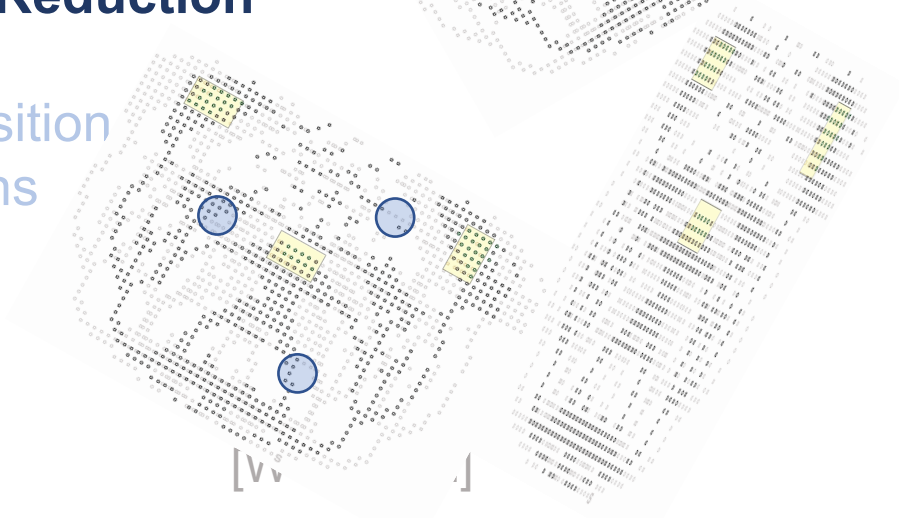


- **Cost**
  - Storage ◻ Memory [200k²]
  - Processing Speed

- **Inefficiency**
  - Only part of input data Is informative



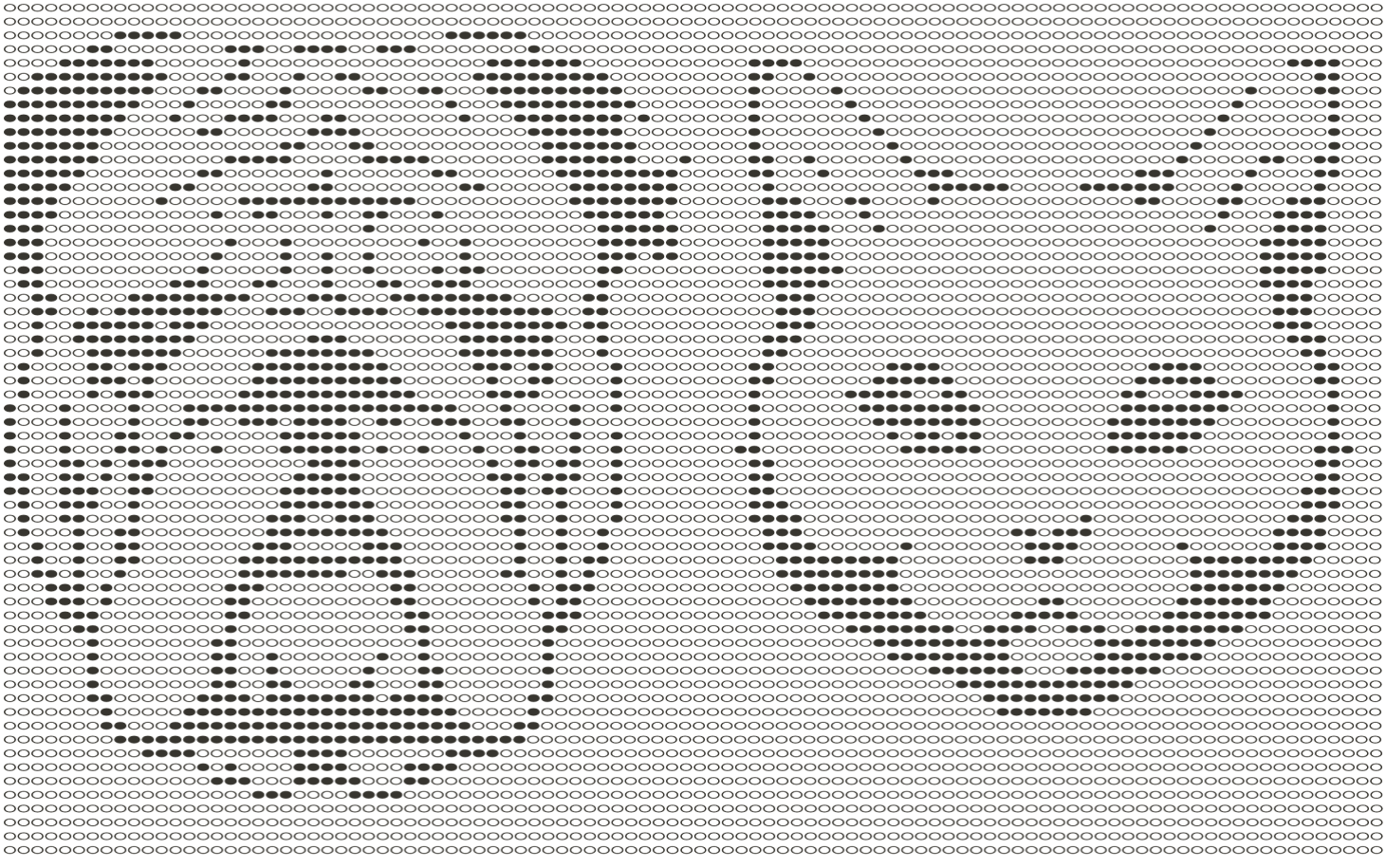
- **Solution: Dimension Reduction**
  - Principal Components
  - Eigen Value Decomposition
  - Numeric Approximations



Reminder

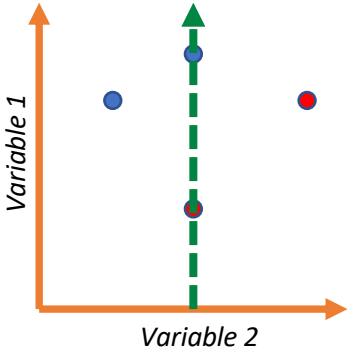
Supervised VS Unsupervised

What is different?



PCA: Dimension Reduction with Principal Components

Task: Use only one column (variable) to make best possible prediction of the type (color) of each point

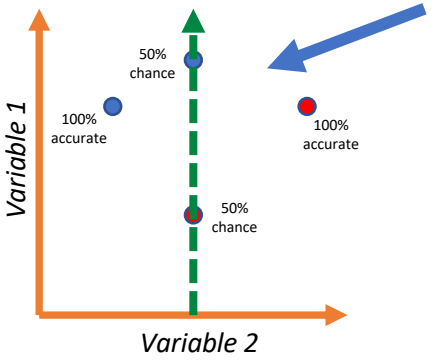


We use the variable to split the data according to some specific value (e.g., V2 = 0.4)

We choose V2=x to maximize the separation between red and blue dots

PCA: Dimension Reduction with Principal Components

Task: Use only one column (variable) to make best possible prediction of the type (color) of each point



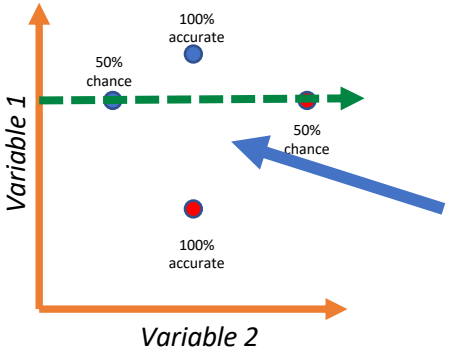
*If we use **V2**: on average, we predict the color of each dot with 50% accuracy*

We use the variable to split the data according to some specific value (e.g., V2 = 0.4)

We choose V2=x to maximize the separation between red and blue dots

PCA: Dimension Reduction with Principal Components

Task: Use only one column (variable) to make best possible prediction of the type (color) of each point



*If we use **V1**: on average, we also predict the color of each dot with 50% accuracy*

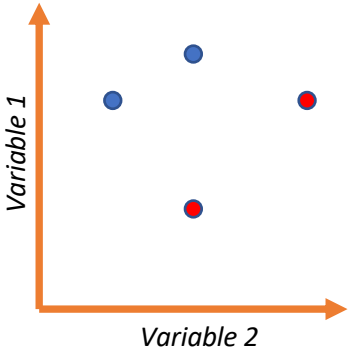
We use the variable to split the data according to some specific value (e.g., V2 = 0.4)

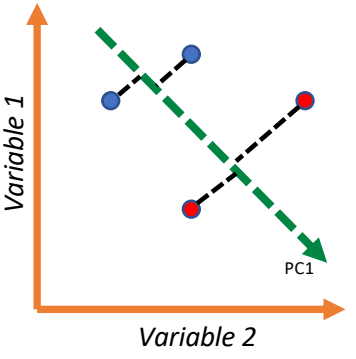
We choose V2=x to maximize the separation between red and blue dots

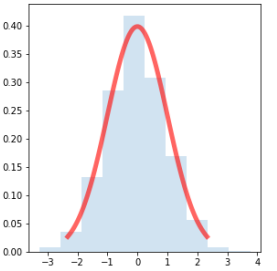
PCA: Dimension Reduction with Principal Components

Task: Use only one column (variable) to make best possible prediction of the type (color) of each point

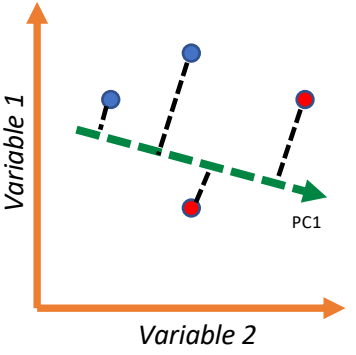
What if we rotate axes?

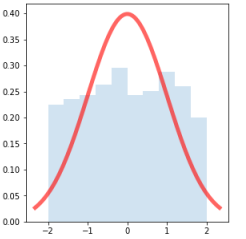






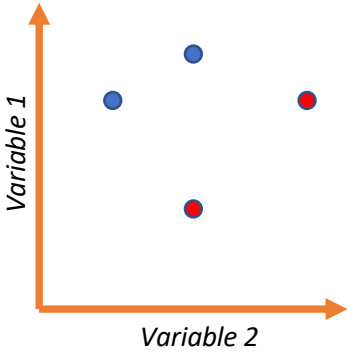
Best rotated axis should have the largest possible distribution of values

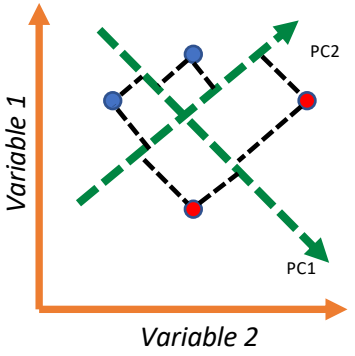


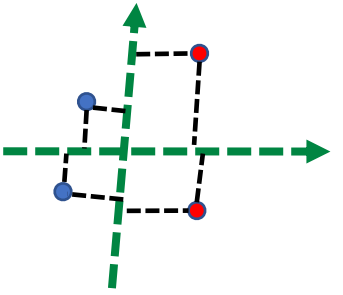


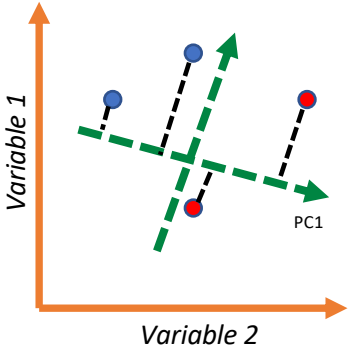
PCA: Dimension Reduction with Principal Components

Task: Use only one column (variable) to make best possible prediction of the type (color) of each point



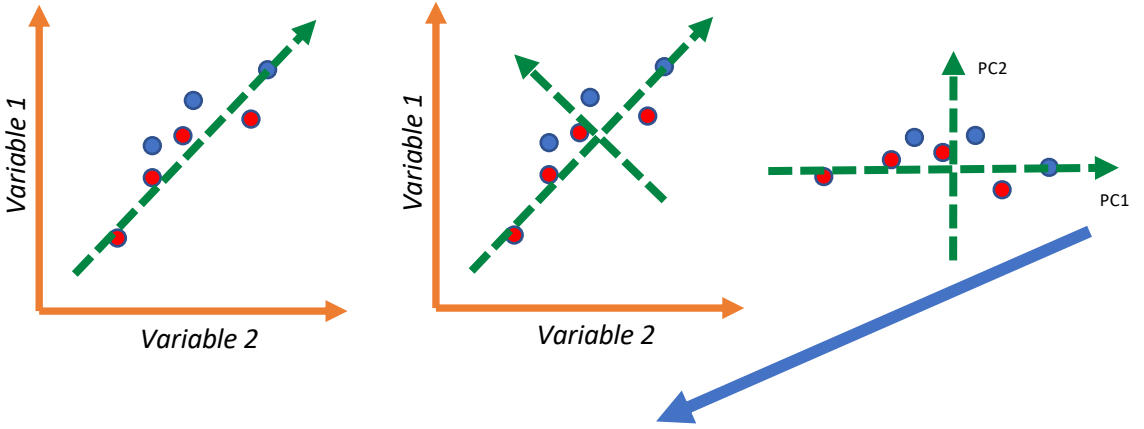






PCA: Dimension Reduction with Principal Components

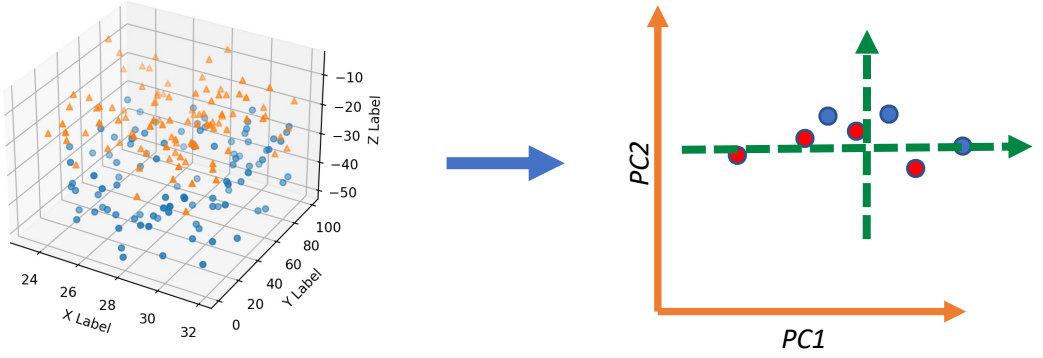
PCA also solves the problem
of 'duplicates with noise'



The first PC contains all information
shared by Variable 1 and Variable 2

PCA: Dimension Reduction with Principal Components

We can easily extend PCA-approach to multiple dimensions

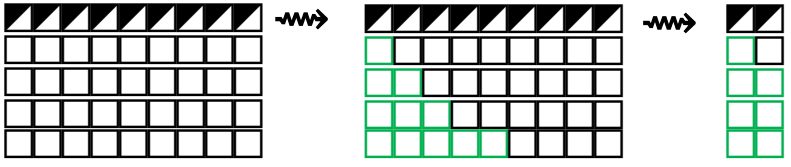


PCA: Issues it solves

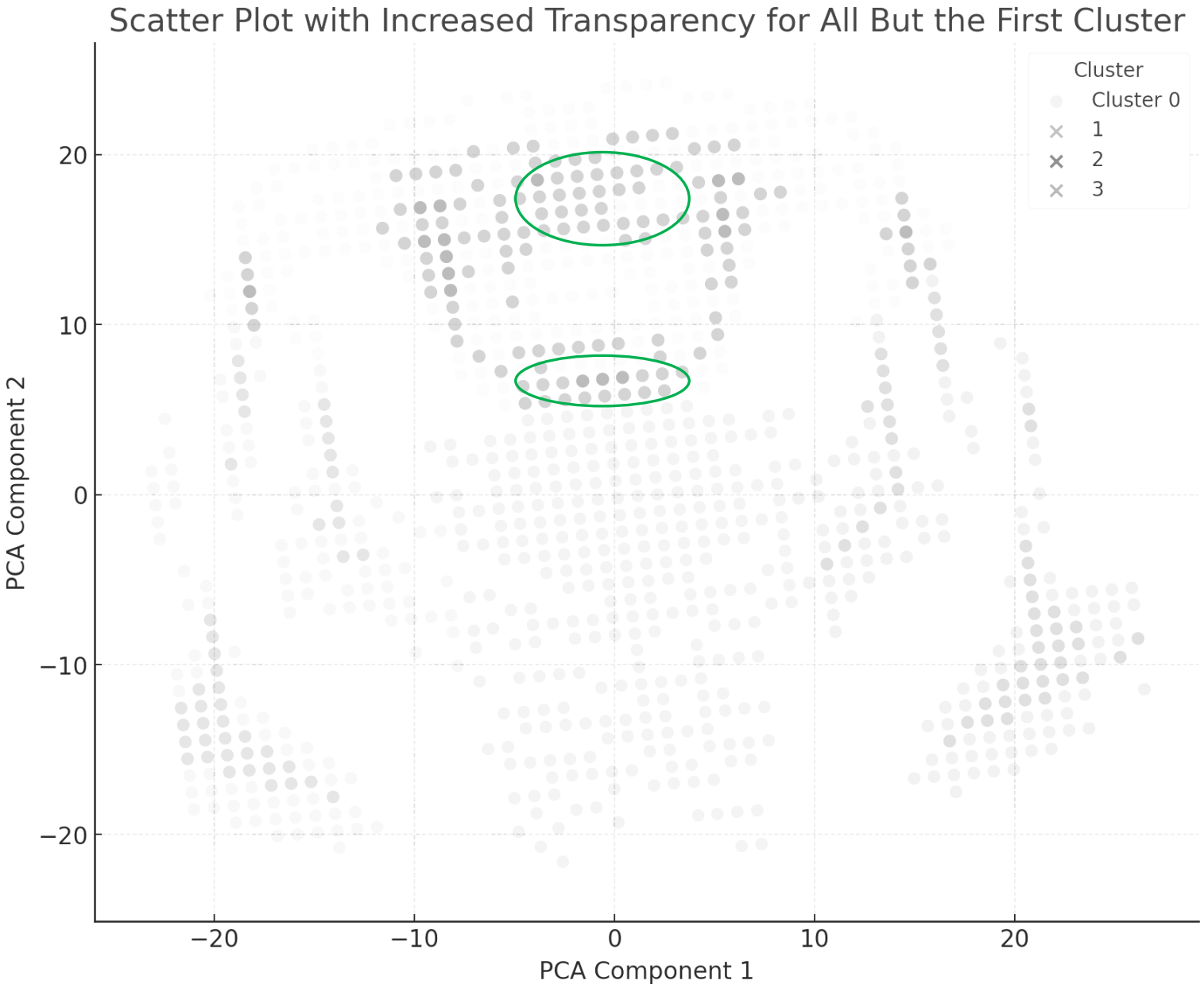
Task: Classification

- **Rotation**
  - [Decision-Tree works always now]

- **Memory & Computational Capacity**
  - [Decision-Tree works always now]



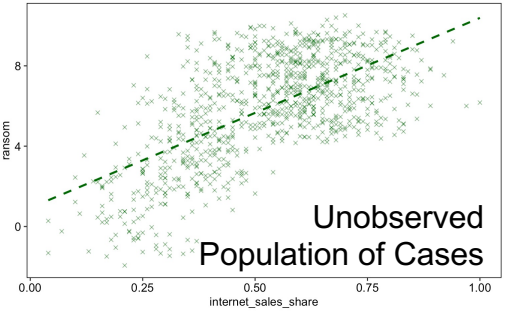
- **Overuse of Information [Data Duplication] [Wrong Clustering]**


Scatter Plot with Increased Transparency for All But the First Cluster
Cluster
Cluster 0
1
2
3
20
10
0
−10
−20
PCA Component 2
−20
−10
0
10
20
PCA Component 1

Machine Learning

Goals

- ❑ Pattern Inference ⇝ Prediction
  - ❑ Prediction ≠ Forecast



- ❑ Clustering (groups based on similarity)
- ❑ Anomaly Detection (fraudulent transaction)
- ❑ Regression (phishing attacks)
- ❑ Classification (cats VS dogs)
- ❑ Advanced Simulation (deepfakes, play chess, gpt-chat)

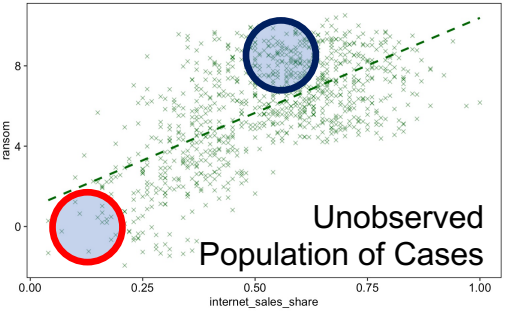
Machine Learning

Goals

- ❑ Pattern Inference ⇝ Prediction
  - ❑ Prediction ≠ Forecast

- ❑ Clustering (groups based on similarity)
- ❑ Anomaly Detection (fraudulent transaction)
- ❑ Regression (phishing attacks)
- ❑ Classification (cats VS dogs)
- ❑ Advanced Simulation (deepfakes, play chess, gpt-chat)

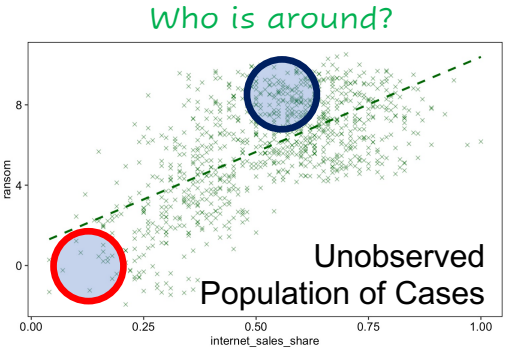
Who is around?



Unsupervised Learning

Machine Learning Workflow

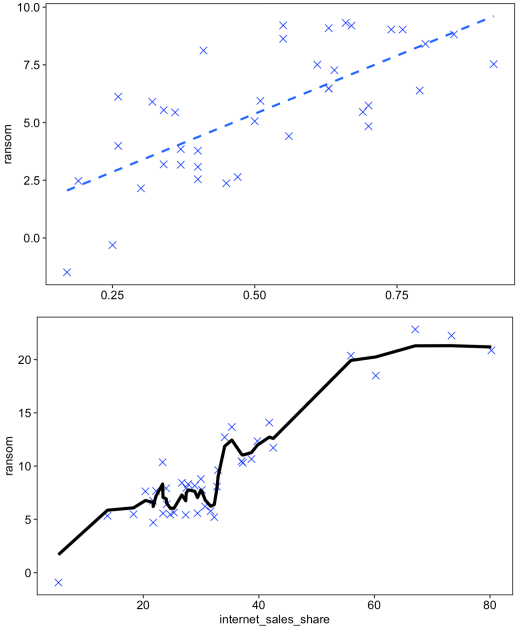
1. Raw Data Sampling
2. Feature Extraction
3. Data Split
4. Train Model
5. Evaluate Performance
6. Make Predictions!



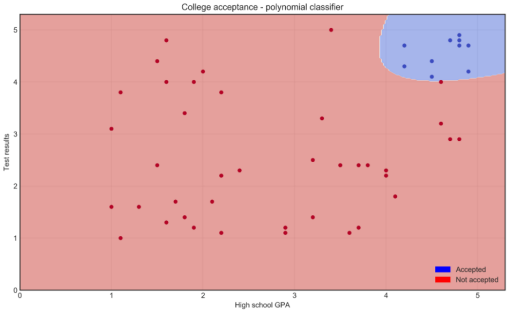
Unsupervised Learning

What is a ML-model?

Regression



Anomaly Detection



Clustering

Sampling Techniques

Over-sampling, under-sampling, synthetic control