

Capstone in Cyber Spring '24

Security & Policy

Module [2] | Topic 02[1]
Machine Learning for Cyber

ML: Typical Setting

Task: Classification

■ Pattern Inference \rightsquigarrow Prediction

- Prediction \neq Forecast
- Where does Q come from?

■ 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?]

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

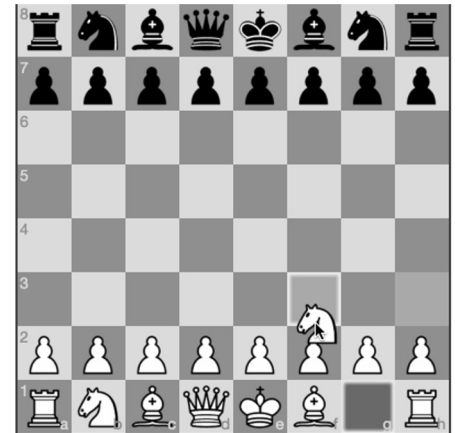
■ CS' Algorithm Approach

[b/w] \rightsquigarrow [w] \rightsquigarrow [a2:a3] \rightsquigarrow [a7:a6]

[.....] [.....]

[h2:h3] [h7:h6] \rightsquigarrow result_i

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



ML: Typical Setting

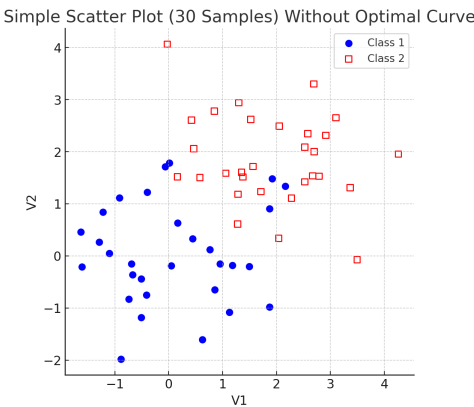
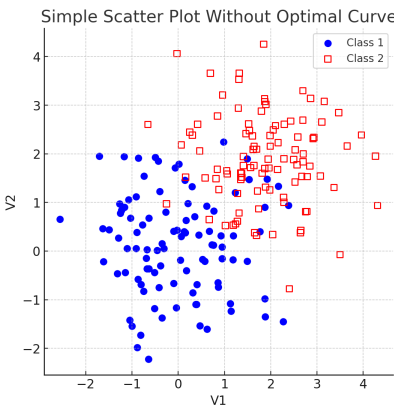
Task: Classification

■ Stats' Approach

- **Pr() Model:** Expected move ~ observed data
- Prediction \neq Forecast

■ Pattern Inference \rightsquigarrow Prediction

- Prediction \neq Forecast
- Where does Q come from?



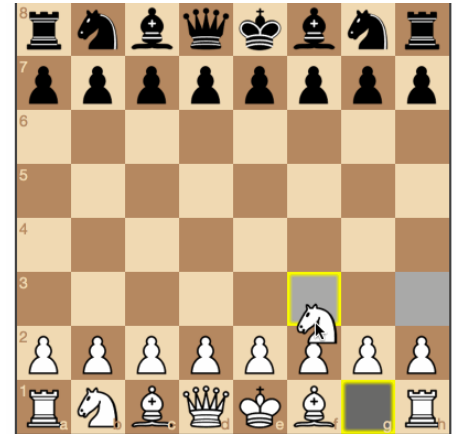
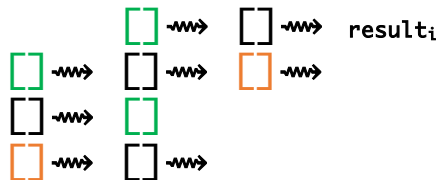
ML: Typical Task

□ Algorithm: [1] if

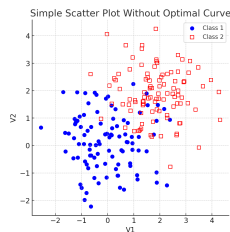
[game theory analogy]

[dominant approach most of the history]

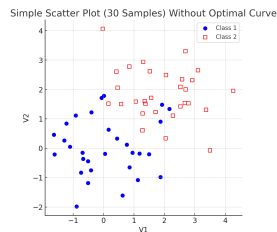
[chess gamedev]



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



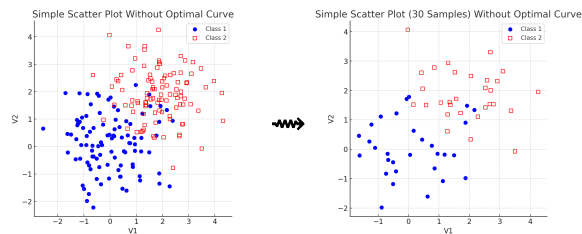
~>



~> next step

ML: Typical Task

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

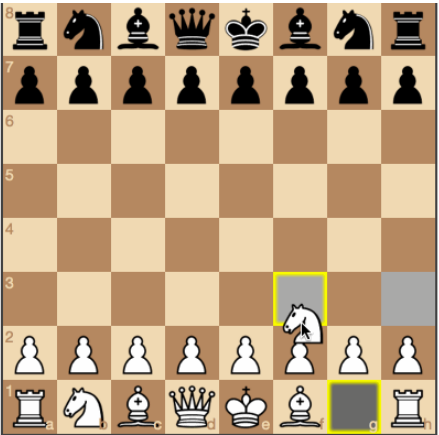


[queen with 5 choices example]

□ Model Structure:

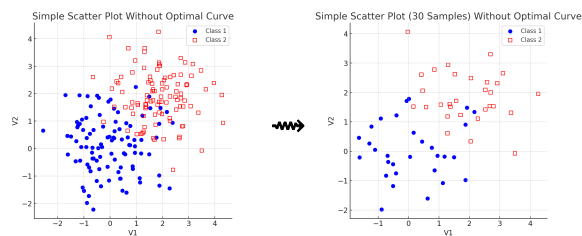
[Probability == Filtering]

	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

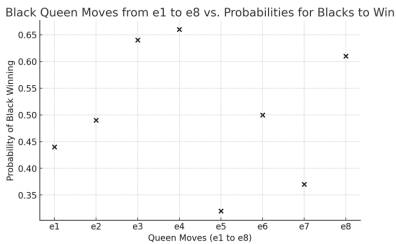
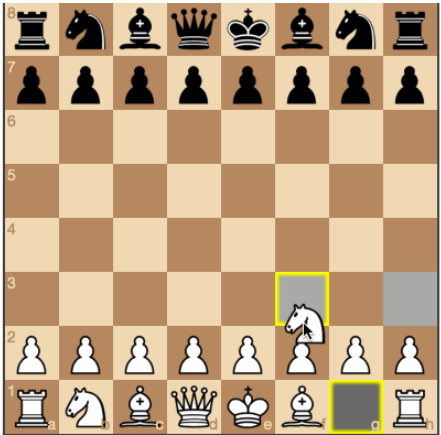
□ Pr() Model: Expected move [value] ~ observed games



[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]

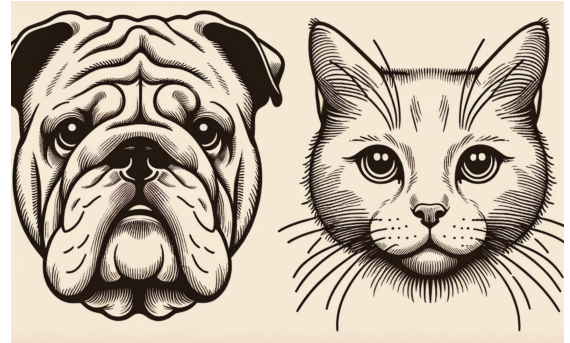
Statistical Approach

[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 \leadsto Statistical Model \rightsquigarrow **ML**
- Cats VS Dogs [How?]
- [3] Game-changers [Revolutions]

■ Unsupervised Learning

- Efficient Information Aggregation
 - \rightsquigarrow □ Dimensionality Reduction
- Clustering
- Anomaly Detection

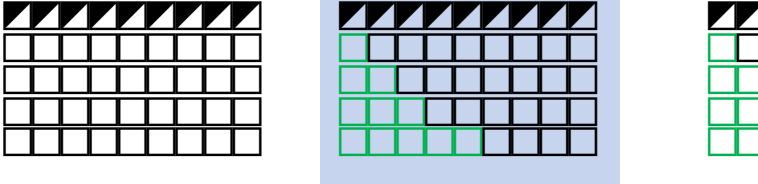
The diagram illustrates the hierarchical structure of a 3D convolutional layer, showing three levels of feature maps:

- Level 1:** A single cell (1x1x1).
- Level 2:** A 2x2x2 grid of cells (8 cells total). A sub-region of 2x2x2 cells is highlighted.
- Level 3:** A 4x4x4 grid of cells (64 cells total). A sub-region of 4x4x4 cells is highlighted.

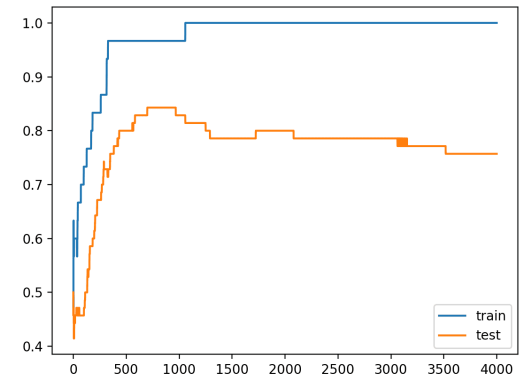
Arrows indicate the flow from Level 1 to Level 2, and from Level 2 to Level 3, showing how the spatial dimensions increase at each level.

[illegible]

Media Storage / File Storage



■ Revolution [3]: Iterations and Batch Piece-by-Piece Analysis \rightsquigarrow Same result



Subheader ##

Subheader ##

in thi



Naïve ML: Issues

Subheader ##
Subheader ##

■ Location Dependency

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

■ Cost

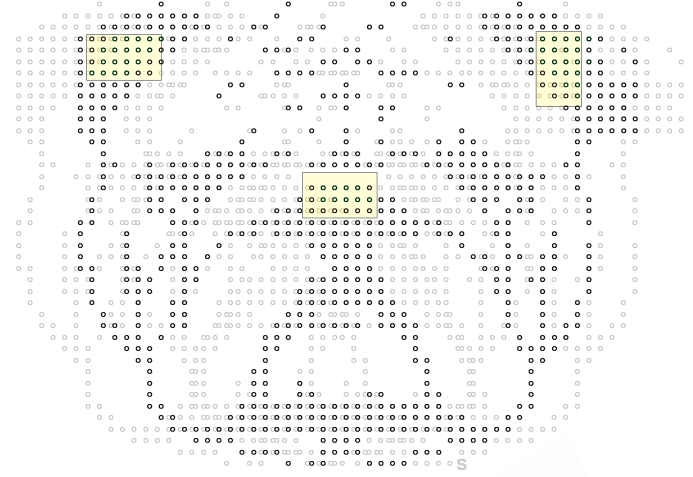
- Storage □ Memory [200k²]
- Processing Speed

■ Inefficiency

- Only part of input data is informative

■ Solution: Dimension Reduction

- Principal Components
- Eigen Value Decomposition
- Numeric Approximations



Naïve ML: Issues

Subheader ##
Subheader ##

■ Location Dependency

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

■ Cost

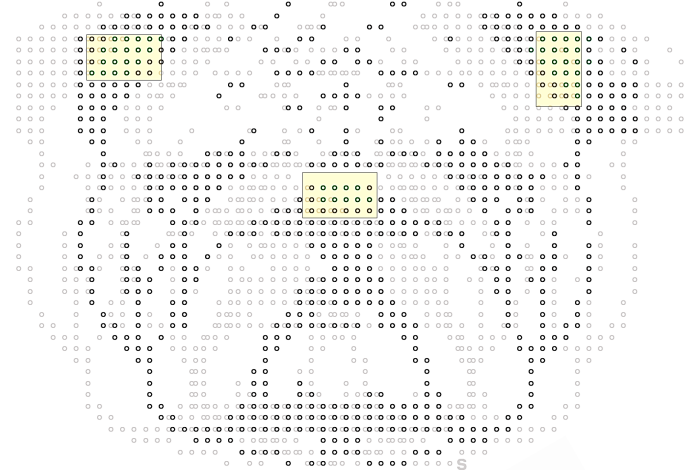
- Storage □ Memory [200k²]
- Processing Speed

■ Inefficiency

- Only part of input data is informative

■ Solution: Dimension Reduction

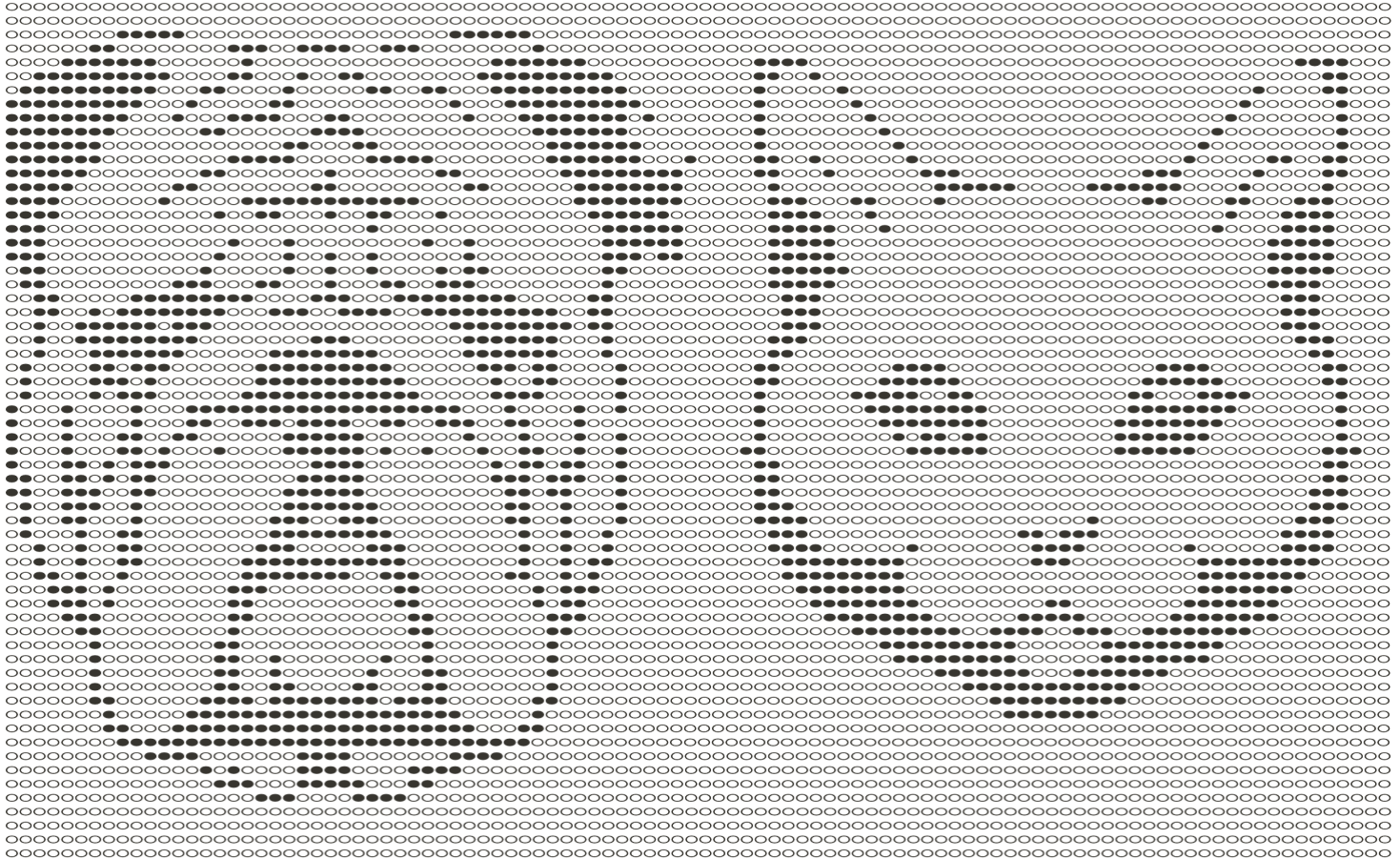
- Principal Components
- Eigen Value Decomposition
- Numeric Approximations



Reminder

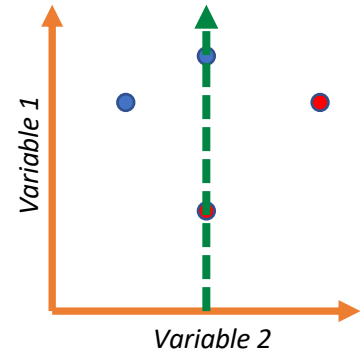
Supervised VS Unsupervised

What is different?



PCA: Dimension Reduction with Principal Components

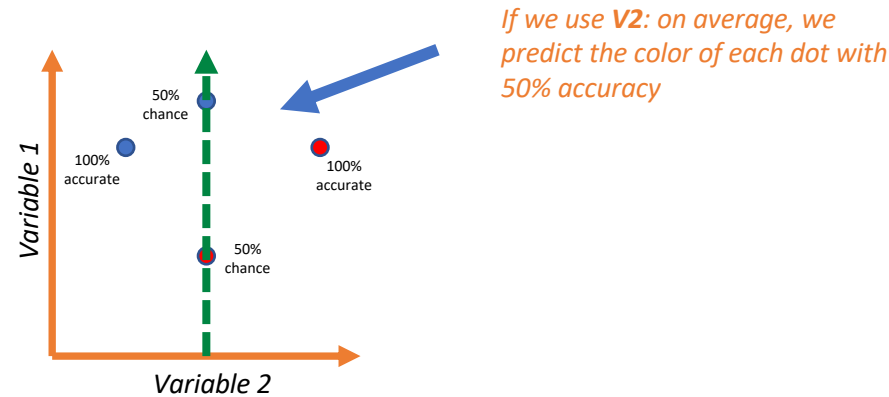
Task: Use only one column (variable) to make best possible prediction of the type (color) of each point



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

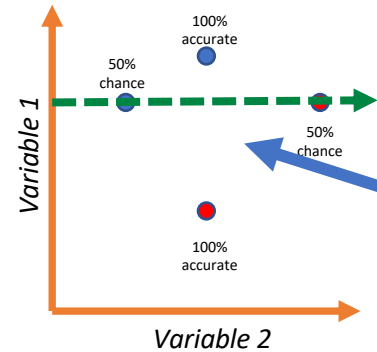
Task: Use only one column (variable) to make best possible prediction of the type (color) of each point



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



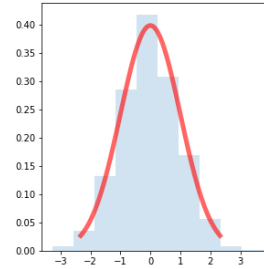
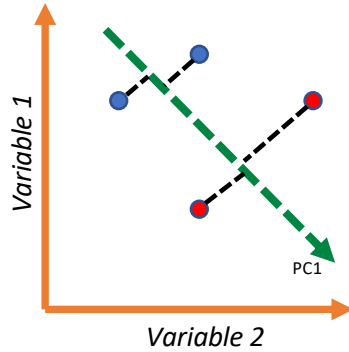
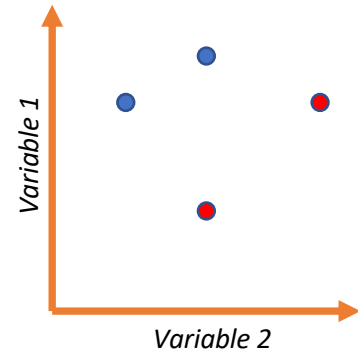
*If we use **V1**: on average, we also predict the color of each dot with 50% accuracy*

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

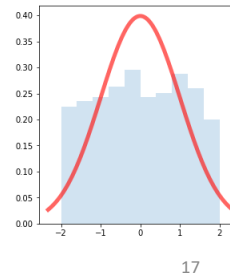
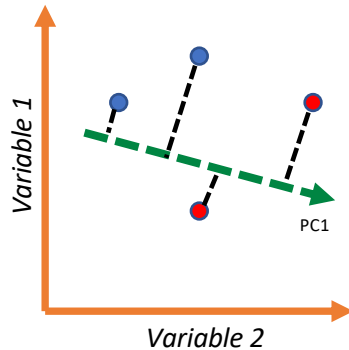
PCA: Dimension Reduction with Principal Components

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

What if we rotate axes?

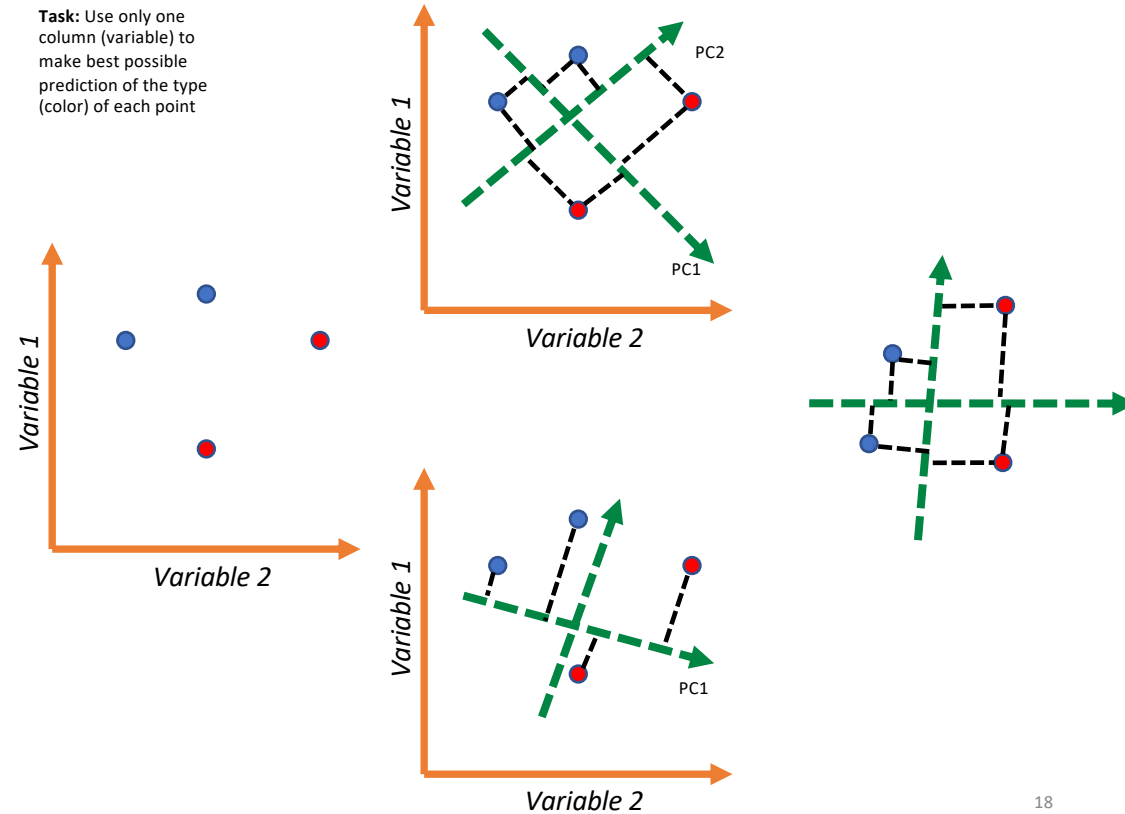


Best rotated axis should have the largest possible distribution of values



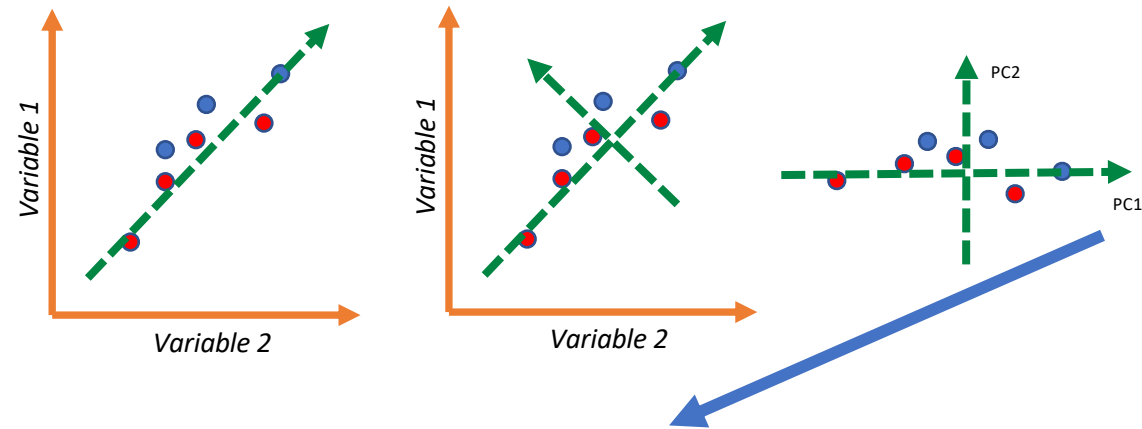
PCA: Dimension Reduction with Principal Components

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



PCA: Dimension Reduction with Principal Components

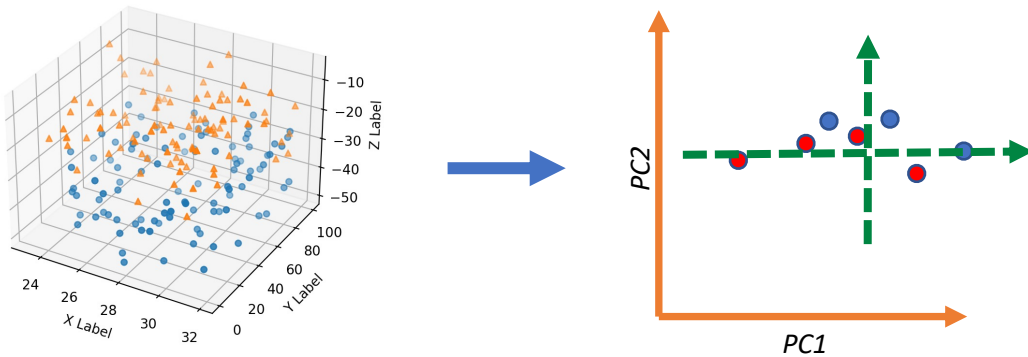
PCA also solves the problem of 'duplicates with noise'



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

Subheader ##
Subheader ##

Task: Classification

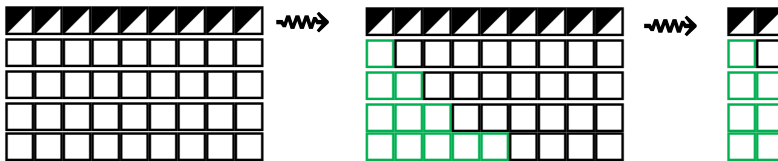
■ Rotation

□ [Decision-Tree works always now]



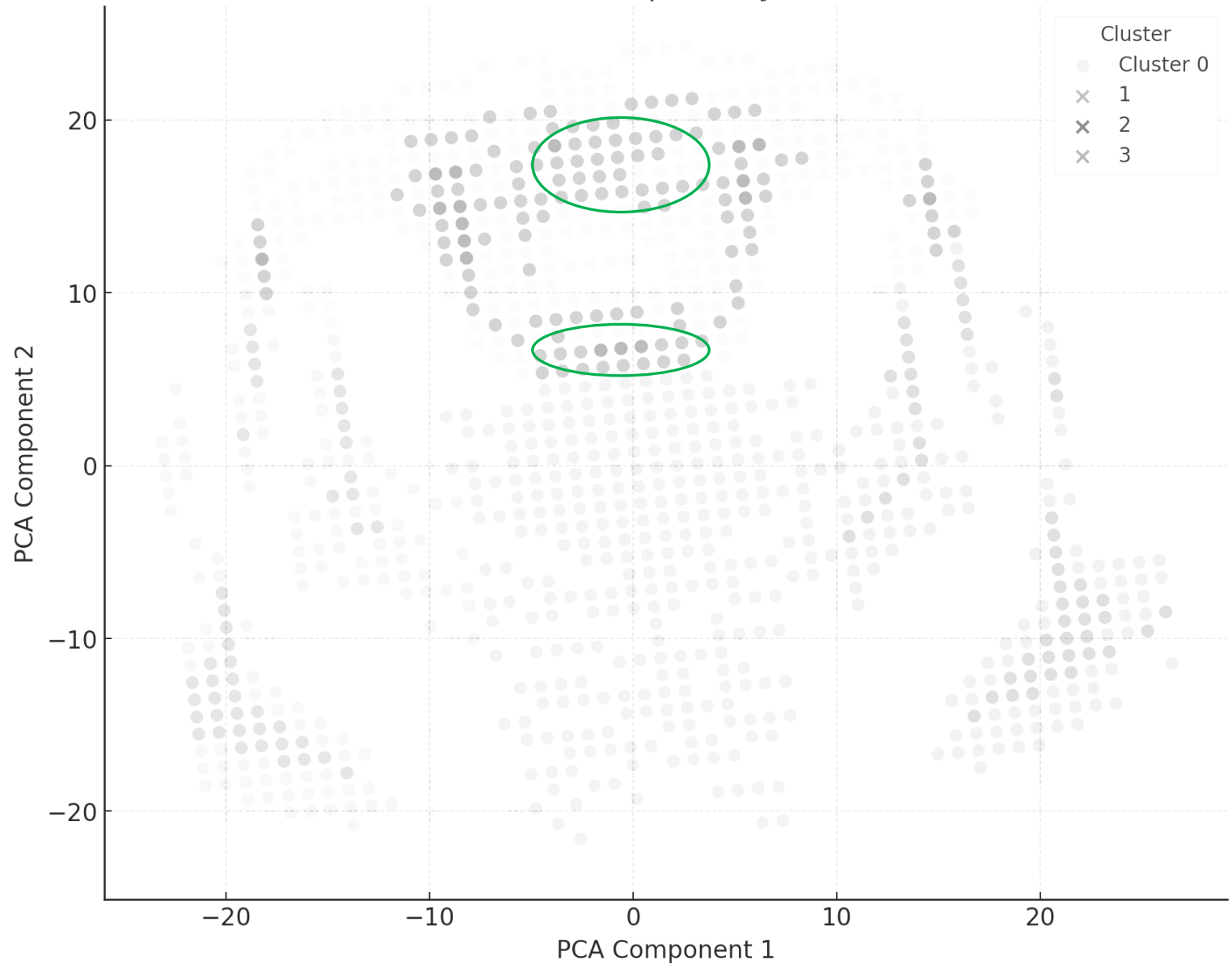
■ Memory & Computational Capacity

□ [Decision-Tree works always now]




■ Overuse of Information [Data Duplication] [Wrong Clustering]

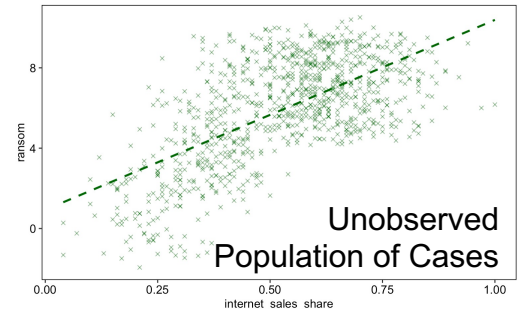
Scatter Plot with Increased Transparency for All But the First Cluster



Machine Learning


Goals

- ☐ Pattern Inference  Prediction
 - ☐ Prediction \neq Forecast
- ☐ Clustering (groups based on similarity)
- ☐ Anomaly Detection (fraudulent transaction)
- ☐ Regression (phishing attacks)
- ☐ Classification (cats VS dogs)
- ☐ Advanced Simulation (deepfakes, play chess, gpt-chat)

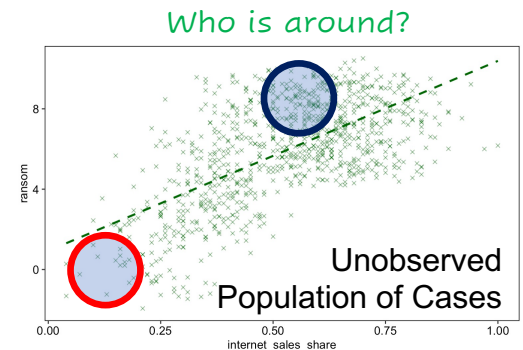


Machine Learning

Goals

- ☐ Pattern Inference  Prediction
- ☐ Prediction \neq Forecast

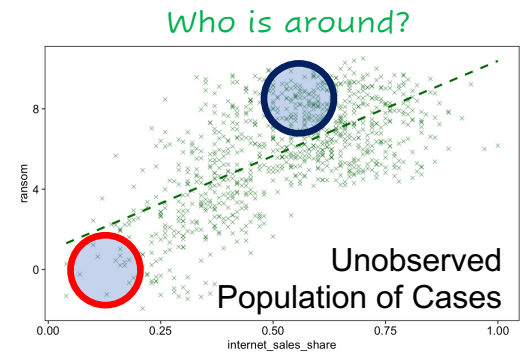
- ☐ Clustering (groups based on similarity)
- ☐ Anomaly Detection (fraudulent transaction)
- ☐ Regression (phishing attacks)
- ☐ Classification (cats VS dogs)
- ☐ Advanced Simulation (deepfakes, play chess, gpt-chat)



Unsupervised Learning

Machine Learning Workflow

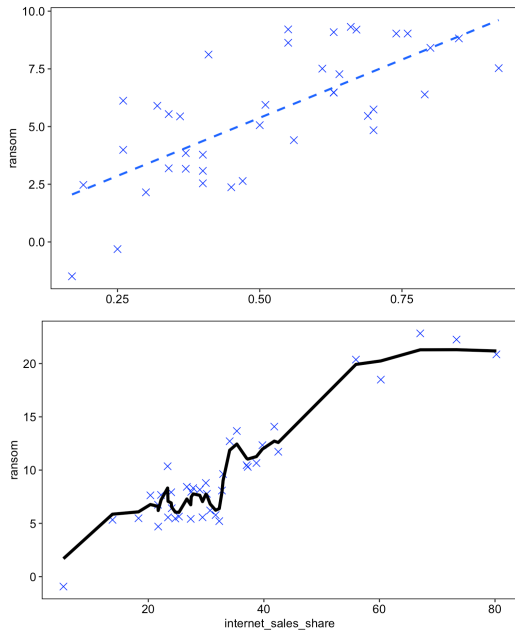
1. Raw Data Sampling
2. Feature Extraction
3. Data Split
4. Train Model
5. Evaluate Performance
6. Make Predictions!



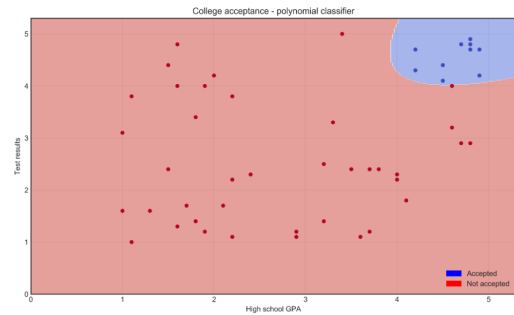
Unsupervised Learning

What is a ML-model?

Regression



Anomaly Detection



Clustering



Sampling Techniques

Over-sampling, under-sampling,
synthetic control