Feature Engineering

Rina BUOY, PhD

ChatGPT 4.0

Disclaimer

Adopted from

6.390

Introduction to Machine Learning
(Fall 2024)

https://introml.mit.edu/fall24

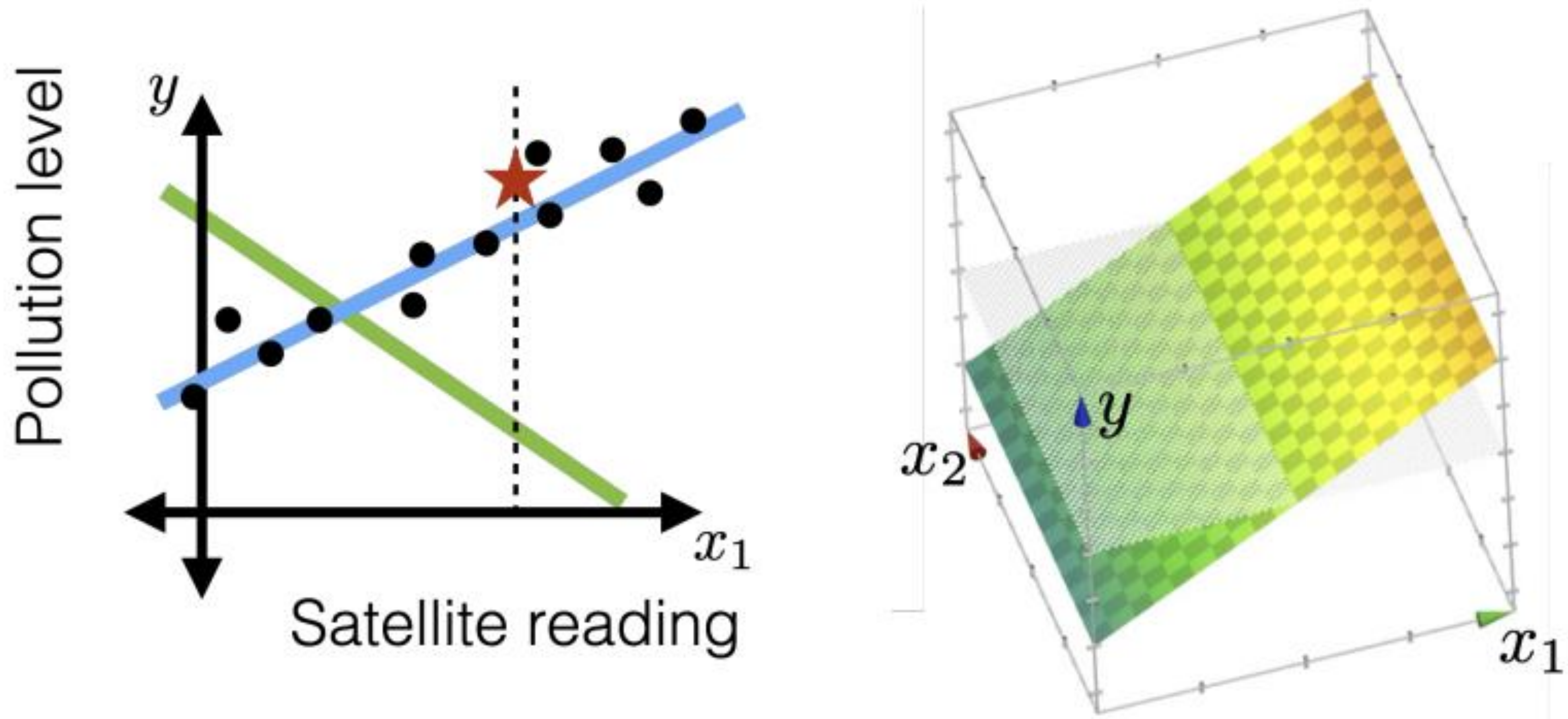
Adopted from : https://introml.mit.edu

Outline

- Recap, linear models and beyond
- Systematic feature transformations
  - Polynomial features
  - Expressive power
- Hand-crafting features
  - One-hot
  - Factored
  - Standardization/normalization
  - Thermometer

Adopted from : https://introml.mit.edu

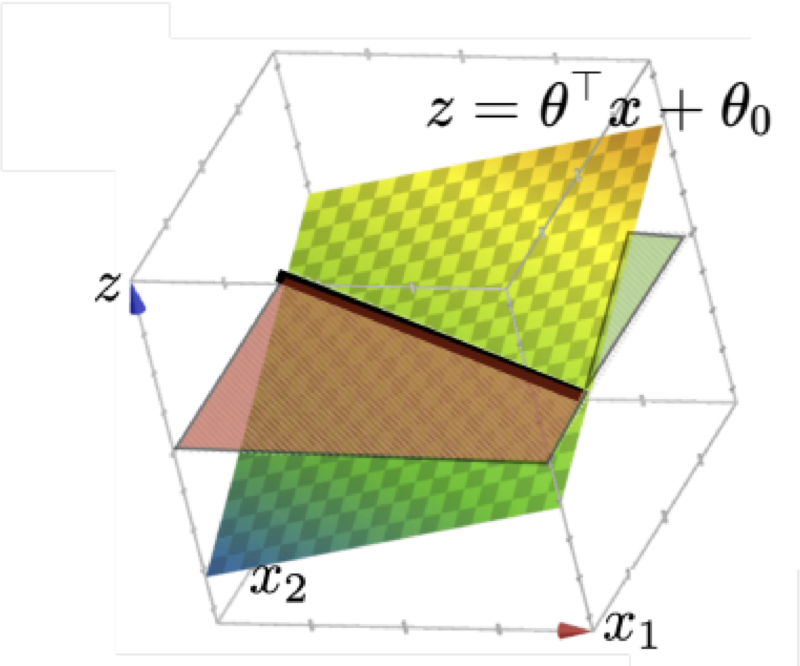
Recap: linear regressor $y = \theta^\top x + \theta_0$



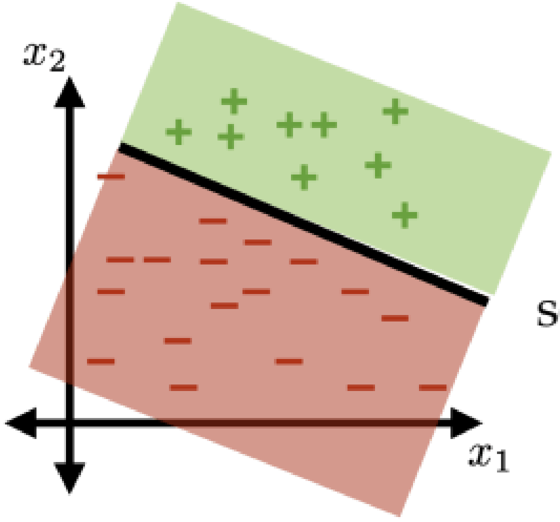
the regressor is **linear** in the feature x

Adopted from : https://introml.mit.edu

Recap: linear (sign-based) classifier



$$\{x : \theta^\top x + \theta_0 > 0\}$$



separator

$$\{x : \theta^\top x + \theta_0 = 0\}$$

$$\{x : \theta^\top x + \theta_0 < 0\}$$

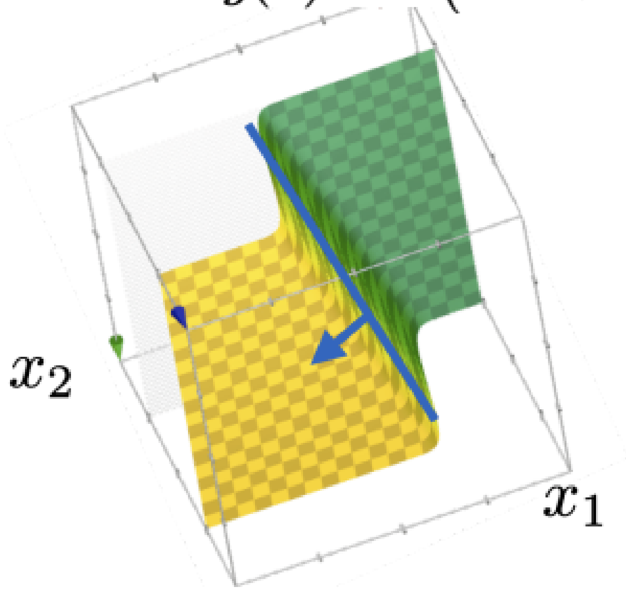
the separator is **linear** in the feature x

Adopted from : https://introml.mit.edu

Recap:

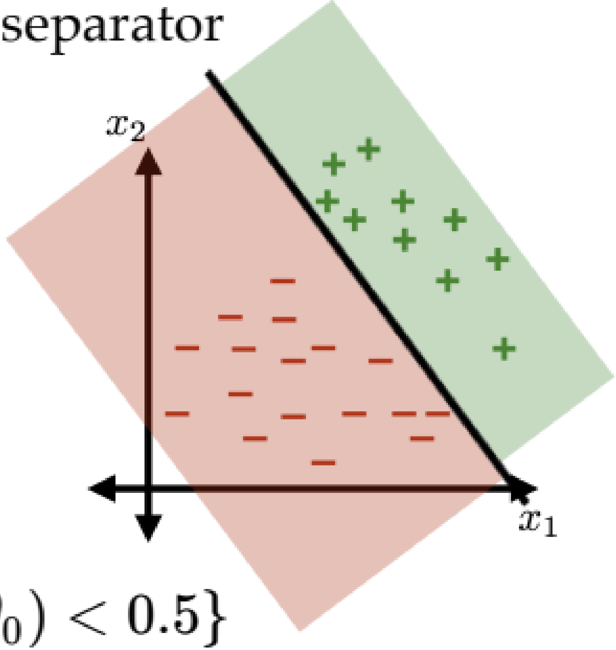
linear logistic classifier

$g(x) = \sigma\left(\theta^\top x + \theta_0\right)$



separator

$\{x : \sigma(\theta^\top x + \theta_0) > 0.5\}$



$\{x : \sigma(\theta^\top x + \theta_0) < 0.5\}$

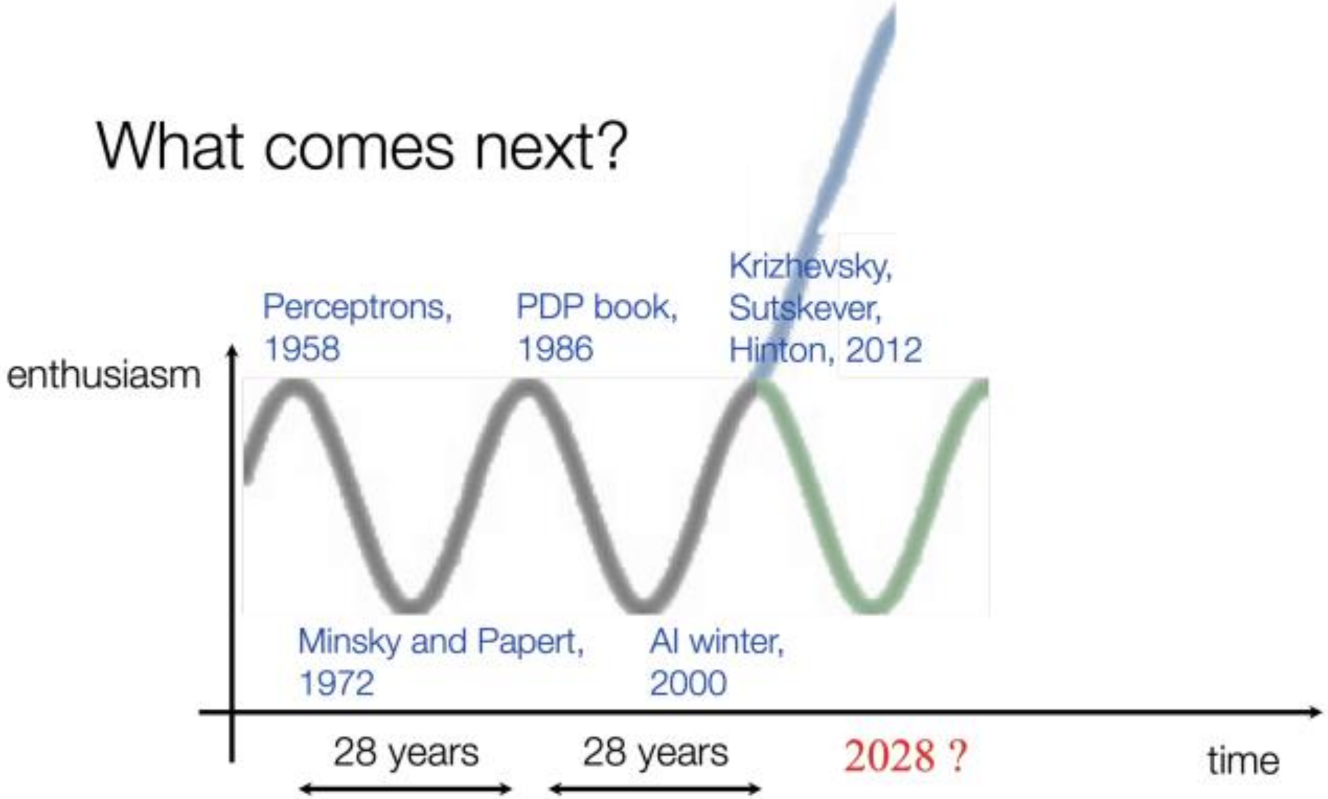
$\{x : \theta^\top x + \theta_0 = 0\}$

the separator is **linear** in the feature x

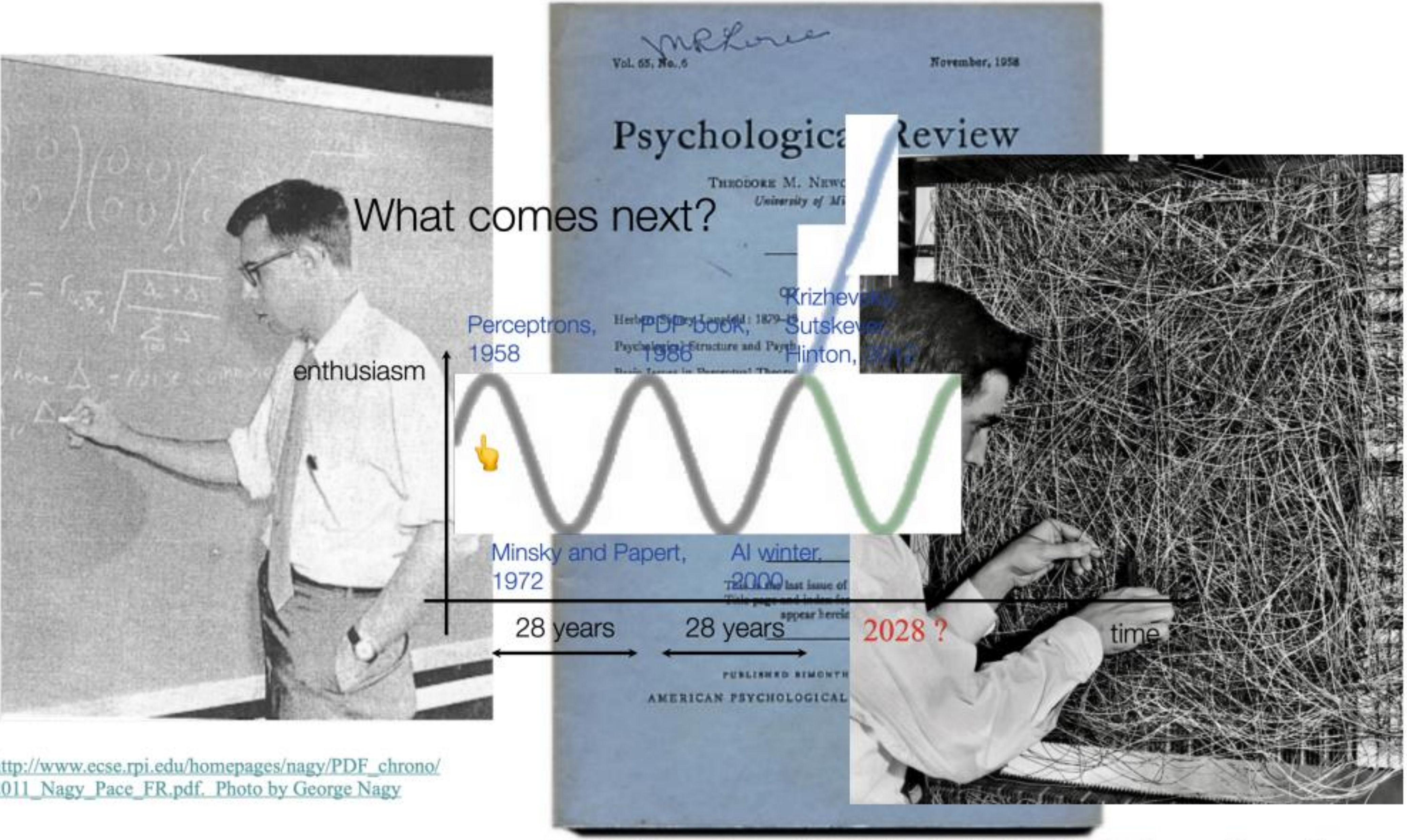
Adopted from : https://introml.mit.edu

Linear classification played a pivotal role in kicking off the first wave of AI enthusiasm.

Image classification played a pivotal role in kicking off the current wave of AI enthusiasm.



Adopted from : https://introml.mit.edu



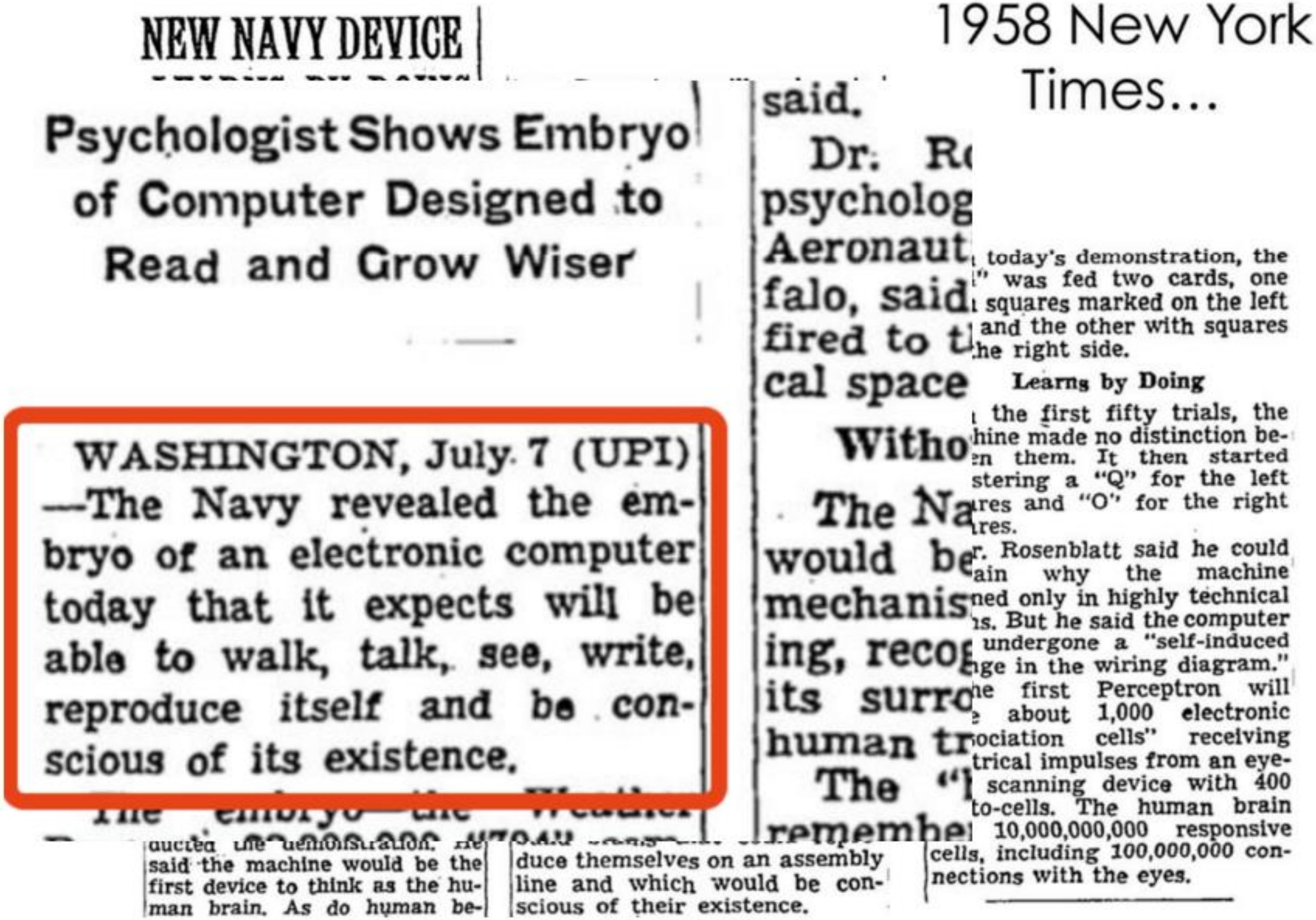
What comes next?

http://www.ecse.rpi.edu/homepages/nagy/PDF_chrono/2011_Nagy_Pace_FR.pdf. Photo by George Nagy

http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.335.3398&rep=rep1&type=pdf

Adopted from : https://introml.mit.edu

1958 New York Times...



NEW NAVY DEVICE

Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

WASHINGTON, July 7 (UPI) —The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

The embryo—the Weather

ducted the demonstration. He said the machine would be the first device to think as the human brain. As do human be-

said.

Dr. R

psycholog

Aeronaut

falo, said

fired to t

cal space

Witho

The Na

would be

mechanis

ing, recog

its surro

human tr

The "

remember

duce themselves on an assembly line and which would be conscious of their existence.

today's demonstration, the " was fed two cards, one squares marked on the left and the other with squares he right side.

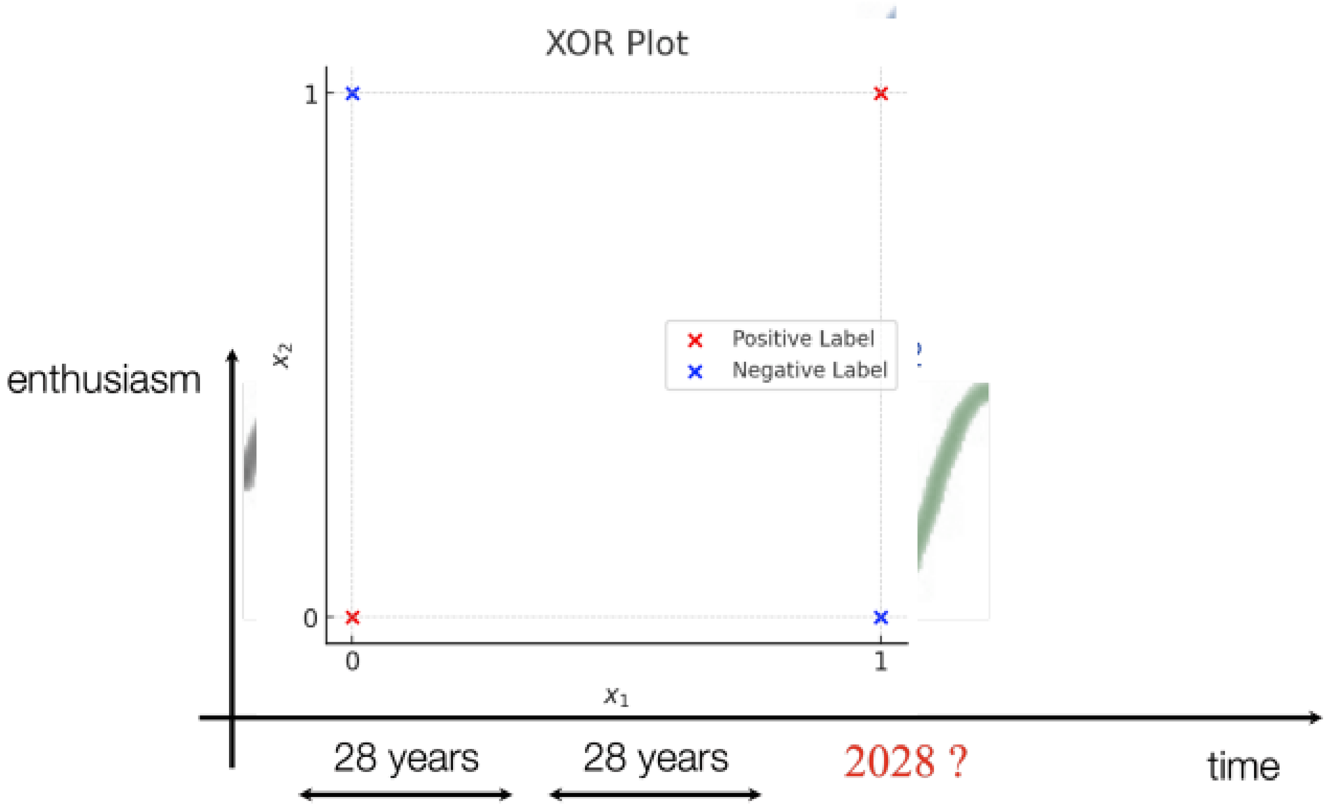
Learns by Doing

the first fifty trials, the hine made no distinction been them. It then started stering a "Q" for the left res and "O" for the right res.

r. Rosenblatt said he could ain why the machine ned only in highly technical s. But he said the computer undergone a "self-induced ge in the wiring diagram."

he first Perceptron will about 1,000 electronic ociation cells" receiving trical impulses from an eye- scanning device with 400 to-cells. The human brain 10,000,000,000 responsive cells, including 100,000,000 connections with the eyes.

Adopted from : https://introml.mit.edu



Not **linearly** separable.

~~Linear tools cannot solve interesting tasks.~~

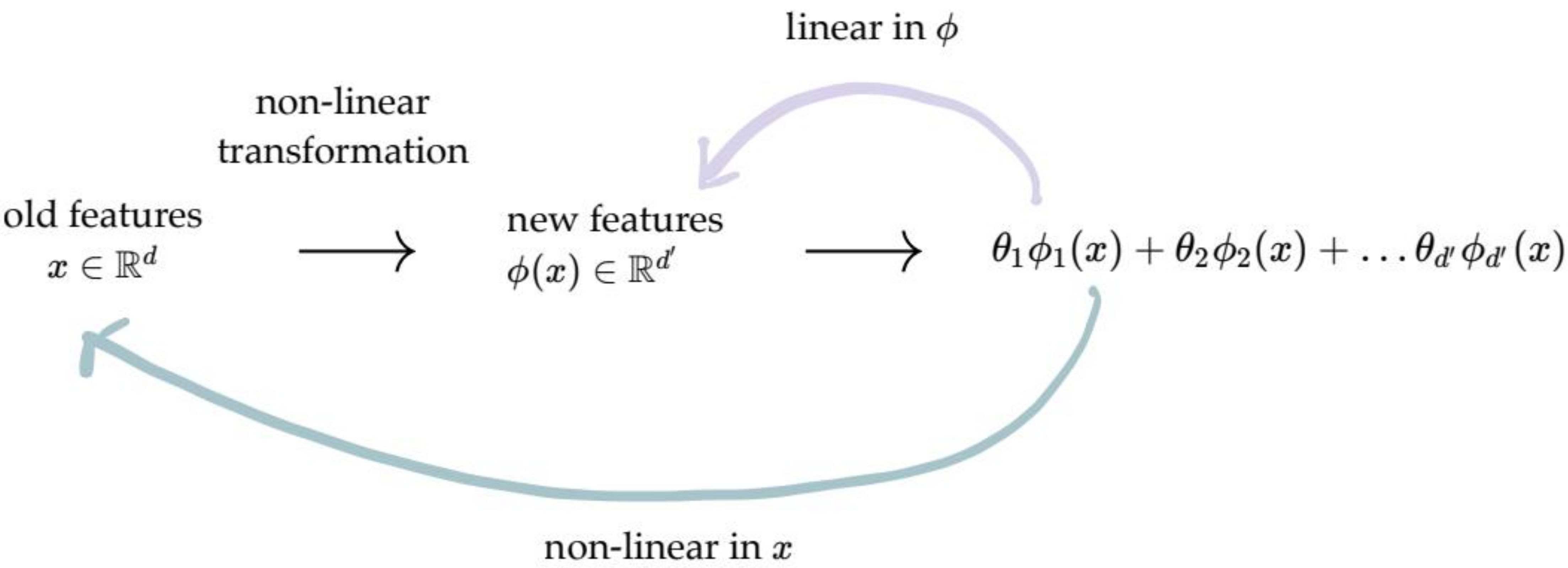
Linear tools cannot, *by themselves*, solve interesting tasks.

Many cool ideas can "help out" linear tools. We'll focus on one today.

Outline

- Recap, linear models and beyond
- Systematic feature transformations
  - Polynomial features
  - Expressive power
- Hand-crafting features
  - One-hot
  - Factored
  - Standardization / normalization
  - Thermometer

Adopted from : https://introml.mit.edu



linear in ϕ

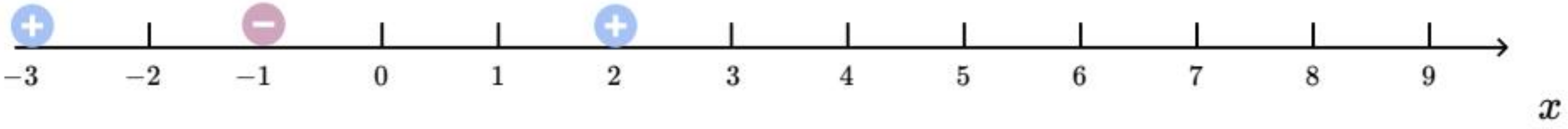
non-linear transformation

old features $x \in \mathbb{R}^d$ $\longrightarrow$ new features $\phi(x) \in \mathbb{R}^{d'}$ $\longrightarrow$ $\theta_1\phi_1(x) + \theta_2\phi_2(x) + \ldots \theta_{d'}\phi_{d'}(x)$

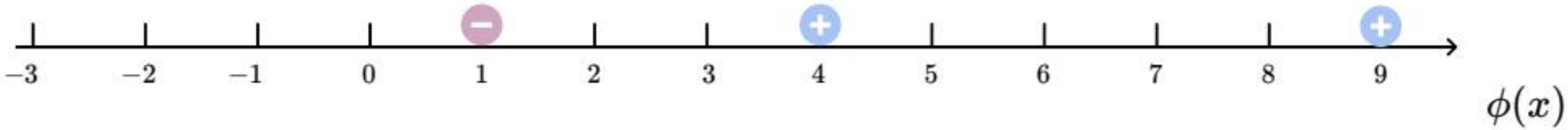
non-linear in x

Adopted from : https://introml.mit.edu

Not linearly separable in x space



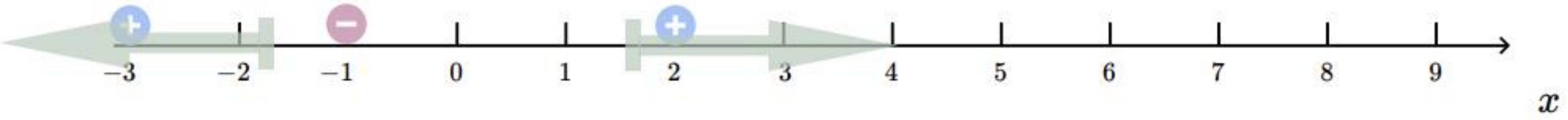
$\Downarrow$ transform via $\phi(x) = x^2$



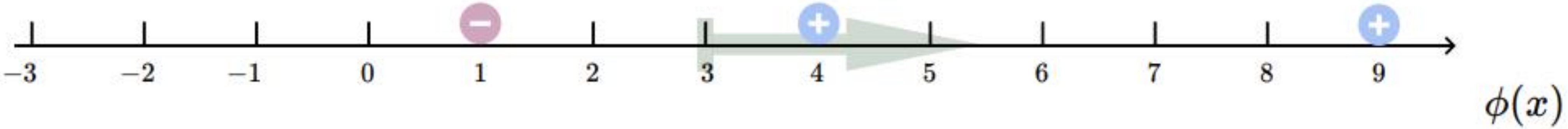
Linearly separable in $\phi(x) = x^2$ space

Adopted from : https://introml.mit.edu

Non-linearly separated in x space, e.g. predict positive if $x^2 \geq 3$

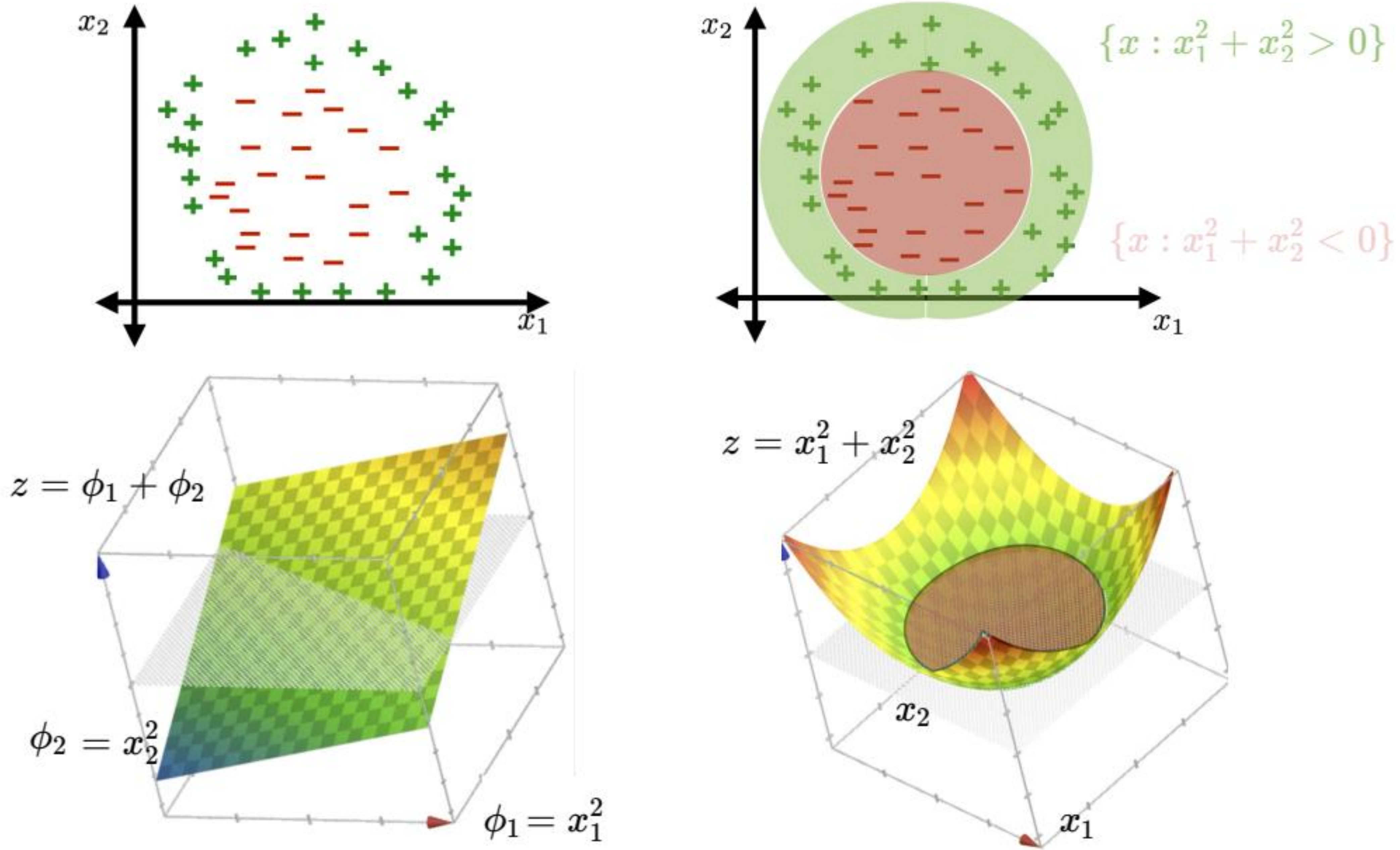


$\Downarrow$ transform via $\phi(x) = x^2$



Linearly separated in $\phi(x) = x^2$ space, e.g. predict positive if $\phi \geq 3$

Adopted from : https://introml.mit.edu



Adopted from : https://introml.mit.edu

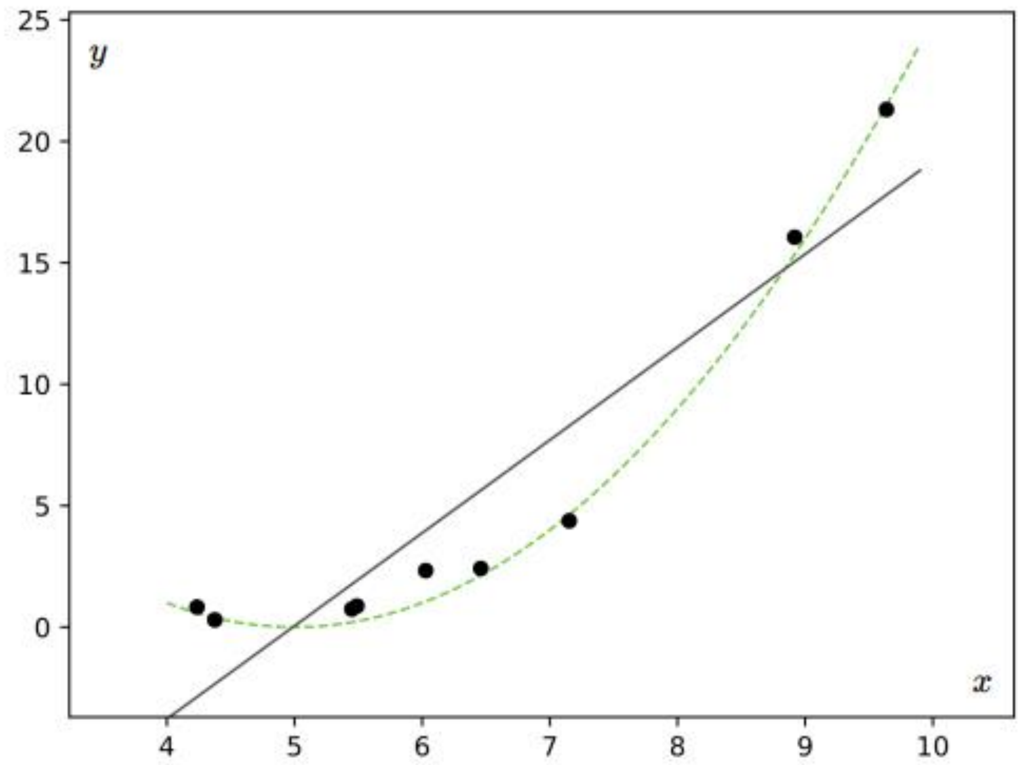
systematic polynomial feature transformation construction

	$d = 1$	$d = 2$	$\cdots$
$k = 0$	1	1	
$k = 1$	$1, x_1$	$1, x_1, x_2$	
$k = 2$	$1, x_1, x_1^2$	$1, x_1, x_2, x_1^2, x_1x_2, x_2^2$	
$k = 3$	$1, x_1, x_1^2, x_1^3$	$1, x_1, x_2, x_1^2, x_1x_2, x_2^2, x_1^3, x_1^2x_2, x_1x_2^2, x_2^3$	
$\cdots$			

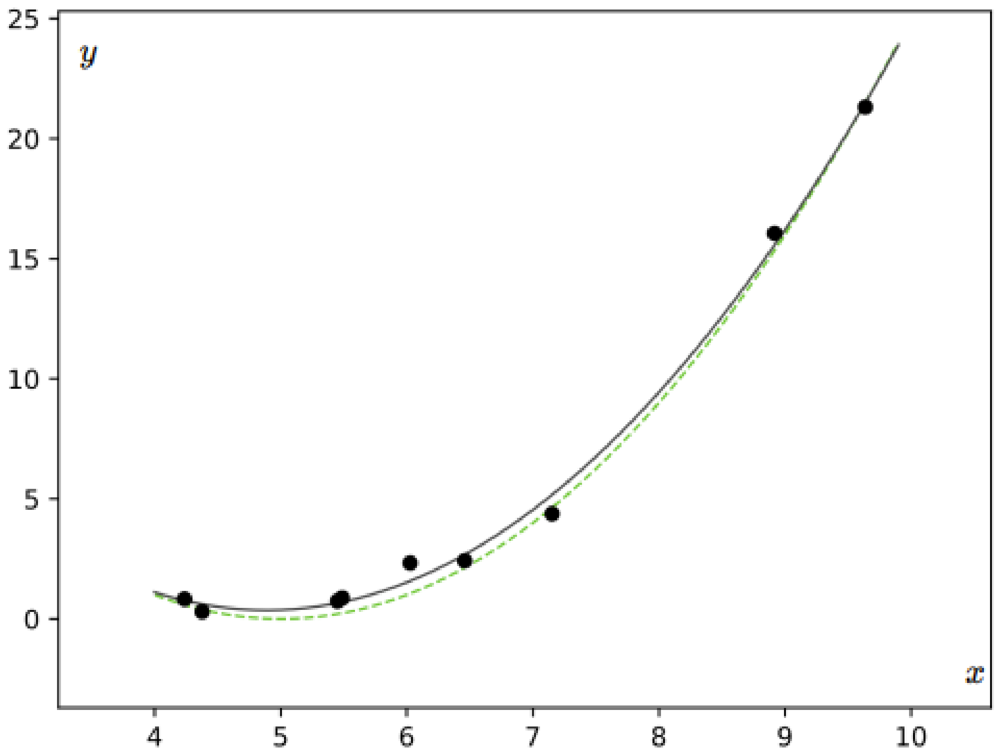
- Elements in the basis are the monomials of original features raised up to power k
- With a given d and a fixed k, the basis is **fixed**.

Adopted from : https://introml.mit.edu

9 data points; each has feature $x \in \mathbb{R}$, label $y \in \mathbb{R}$

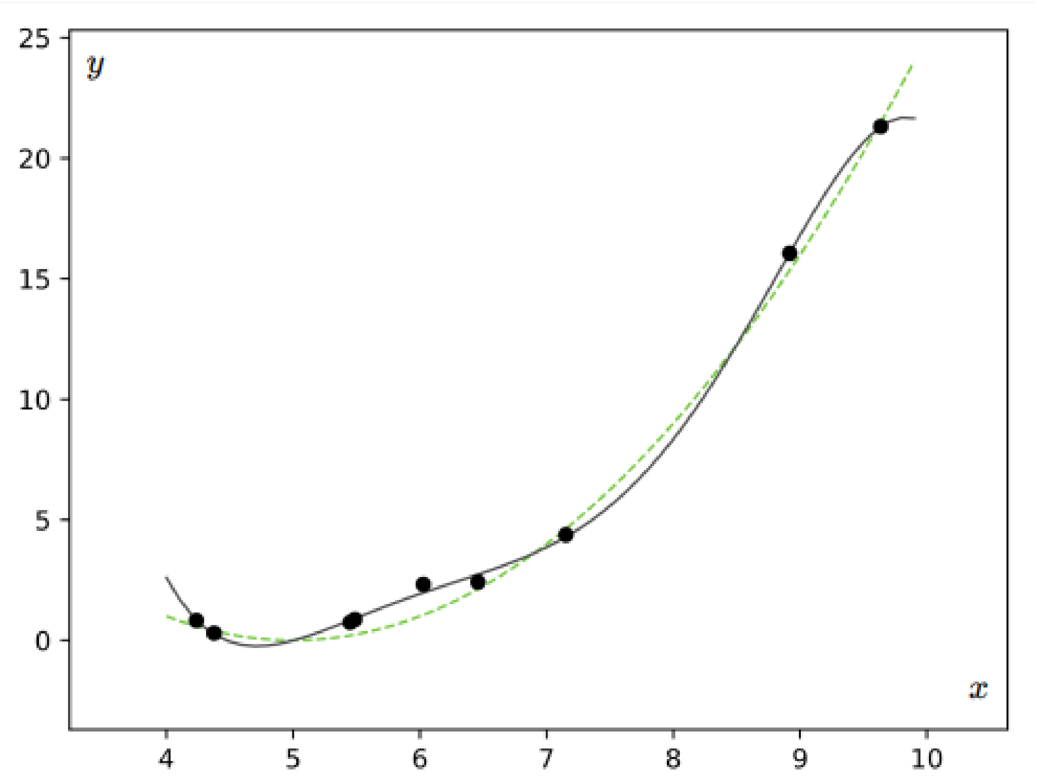


- Choose $k = 1$
- New features $\phi = [1; x]$
- $h(x; \theta) = \theta_0 + \theta_1 x$
- Learn 2 parameters for linear function



- Choose $k = 2$
- New features $\phi = [1; x; x^2]$
- $h(x; \theta) = \theta_0 + \theta_1 x + \theta_2 x^2$
- Learn 3 parameters for quadratic function

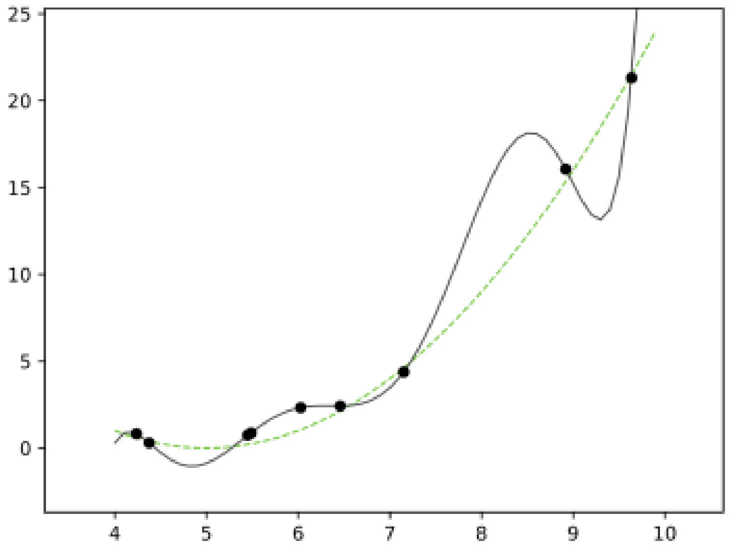
Adopted from : https://introml.mit.edu



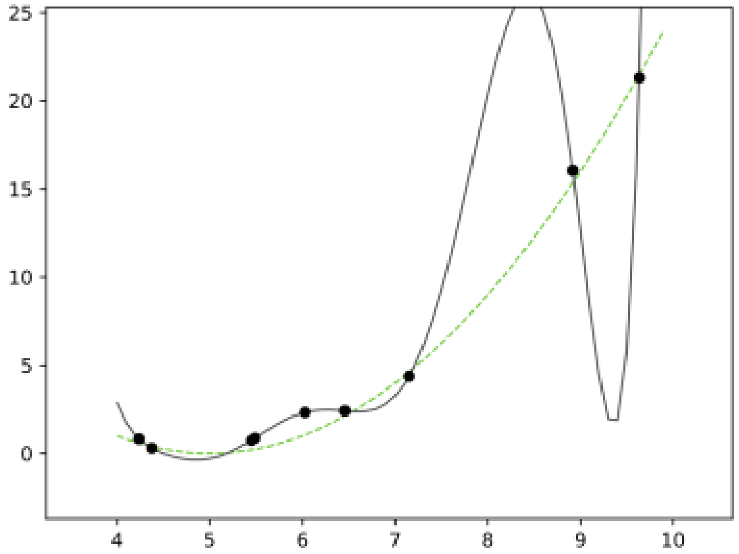
- Choose $k = 5$
- New features $\phi = [1; x; x^2; x^3; x^4; x^5]$
- $h(x; \theta) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4 + \theta_5 x^5$
- Learn 6 parameters for degree-5 polynomial function

Adopted from : https://introml.mit.edu

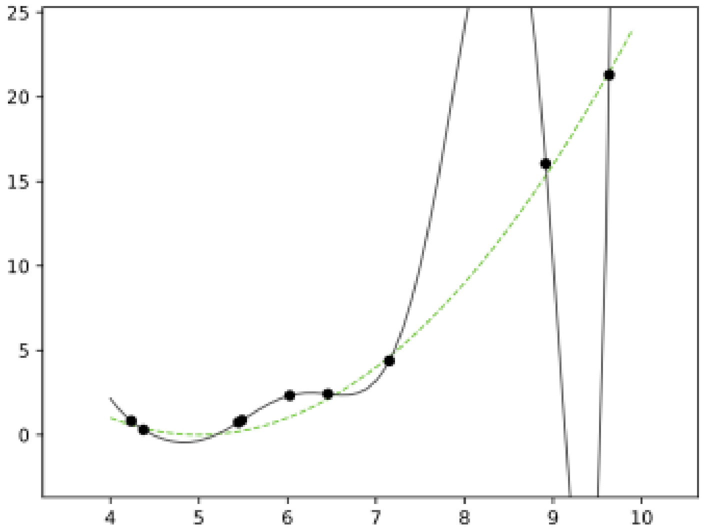
$k = 7$



$k = 8$



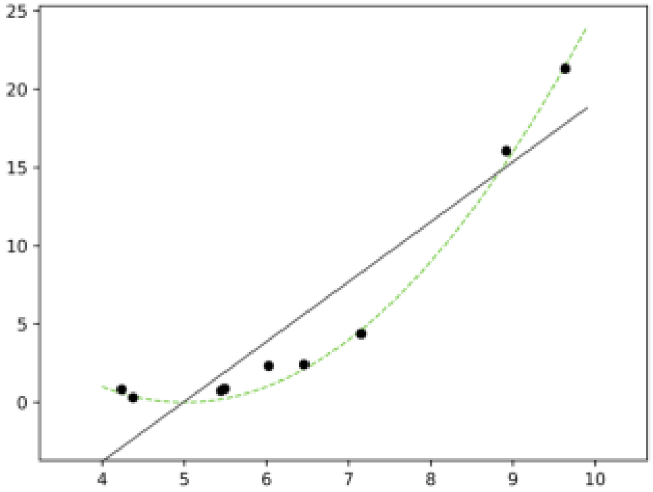
$k = 10$



Adopted from : https://introml.mit.edu

Underfitting

$k = 1$

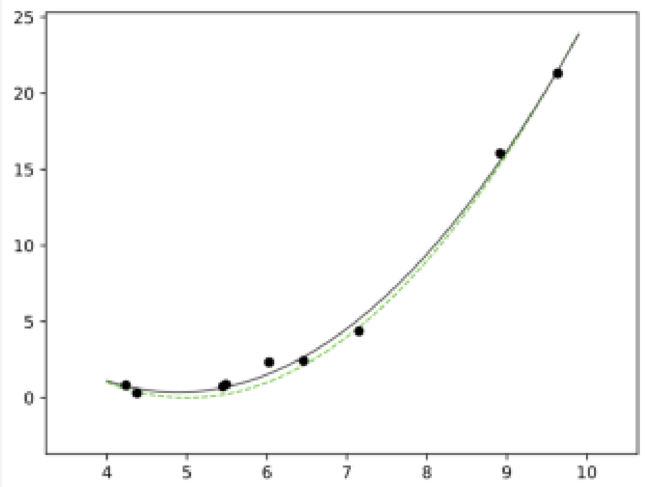


high error on train set

high error on test set

Appropriate model

$k = 2$

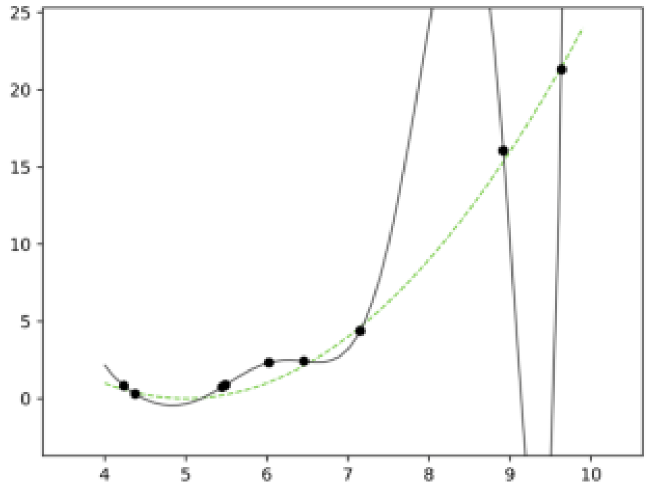


low error on train set

low error on test set

Overfitting

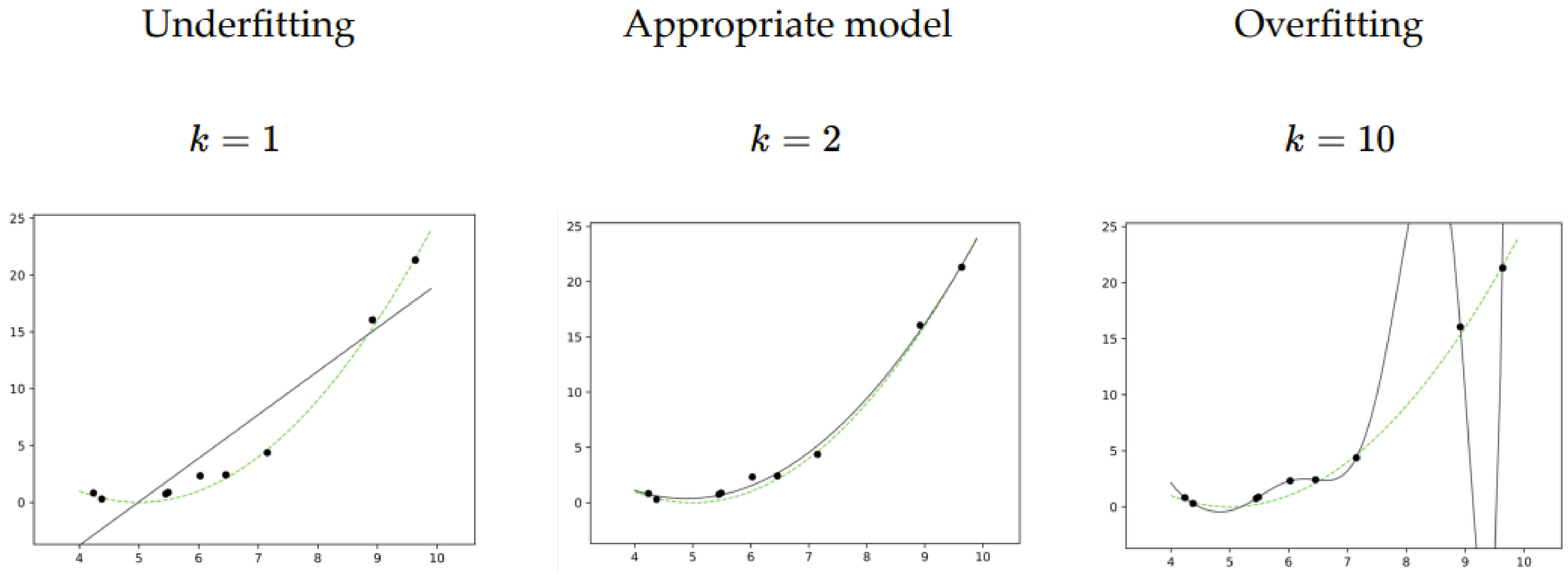
$k = 10$



very low error on train set

very high error on test set

Adopted from : https://introml.mit.edu



Underfitting

$k = 1$

Appropriate model

$k = 2$

Overfitting

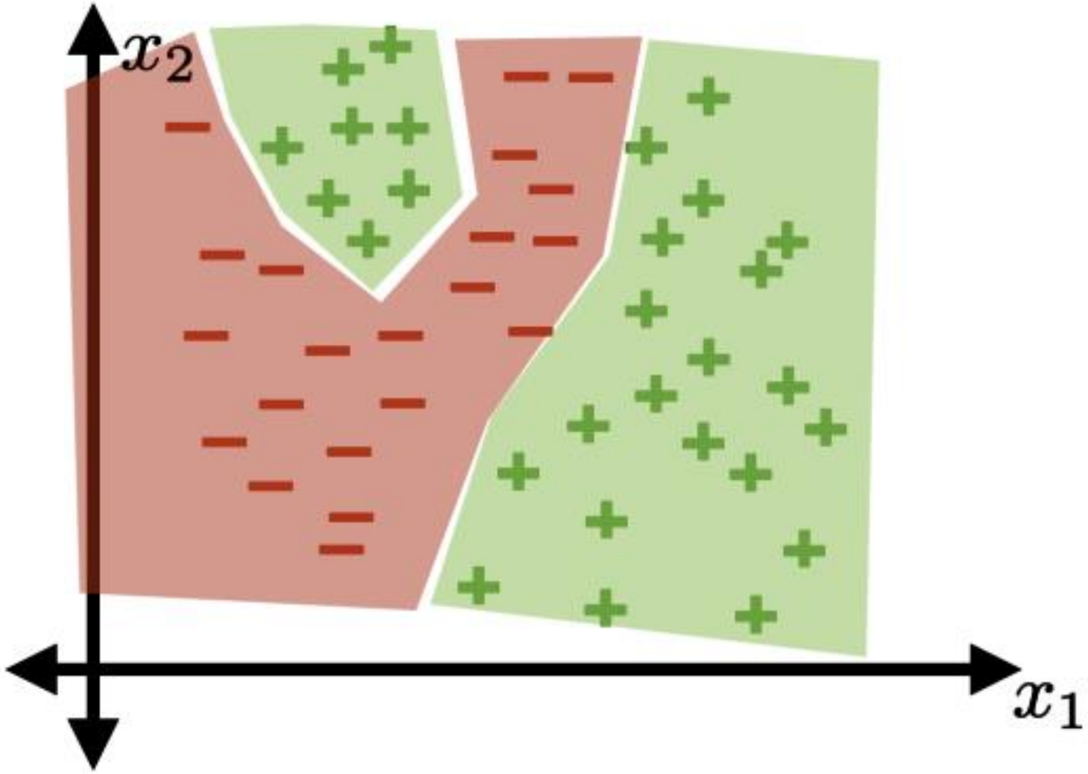
$k = 10$

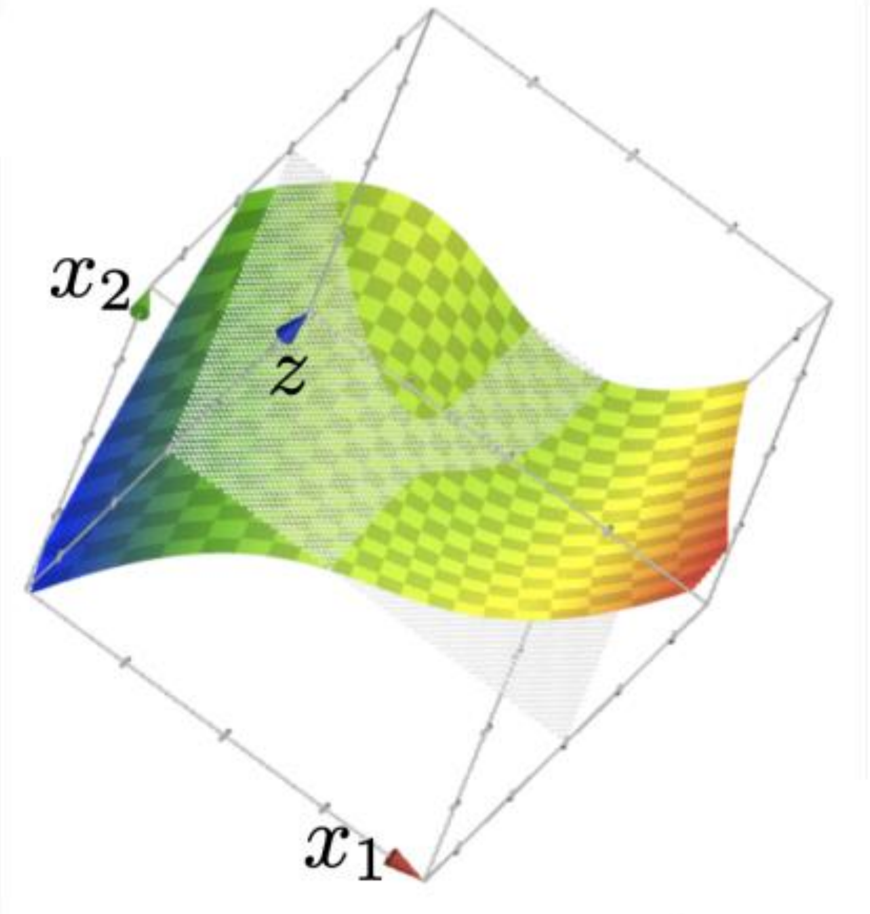
- k is a hyperparameter that controls the capacity (expressiveness) of the hypothesis class.
- Complex models with many rich features and free parameters have high capacity.
- How to choose k? Validation/cross-validation.

Adopted from : https://introml.mit.edu

Similar overfitting can happen in classification

Using polynomial features of order 3





Adopted from : https://introml.mit.edu

Quick summary

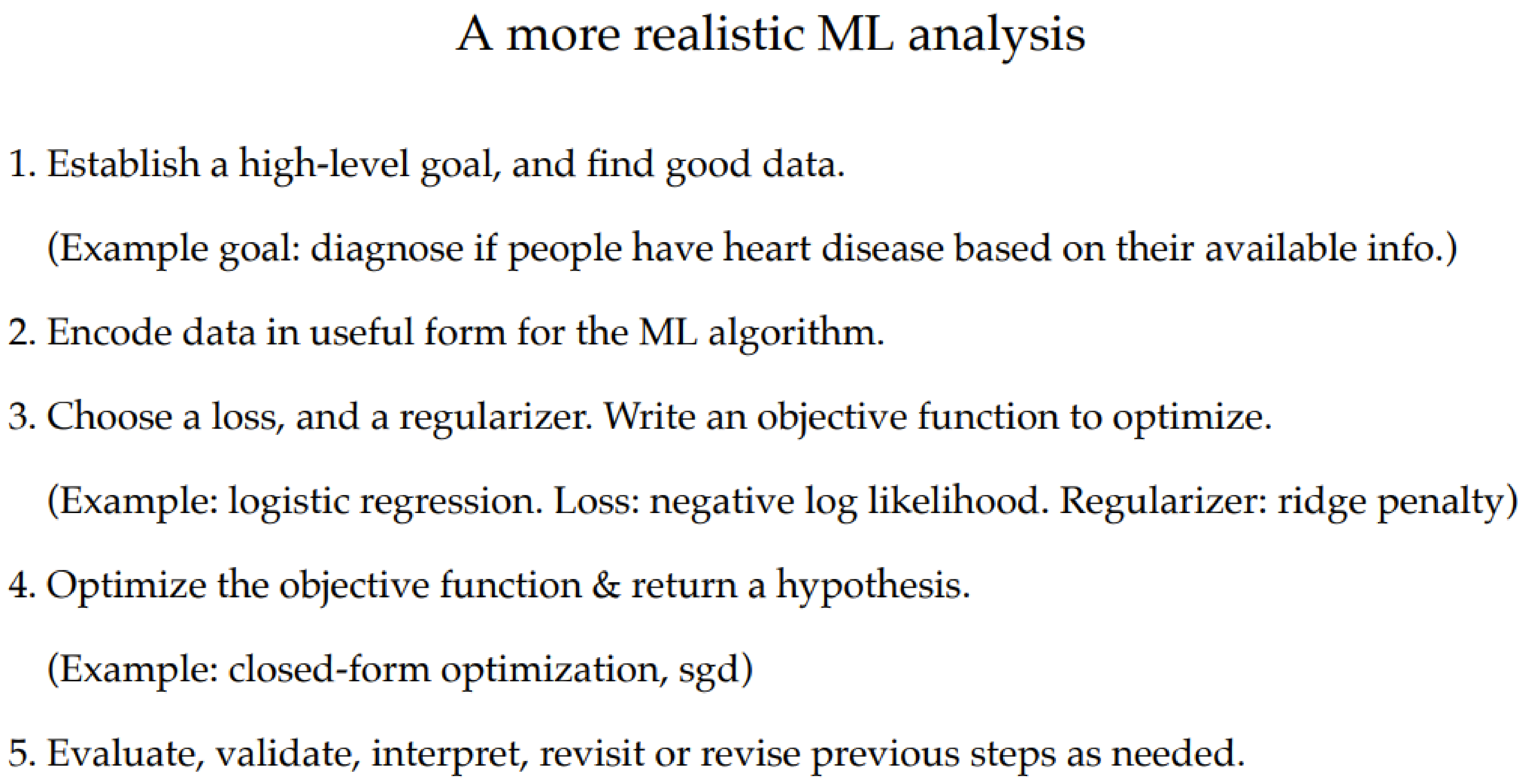
- Linear models are mathematically and algorithmically convenient but not expressive enough -- by themselves -- for most jobs.
- We can express really rich hypothesis classes by performing a **fixed** non-linear feature transformation first, then applying our linear regression or classification methods.
- Can think of fixed transformation as "adapters", enabling us to use old tools in broader situations.
- Standard feature transformations: polynomials; radial basis functions, absolute-value function.
- Historically, for a period of time, the gist of ML boils down to "feature engineering".
- Nowadays, neural networks can automatically extract out features.

Adopted from : https://introml.mit.edu

Outline

- Recap, linear models and beyond
- Systematic feature transformations
  - Polynomial features
  - Expressive power
- Hand-crafting features
  - One-hot
  - Factored
  - Standardization/normalization
  - Thermometer

Adopted from : https://introml.mit.edu



A more realistic ML analysis

1. Establish a high-level goal, and find good data.

   (Example goal: diagnose if people have heart disease based on their available info.)

2. Encode data in useful form for the ML algorithm.

3. Choose a loss, and a regularizer. Write an objective function to optimize.

   (Example: logistic regression. Loss: negative log likelihood. Regularizer: ridge penalty)

4. Optimize the objective function & return a hypothesis.

   (Example: closed-form optimization, sgd)

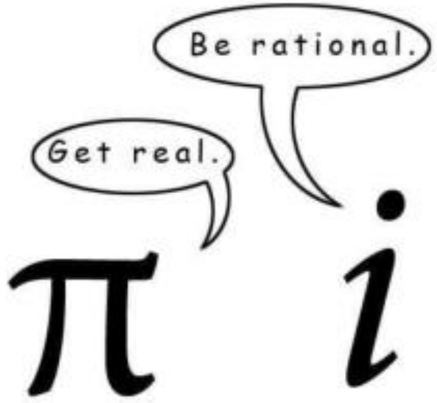
5. Evaluate, validate, interpret, revisit or revise previous steps as needed.

Adopted from : https://introml.mit.edu

Encode data in useful form for the ML algorithm.

Identify relevant info and encode as **real** numbers

Encode in such a way that's **reasonable** for the task.



Adopted from : https://introml.mit.edu

Example: diagnose whether people have heart disease based on their available info.

- go collect training data.

	has heart disease?	pain?	job	medicines	resting heart rate (bpm)	family income (USD)	
p1	no $y^{(1)}$	no	nurse	aspirin	55	133000	$x^{(1)}$
p2	no	no	admin	beta blockers, aspirin	71	34000	
p3	yes	yes	nurse	beta blockers	89	40000	
p4	no	no	doctor	none	67	120000	

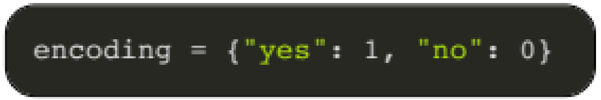
⋮

label

features

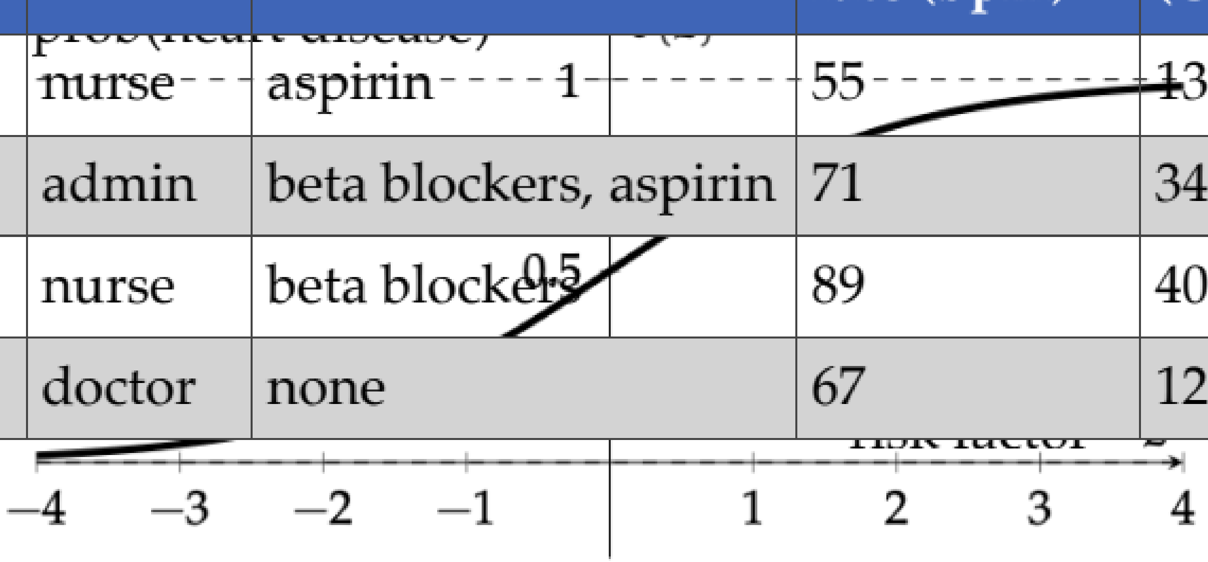
Adopted from : https://introml.mit.edu

- Turn binary labels to {0,1}, save mapping to recover predictions of new points



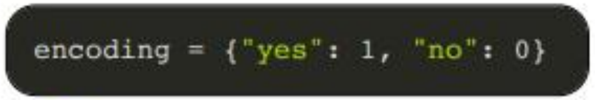
```
encoding = {"yes": 1, "no": 0}
```

	has heart disease?	pain?	job	medicines	resting heart rate (bpm)	family income (USD)
p1	no	**no**	nurse	aspirin	55	133000
p2	no	no	admin	beta blockers, aspirin	71	34000
p3	yes	**yes**	nurse	beta blockers	89	40000
p4	no	no	doctor	none	67	120000



$$\sigma(z) = \sigma\left(\theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}} x_{\text{job}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}\right)$$

- Encode binary feature answers to {0,1}, has nice interpretation



```
encoding = {"yes": 1, "no": 0}
```

😍

$$z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}} x_{\text{job}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$$

	pain?	job	medicines	resting heart rate (bpm)	family income (USD)
p1	0	nurse	aspirin	55	133000
p2	0	admin	beta blockers, aspirin	71	34000
p3	1	nurse	beta blockers	89	40000
p4	0	doctor	none	67	120000

person feeling pain has $z = \theta_{\text{pain}} + \theta_{\text{job}} x_{\text{job}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$

person not feeling pain has $z = \theta_{\text{job}} x_{\text{job}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$

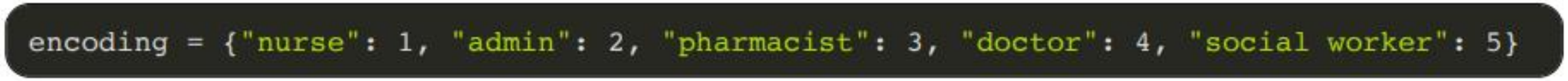
Adopted from : https://introml.mit.edu

Outline

- Recap, linear models and beyond
- Systematic feature transformations
  - Polynomial features
  - Expressive power
- Hand-crafting features
  - One-hot
  - Factored
  - Standardization/normalization
  - Thermometer

Adopted from : https://introml.mit.edu

For "jobs", if use natural number encoding:

🥺 

```
encoding = {"nurse": 1, "admin": 2, "pharmacist": 3, "doctor": 4, "social worker": 5}
```

$$z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}} x_{\text{job}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$$

nurse has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$

admin has $z = \theta_{\text{pain}} x_{\text{pain}} + 2\theta_{\text{job}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$

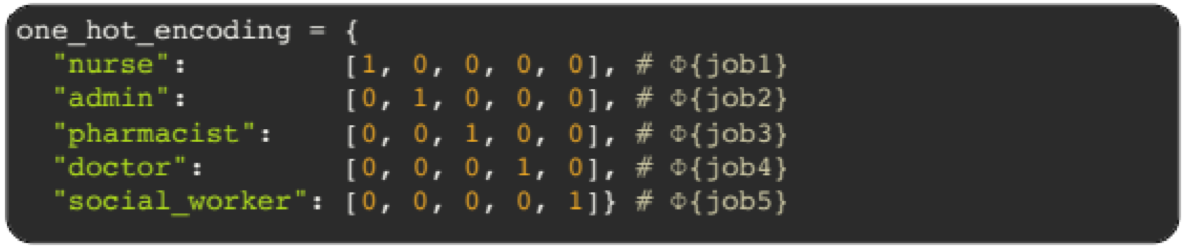
pharmacist has $z = \theta_{\text{pain}} x_{\text{pain}} + 3\theta_{\text{job}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$

problem with this idea:

- Ordering matters
- Incremental in job category affects z by a fixed θ_{job} amount

Adopted from : https://introml.mit.edu

😍



```
one_hot_encoding = {
  "nurse":         [1, 0, 0, 0, 0], # Φ{job1}
  "admin":         [0, 1, 0, 0, 0], # Φ{job2}
  "pharmacist":    [0, 0, 1, 0, 0], # Φ{job3}
  "doctor":        [0, 0, 0, 1, 0], # Φ{job4}
  "social_worker": [0, 0, 0, 0, 1]} # Φ{job5}
```

$$z = \theta_{\text{pain}} x_{\text{pain}} + \underbrace{\theta_{\text{job}}^T x_{\text{job}}}_{\theta_{\text{job1}}\phi_{\text{job1}} + \theta_{\text{job2}}\phi_{\text{job2}} + \theta_{\text{job3}}\phi_{\text{job3}} + \theta_{\text{job4}}\phi_{\text{job4}} + \theta_{\text{job5}}\phi_{\text{job5}}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$$

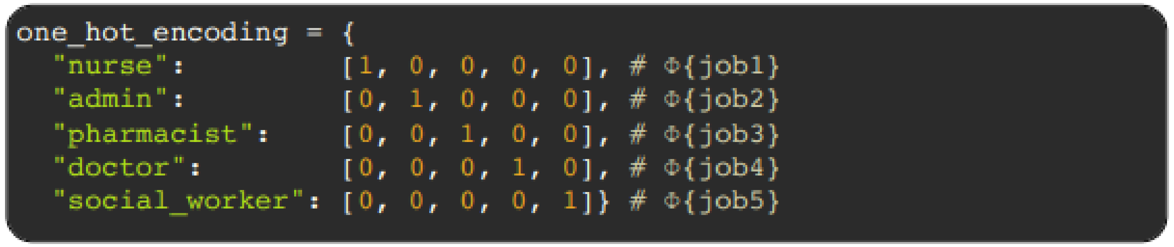
nurse has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job1}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$

admin has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job2}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$

pharmacist has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job3}} + \theta_{\text{pill}} x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$

Adopted from : https://introml.mit.edu

😍



```
one_hot_encoding = {
    "nurse":         [1, 0, 0, 0, 0], # Φ{job1}
    "admin":         [0, 1, 0, 0, 0], # Φ{job2}
    "pharmacist":    [0, 0, 1, 0, 0], # Φ{job3}
    "doctor":        [0, 0, 0, 1, 0], # Φ{job4}
    "social_worker": [0, 0, 0, 0, 1]} # Φ{job5}
```

	pain?	job	medicines	resting heart rate (bpm)	family income (USD)
p1	0	[1,0,0,0,0]	aspirin	55	133000
p2	0	[0,1,0,0,0]	beta blockers, aspirin	71	34000
p3	1	[1,0,0,0,0]	beta blockers	89	40000
p4	0	[0,0,0,1,0]	none	67	120000

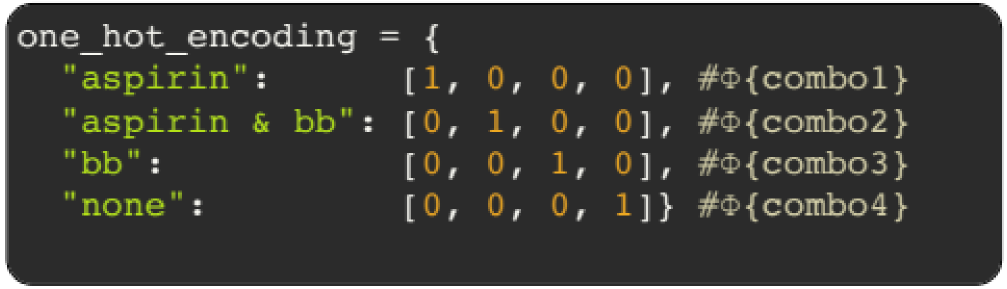
Outline

- Recap, linear models and beyond
- Systematic feature transformations
  - Polynomial features
  - Expressive power
- Hand-crafting features
  - One-hot
  - Factored
  - Standardization/normalization
  - Thermometer

Adopted from : https://introml.mit.edu

For medicines, hopefully obvious why natural number encoding isn't a good idea.
What about one-hot encoding?

🥺



```
one_hot_encoding = {
  "aspirin":       [1, 0, 0, 0], #Φ{combo1}
  "aspirin & bb": [0, 1, 0, 0], #Φ{combo2}
  "bb":            [0, 0, 1, 0], #Φ{combo3}
  "none":          [0, 0, 0, 1]} #Φ{combo4}
```

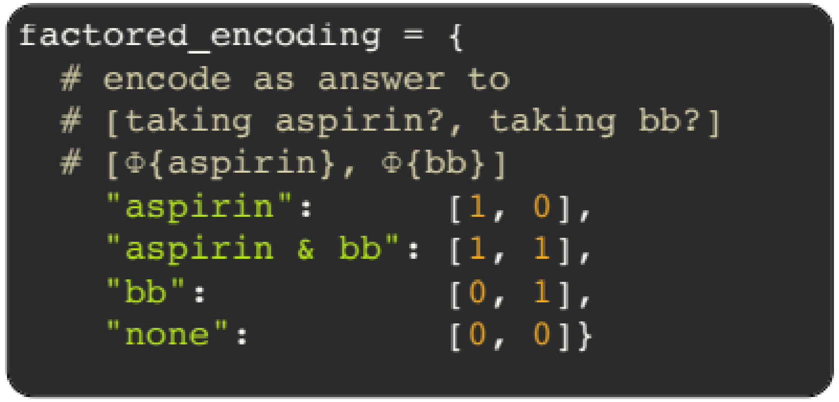
$$z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \underbrace{\theta_{\text{pill}}^T x_{\text{pill}}}_{\theta_{\text{combo1}}\phi_{\text{combo1}} + \theta_{\text{combo2}}\phi_{\text{combo2}} + \theta_{\text{combo3}}\phi_{\text{combo3}} + \theta_{\text{combo4}}\phi_{\text{combo4}}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$$

the natural "association" in combo1, combo2, and combo3 are lost

also, if a combo is very rare (which happens), say only 1 out of 1k surveyed person took combo2, then very hard to learn a meaningful θ_{combo2}

Adopted from : https://introml.mit.edu

😍

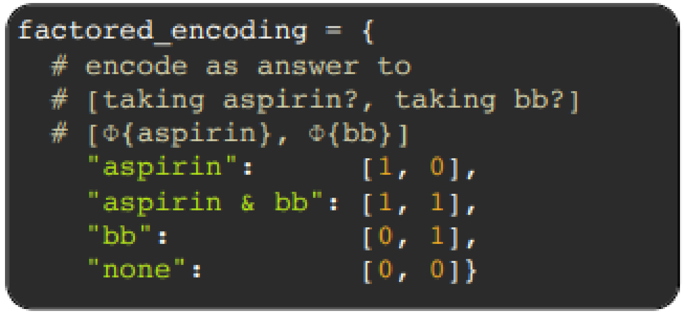


```
factored_encoding = {
  # encode as answer to
  # [taking aspirin?, taking bb?]
  # [Φ{aspirin}, Φ{bb}]
    "aspirin":        [1, 0],
    "aspirin & bb": [1, 1],
    "bb":             [0, 1],
    "none":           [0, 0]}
```

$$z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \underbrace{\theta_{\text{pill}}^T x_{\text{pill}}}_{\theta_{\text{aspirin}} \phi_{\text{aspirin}} + \theta_{\text{beta-blockers}} \phi_{\text{beta-blockers}}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}}$$

Adopted from : https://introml.mit.edu

😍



```
factored_encoding = {
  # encode as answer to
  # [taking aspirin?, taking bb?]
  # [Φ{aspirin}, Φ{bb}]
    "aspirin":        [1, 0],
    "aspirin & bb": [1, 1],
    "bb":             [0, 1],
    "none":           [0, 0]}
```

	pain?	job	medicines	resting heart rate (bpm)	family income (USD)
p1	0	[1,0,0,0,0]	[1,0]	55	133000
p2	0	[0,1,0,0,0]	[1,1]	71	34000
p3	1	[1,0,0,0,0]	[0,1]	89	40000
p4	0	[0,0,0,1,0]	[0,0]	67	120000

Adopted from : https://introml.mit.edu

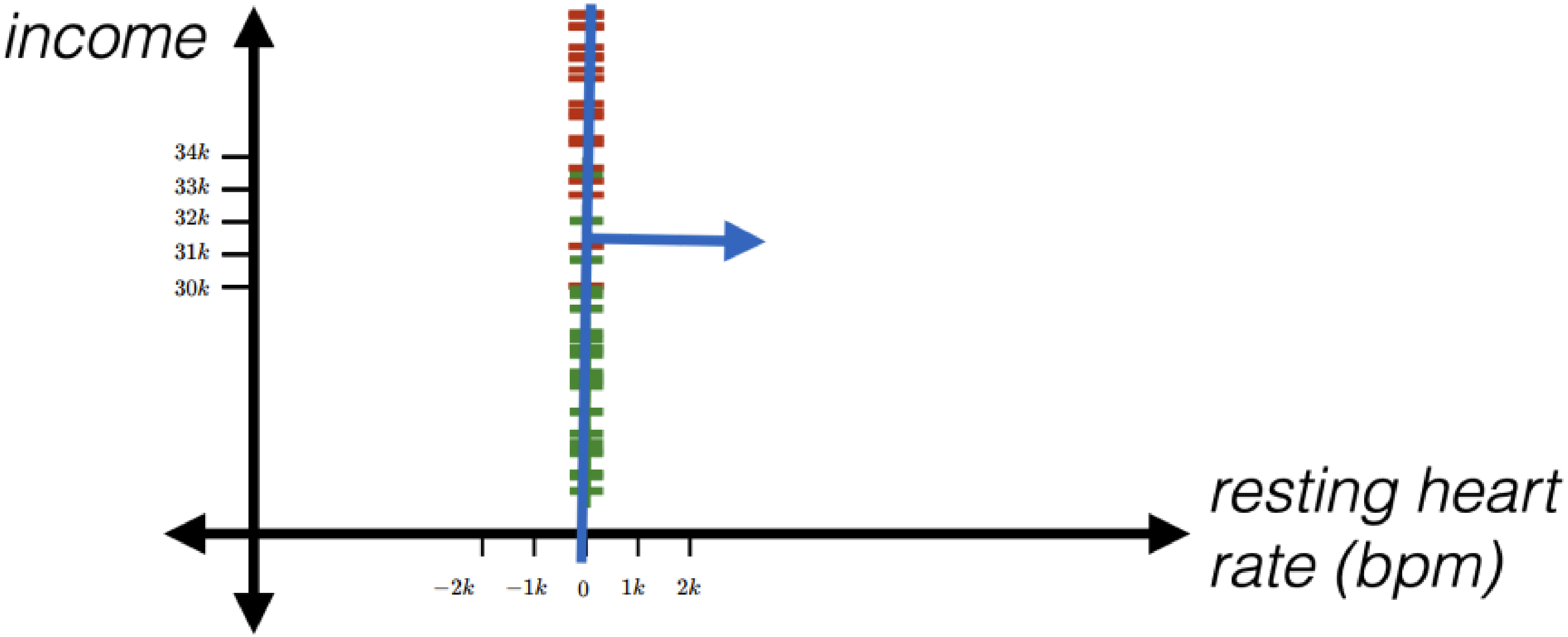
Outline

- Recap, linear models and beyond
- Systematic feature transformations
    - Polynomial features
    - Expressive power
- Hand-crafting features
    - One-hot
    - Factored
    - Standardization / normalization
    - Thermometer

Adopted from : https://introml.mit.edu

	resting heart rate (bpm)	family income (USD)
p1	55	133000
p2	71	34000
p3	89	40000
p4	67	120000

🥺

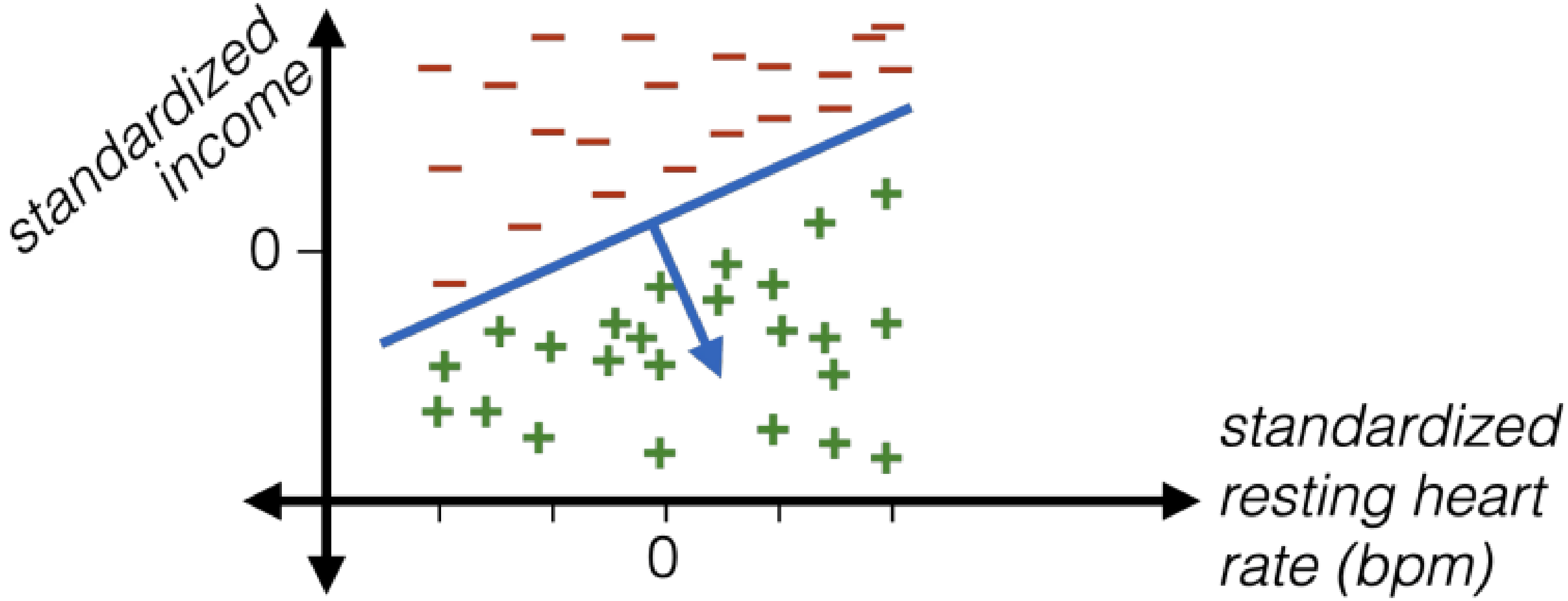


Adopted from : https://introml.mit.edu

- Idea: standardize numerical data. For ith feature and data point j:

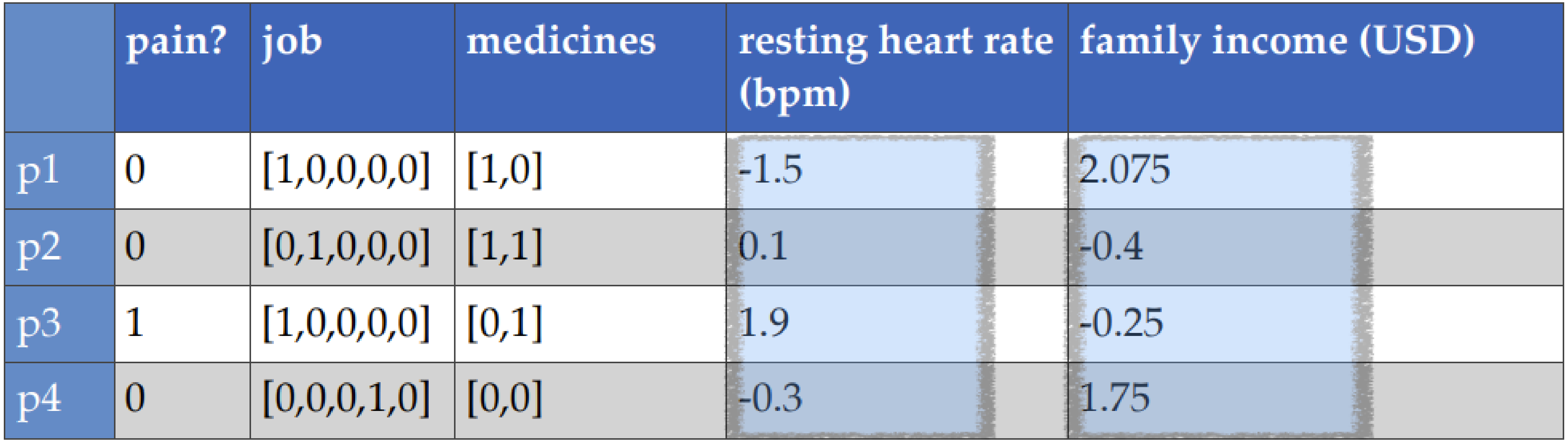
😍

$$\phi_i^{(j)} = \frac{x_i^{(j)} - \mathrm{mean}_i}{\mathrm{stddev}_i}$$



may also be easier to visualize and interpret learned parameters if we standardize data.

Adopted from : https://introml.mit.edu



	pain?	job	medicines	resting heart rate (bpm)	family income (USD)
p1	0	[1,0,0,0,0]	[1,0]	-1.5	2.075
p2	0	[0,1,0,0,0]	[1,1]	0.1	-0.4
p3	1	[1,0,0,0,0]	[0,1]	1.9	-0.25
p4	0	[0,0,0,1,0]	[0,0]	-0.3	1.75

Outline

- Recap, linear models and beyond
- Systematic feature transformations
  - Polynomial features
  - Expressive power
- Hand-crafting features
  - One-hot
  - Factored
  - Standardization/normalization
  - Thermometer

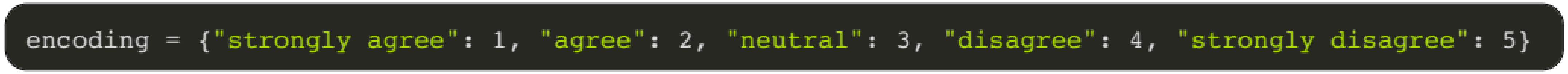
Imagine we added another question in survey: "how much do you agree that exercising could help preventing heart disease?"

	pain?	job	medicines	resting heart rate (bpm)	family income (USD)	agree exercising helps?
p1	0	[1,0,0,0,0]	[1,0]	-1.5	2.075	strongly disagree
p2	0	[0,1,0,0,0]	[1,1]	0.1	-0.4	disagree
p3	1	[1,0,0,0,0]	[0,1]	1.9	-0.25	neutral
p4	0	[0,0,0,1,0]	[0,0]	-0.3	1.75	agree

$$z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}} + \theta_{\substack{\text{deg of}\\\text{agreement}}} x_{\substack{\text{deg of}\\\text{agreement}}}$$

Adopted from : https://introml.mit.edu

🥺 For "degree of agreemenet", if use natural number encoding:



```
encoding = {"strongly agree": 1, "agree": 2, "neutral": 3, "disagree": 4, "strongly disagree": 5}
```

$$z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}} + \theta_{\substack{\text{deg of}\\\text{agreement}}} x_{\substack{\text{deg of}\\\text{agreement}}}$$

disagreed has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}} + 4\theta_{\substack{\text{deg of}\\\text{agreement}}}$

neutral has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}} + 3\theta_{\substack{\text{deg of}\\\text{agreement}}}$

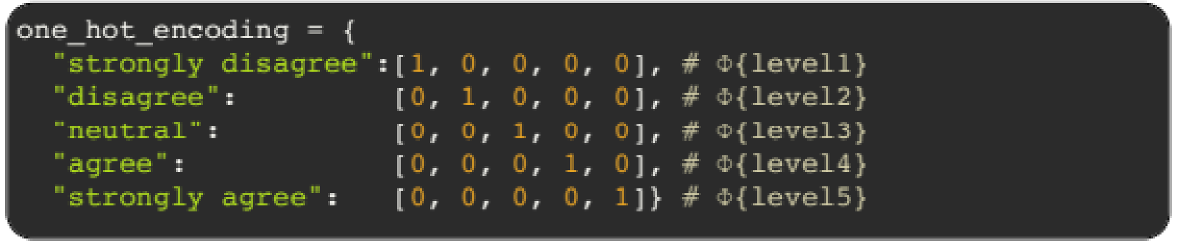
agreed has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\\text{rate}}} x_{\substack{\text{heart}\\\text{rate}}} + \theta_{\text{income}} x_{\text{income}} + \theta_{\substack{\text{deg of}\\\text{agreement}}}$

problem with this idea (again):

- Ordering matters
- Incremental in job category affects z by a fixed $\theta_{\substack{\text{deg of}\\\text{agreement}}}$ amount

Adopted from : https://introml.mit.edu

🥺



```
one_hot_encoding = {
  "strongly disagree":[1, 0, 0, 0, 0], # Φ{level1}
  "disagree":         [0, 1, 0, 0, 0], # Φ{level2}
  "neutral":          [0, 0, 1, 0, 0], # Φ{level3}
  "agree":            [0, 0, 0, 1, 0], # Φ{level4}
  "strongly agree":   [0, 0, 0, 0, 1]} # Φ{level5}
```

$$z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\ \text{rate}}} x_{\substack{\text{heart}\\ \text{rate}}} + \theta_{\text{income}} x_{\text{income}} + \underline{\theta_{\substack{\text{deg of}\\ \text{agreement}}} x_{\substack{\text{deg of}\\ \text{agreement}}}}$$

$$\theta_{\text{level1}} \phi_{\text{level1}} + \theta_{\text{level2}} \phi_{\text{level2}} + \theta_{\text{level3}} \phi_{\text{level3}} + \theta_{\text{level4}} \phi_{\text{level4}} + \theta_{\text{level5}} \phi_{\text{level5}}$$

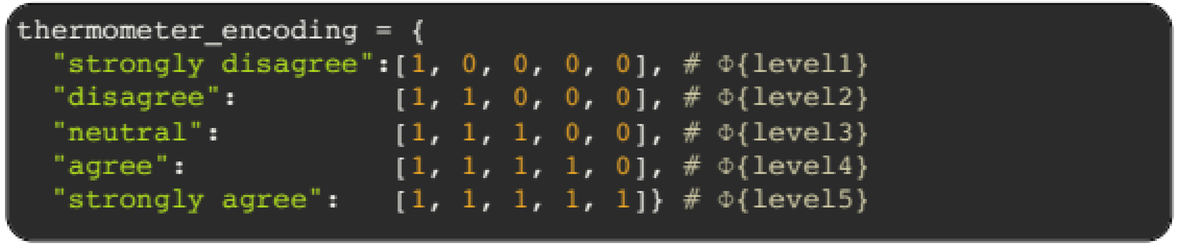
disagreed has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\ \text{rate}}} x_{\substack{\text{heart}\\ \text{rate}}} + \theta_{\text{income}} x_{\text{income}} + \theta_{\text{level2}}$

neutral has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\ \text{rate}}} x_{\substack{\text{heart}\\ \text{rate}}} + \theta_{\text{income}} x_{\text{income}} + \theta_{\text{level3}}$

agreed has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\ \text{rate}}} x_{\substack{\text{heart}\\ \text{rate}}} + \theta_{\text{income}} x_{\text{income}} + \theta_{\text{level4}}$

Adopted from : https://introml.mit.edu

😍



```
thermometer_encoding = {
  "strongly disagree":[1, 0, 0, 0, 0], # Φ{level1}
  "disagree":         [1, 1, 0, 0, 0], # Φ{level2}
  "neutral":          [1, 1, 1, 0, 0], # Φ{level3}
  "agree":            [1, 1, 1, 1, 0], # Φ{level4}
  "strongly agree":   [1, 1, 1, 1, 1]} # Φ{level5}
```

$$z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\ \text{rate}}} x_{\substack{\text{heart}\\ \text{rate}}} + \theta_{\text{income}} x_{\text{income}} + \underbrace{\theta_{\substack{\text{deg of}\\ \text{agreement}}} x_{\substack{\text{deg of}\\ \text{agreement}}}}$$

$$\theta_{\text{level1}} \phi_{\text{level1}} + \theta_{\text{level2}} \phi_{\text{level2}} + \theta_{\text{level3}} \phi_{\text{level3}} + \theta_{\text{level4}} \phi_{\text{level4}} + \theta_{\text{level5}} \phi_{\text{level5}}$$

disagreed has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\ \text{rate}}} x_{\substack{\text{heart}\\ \text{rate}}} + \theta_{\text{income}} x_{\text{income}} + (\theta_{\text{level1}} + \theta_{\text{level2}})$

neutral has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\ \text{rate}}} x_{\substack{\text{heart}\\ \text{rate}}} + \theta_{\text{income}} x_{\text{income}} + (\theta_{\text{level1}} + \theta_{\text{level2}} + \theta_{\text{level3}})$

agreed has $z = \theta_{\text{pain}} x_{\text{pain}} + \theta_{\text{job}}^T x_{\text{job}} + \theta_{\text{pill}}^T x_{\text{pill}} + \theta_{\substack{\text{heart}\\ \text{rate}}} x_{\substack{\text{heart}\\ \text{rate}}} + \theta_{\text{income}} x_{\text{income}}$

$$+(\theta_{\text{level1}} + \theta_{\text{level2}} + \theta_{\text{level3}} + \theta_{\text{level4}})$$

Adopted from : https://introml.mit.edu

Summary

- Linear models are mathematically and algorithmically convenient but not expressive enough -- by themselves -- for most jobs.
- We can express really rich hypothesis classes by performing a **fixed** non-linear feature transformation first, then applying our linear (regression or classification) methods.
- When we “set up” a problem to apply ML methods to it, it’s important to encode the inputs in a way that makes it easier for the ML method to exploit the structure.
- Foreshadowing of neural networks, in which we will learn complicated continuous feature transformations.