Machine Learning in Healthcare

#L09-Introduction to nonlinear models

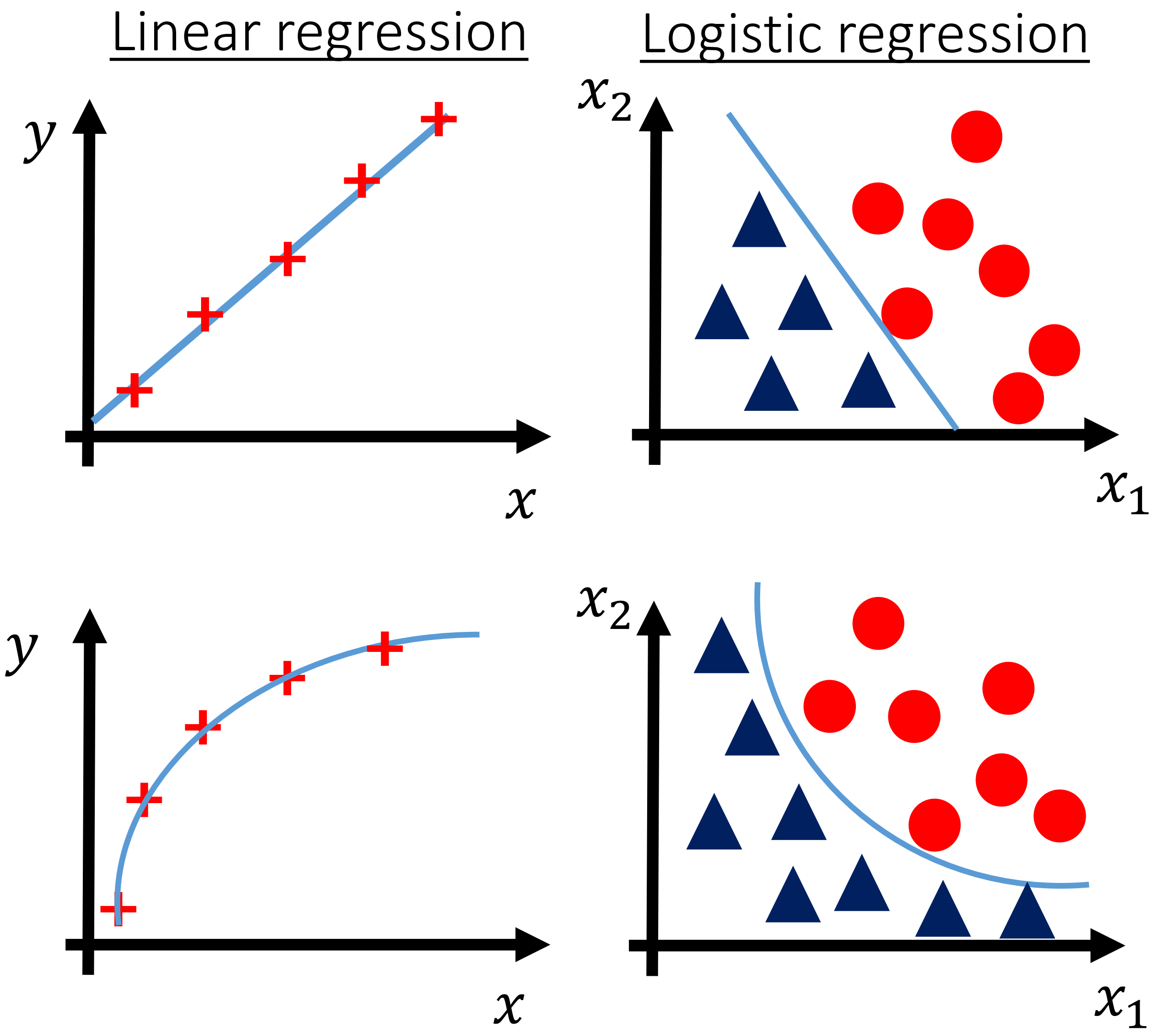
Technion-IIT, Haifa, Israel

Asst. Prof. Joachim Behar
Biomedical Engineering Faculty, Technion-IIT
Artificial intelligence in medicine laboratory (AIMLab.)
https://aim-lab.github.io/
Twitter: @lab_aim

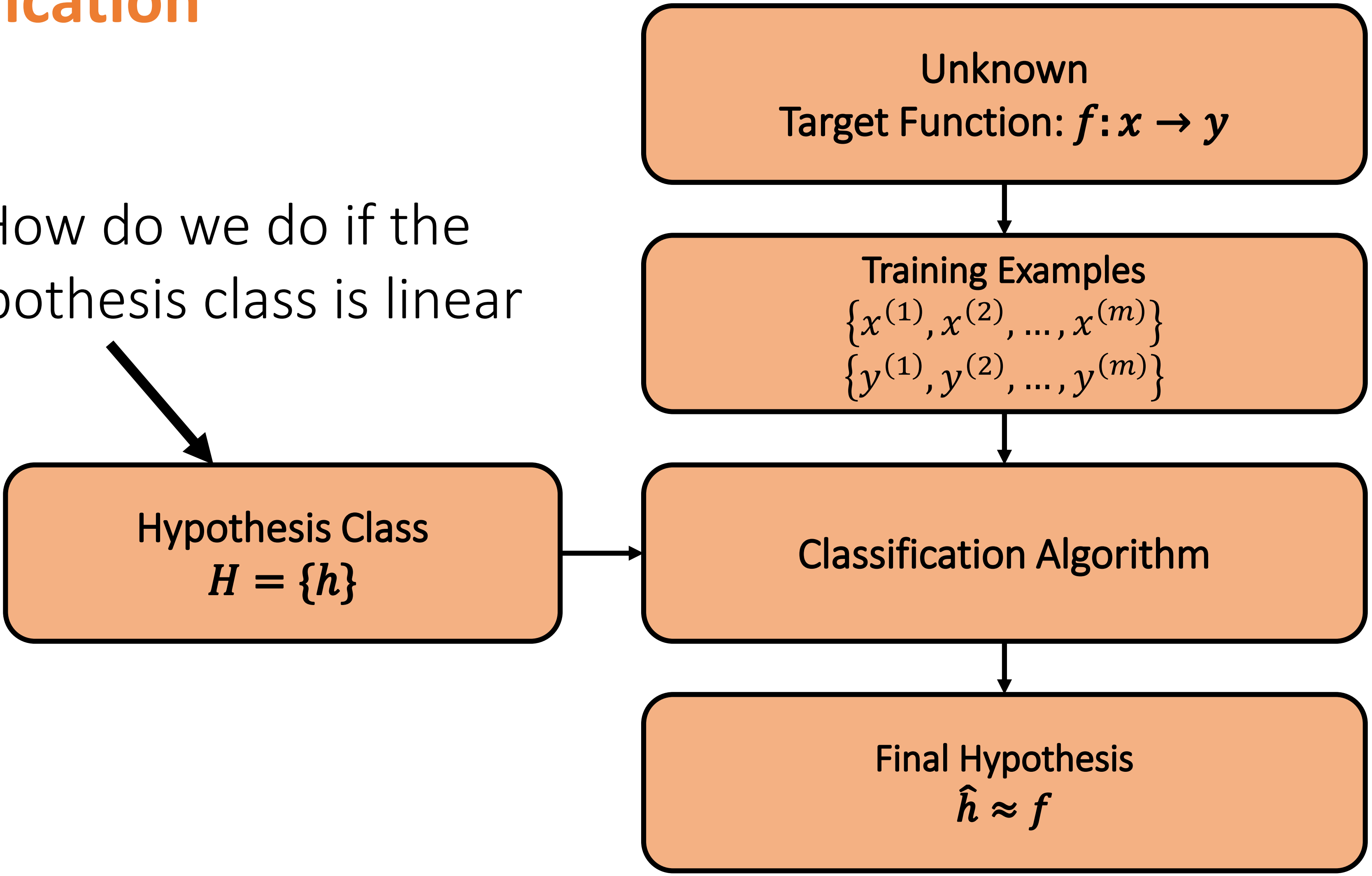
From linear to nonlinear

Hypothesis representation

- Linear regression:
    - Hypothesis class: $h_w(x) = w^T x$
- Logistic regression:
    - Hypothesis class: $h_w(x) = \sigma(w^T x)$
- However life is often nonlinear.
    - How do we learn nonlinear models?



Classification



Unknown
Target Function: $\boldsymbol{f : x \rightarrow y}$

Training Examples
$\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$
$\{y^{(1)}, y^{(2)}, \dots, y^{(m)}\}$

How do we do if the hypothesis class is linear

Hypothesis Class
$\boldsymbol{H = \{h\}}$

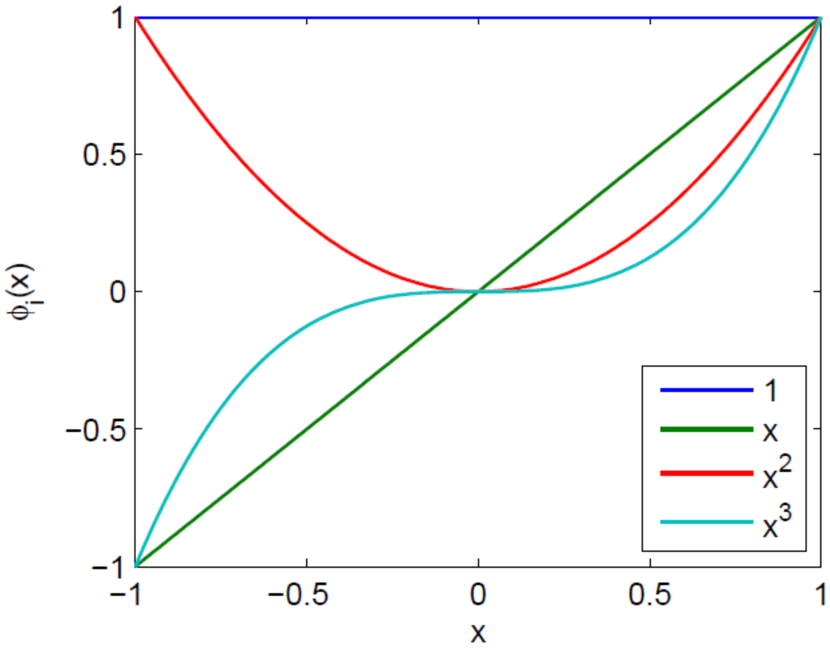
Classification Algorithm

Final Hypothesis
$\boldsymbol{\hat{h} \approx f}$

Using explicit nonlinear features

Linear regression but with nonlinear features

- We may construct explicit feature vectors:
  - Example 1:
    - $x \in \mathbb{R}^2, h_w(x) = w^T x = w_1 x_1 + w_2 x_2$
    - $x \in \mathbb{R}^2 \rightarrow x' \in \mathbb{R}^5$
    - $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \rightarrow \begin{bmatrix} x_1 \\ x_2 \\ x_1^2 \\ x_2^2 \\ x_1 x_2 \end{bmatrix} \begin{matrix} \\ \\ \left.\begin{matrix} \\ \\ \end{matrix}\right\} \text{Higher power} \\ \left.\right\} \text{Interaction} \end{matrix}$
    - $h_w(x') = w^T x' = w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2 + w_4 x_1 x_2$



Plot of polynomial bases

Image credit: Stefano Ermon

Linear regression but with nonlinear features

- We may construct explicit feature vectors:
  - Example 2:

Linear regression but with nonlinear features

- General :
  - Moving to a new space where the examples may be linearly separable.
  - $\begin{cases} \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n'_x} \\ \emptyset : x \rightarrow x' \end{cases}$
  - $h_w(x') = w^T x'$

Logistic regression but with nonlinear features

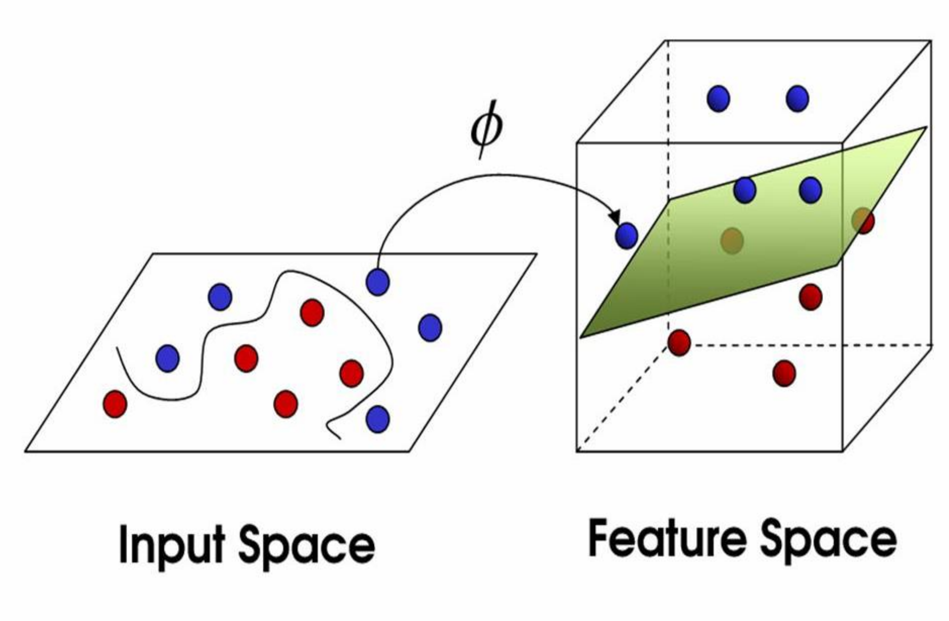
- We may construct explicit feature vectors:
  - Example 1:
    - $x \in \mathbb{R}^2, h_w(x) = \sigma(w^T x) = \sigma(w_1 x_1 + w_2 x_2)$
    - $x \in \mathbb{R}^2 \ \rightarrow x' \in \mathbb{R}^5$
    - $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \rightarrow \begin{bmatrix} x_1 \\ x_2 \\ x_1^2 \\ x_2^2 \\ x_1 x_2 \end{bmatrix}$
    - $h_w(x') = \sigma(w^T x') = \sigma(w_1 x_1 + w_2 x_2 + w_3 x_1^2 + w_4 x_2^2 + w_4 x_1 x_2)$

Logistic regression but with nonlinear features

- We may construct explicit feature vectors:
  - Example 2:

Logistic regression but with nonlinear features

- General :
  - Moving to a new space where the examples may be linearly separable.
  - $\begin{cases} \mathbb{R}^{n_x} \rightarrow \mathbb{R}^{n'_x} \\ \emptyset : x \rightarrow x' \end{cases}$
  - $h_w(x') = \sigma(w^T x')$



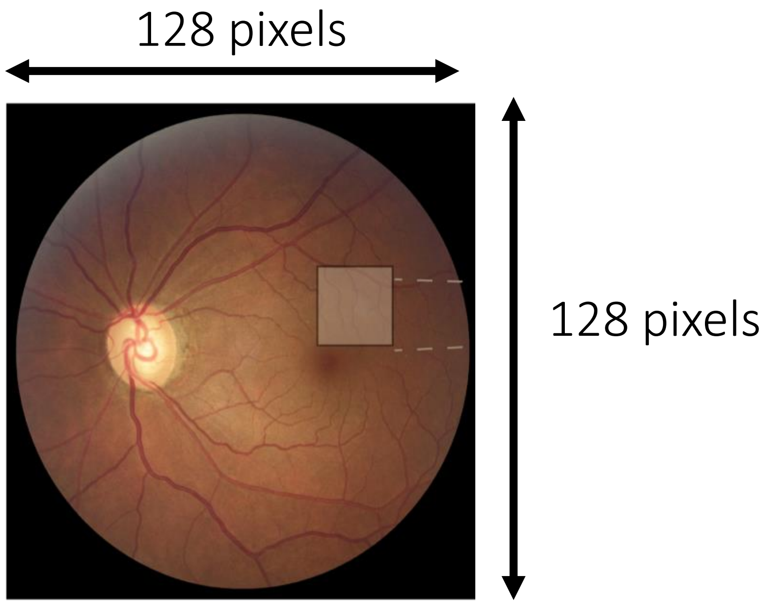
From linear to nonlinear

- So great, it seems that we are done with this lecture, right?
- We know how to perform linear and nonlinear regression/classification.
- Yes, but not quiet…
- What happens if we have many features, i.e. n_x is large?

n_x	d	Complexity	n_x'
60	2	o(2)	1,891
60	3	?	?
60	4	?	?
60	5	?	?

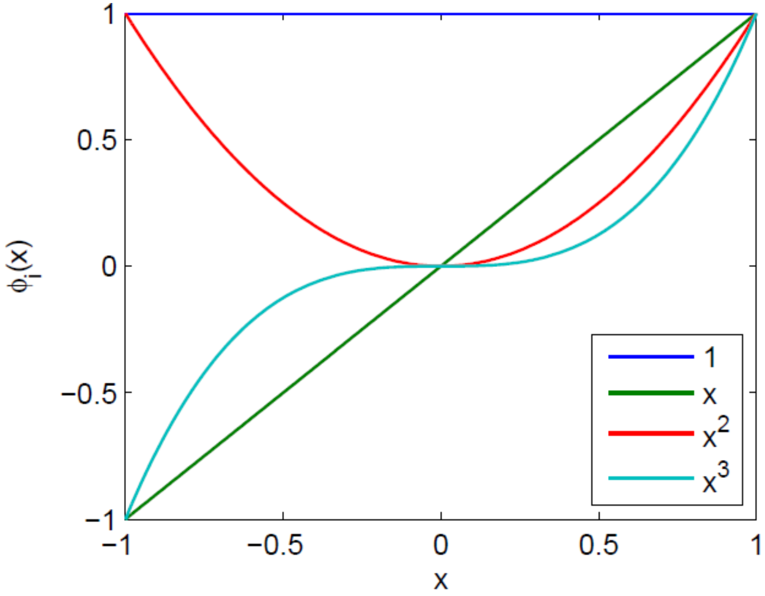
From linear to nonlinear

- More generally, for a choice of degree d and an input vector with n_x features then the feature transformation will lead to:
    - $n'_x = \binom{n_x+d}{d}$
- That is a lot of features to compute for every single example!
- In particular, take an image:
    - 128*128=16384 pixels.
    - That is a lot of features.



Using another basis function

- We gave examples using the polynomial features i.e. a polynomial basis function.
- We could also use alternative basis function such as the radial basis function (RBF):
  - $\emptyset_j(x) = exp\left(\frac{-\|x-\mu_j\|^2}{2\sigma^2}\right)$
  - With $\boldsymbol{\sigma}$ the bandwidth and $\mu_j$the center of the RBF kernel and $j \in [\![1: n'_x]\!]$.



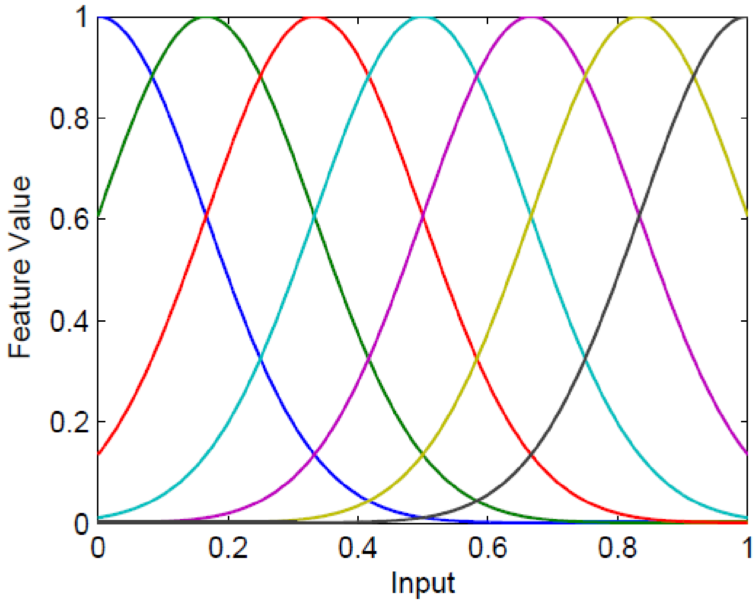
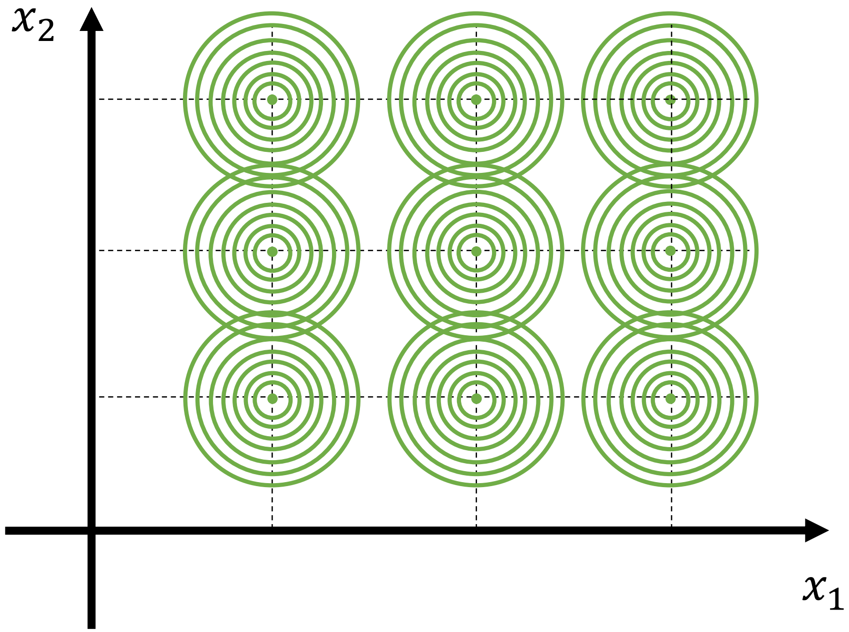


Image credit: Stefano Ermon

Using another basis function

- Complexity: if we assume we make a uniform grid over the input space with d centers along each dimension then:
  - $n_x' = d^{n_x}$
- Example for $n_x = 2$:



From linear to nonlinear

- To summarize:
  - We can use our linear model for nonlinear classification by using a nonlinear transform of the features from the input space and by working in the new space.
  - If you have a few features and a good idea of what transformation might be relevant then that approach might work well.
  - However, if you have a high number of features and/or no prior on a meaningful transformation then this approach is computationally expensive and might lead to overfitting.

Take home

- It is possible to perform nonlinear regression/classification using a linear model by constructing an explicit nonlinear features vector and then performing linear regression in the new feature space.
- Different basis function may be used for this transform e.g. polynomial, RBF.
- However, complexity is high and there is a high risk of overfitting. So this might only be a viable option for a limited number of input features and with some prior about what feature transform may be interesting.
- How do we deal with nonlinear regression and classification otherwise? Next lecture.

References

[1] Lecture notes: Machine Learning 2: Nonlinear Regression. By Stefano Ermon, 2016

URL (access date 20-11-2020): https://cs.stanford.edu/~ermon/cs325/slides/ml_nonlin_reg.pdf

Other good read:

https://kenndanielso.github.io/mlrefined/blog_posts/12_Nonlinear_intro/12_2_Introduction_nonlinear_classification.html