Machine Learning: A Practical Approach to the Statistical Learning Theory

Rodrigo Fernandes de Mello

Associate Professor

Universidade de São Paulo

Instituto de Ciências Matemáticas e de Computação

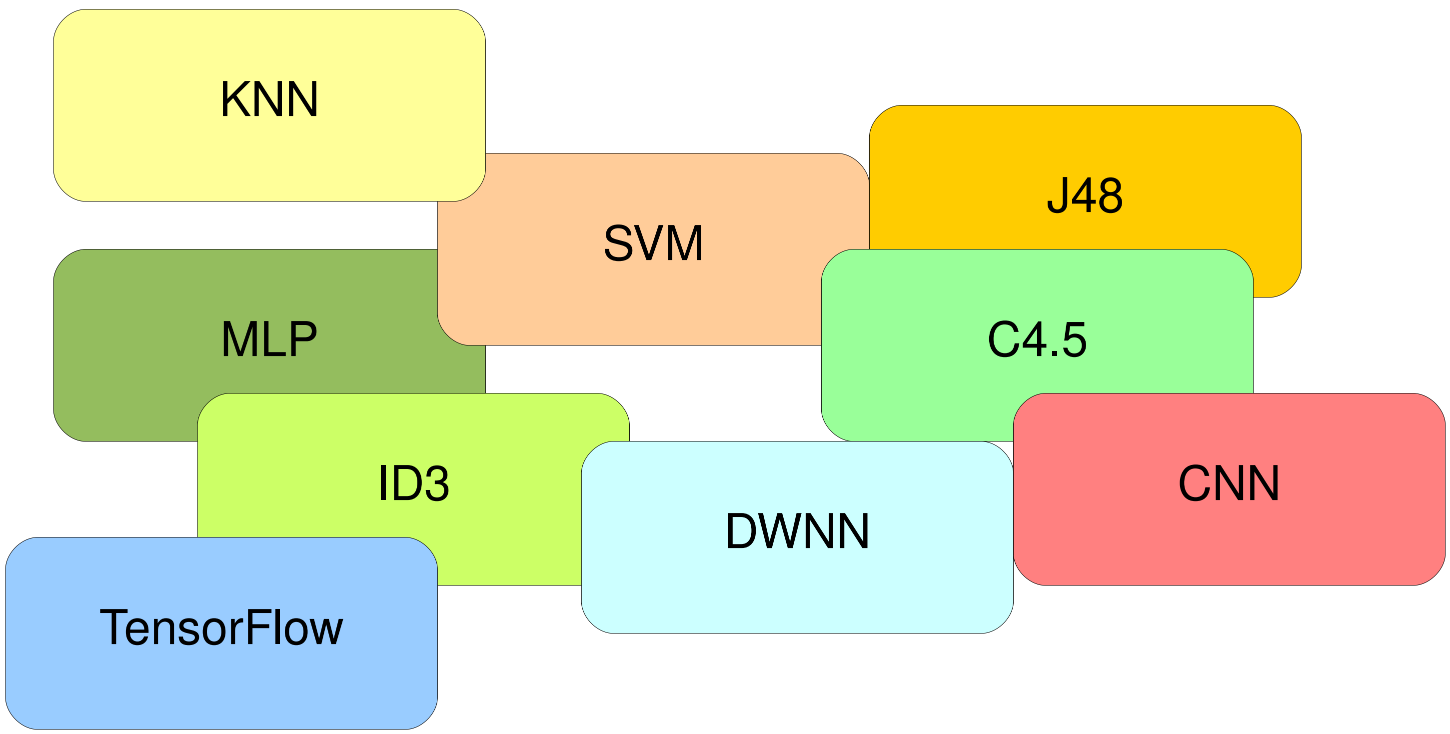
mello@icmc.usp.br

August 7th, 2019

Basics

- Machine Learning
  - Supervised Learning
    - Theoretical Learning Bounds
    - Relies on the Statistical Learning Theory
  - Unsupervised Learning
    - Still lacks in Theoretical Bounds

- So many classification algorithms:
  - How can we assess any of those?
    - K-fold cross validation, leave-one-out, ...
  - How can we prove any of those "learn"?



- First of all, what is "learning" in our context?
  - Concept of Generalization by Vapnik

Sample error

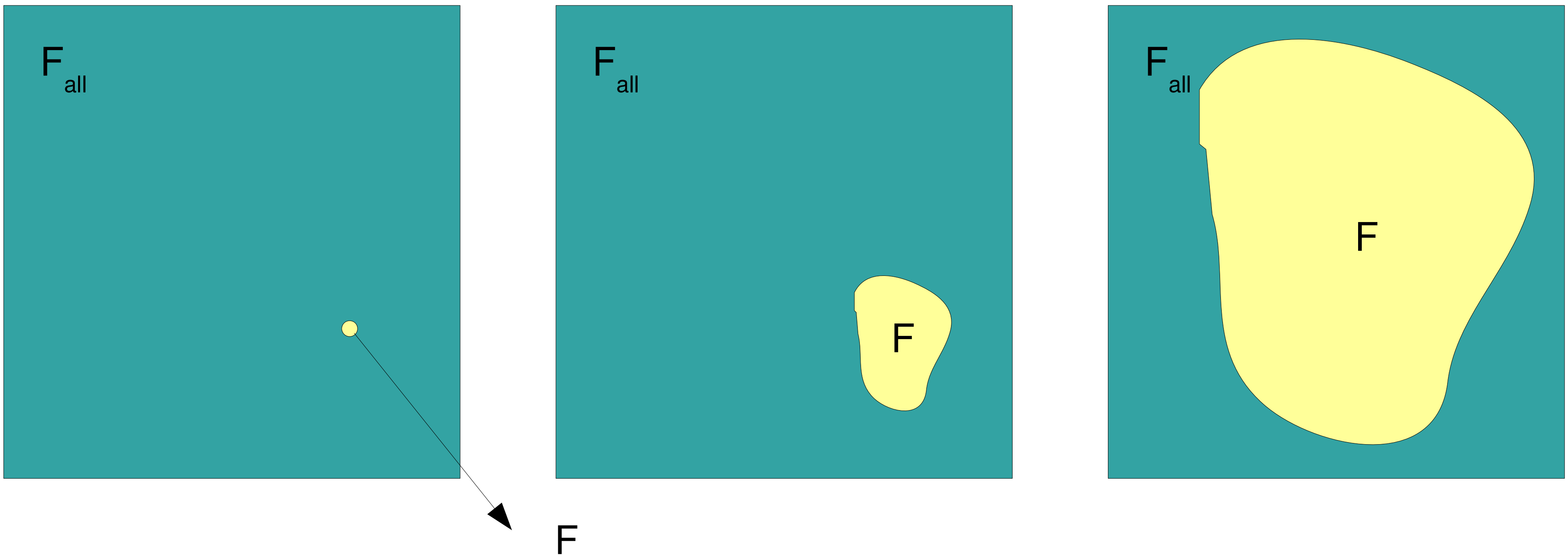
Expected error

$$G = |R_{\text{emp}}(f) - R(f)|$$

This is the concept of Generalization

- In addition, the Empirical Risk must be as small as possible

How that is mapped to the Statistical Learning Theory?

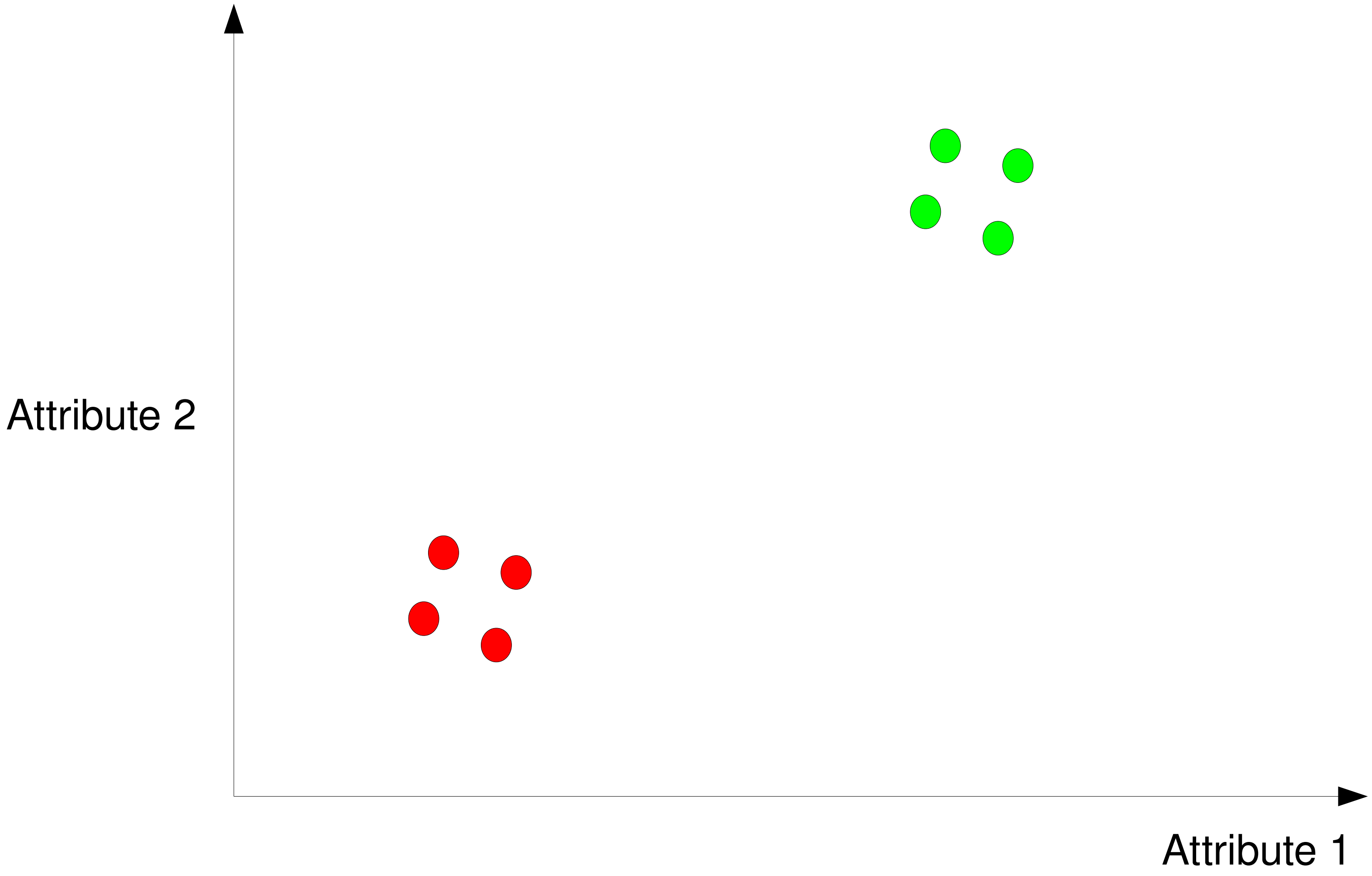


Three examples of algorithm biases

Using the Distance-Weighted Nearest Neighbors to illustrate algorithm biases

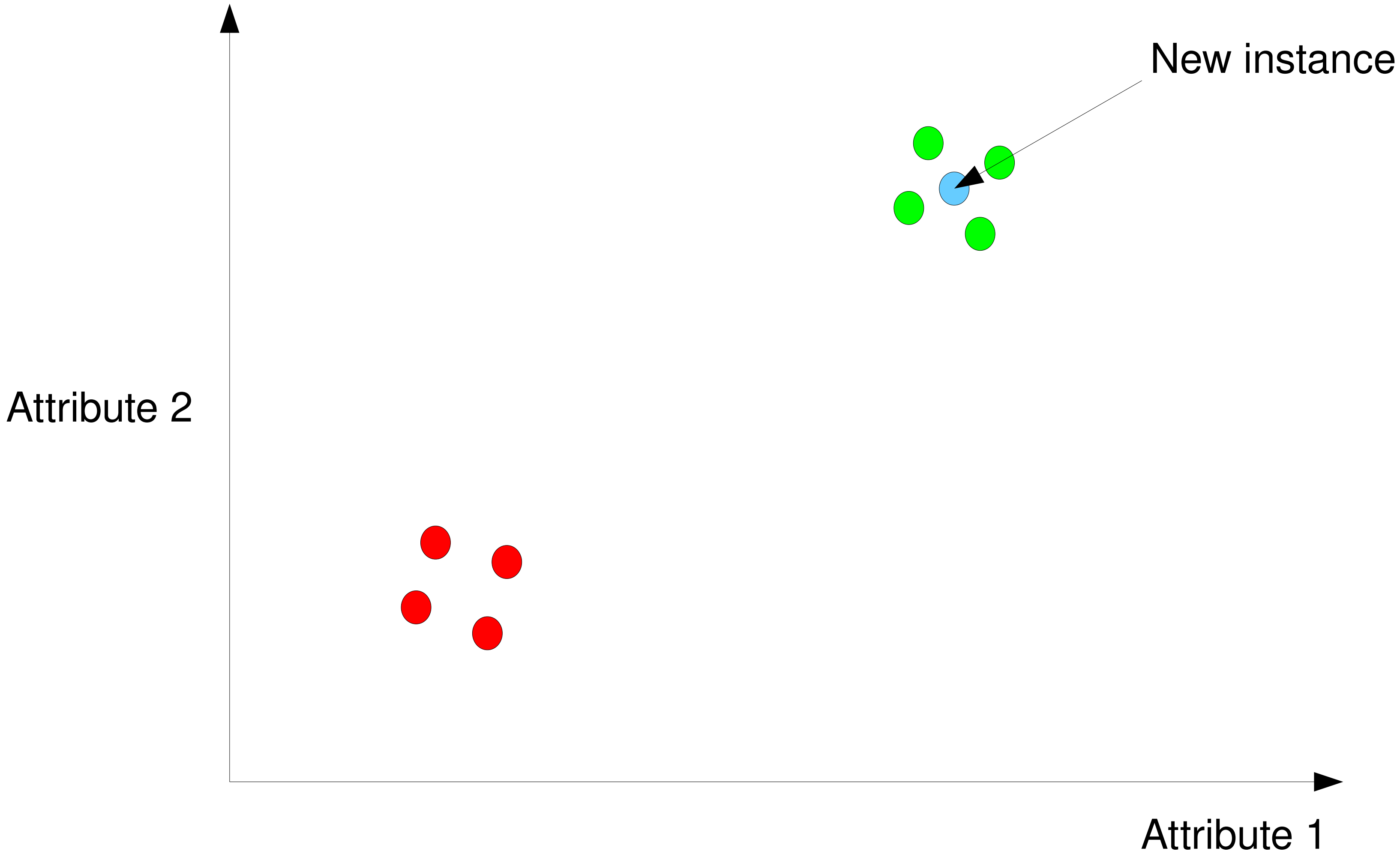
Distance-Weighted Nearest Neighbors

- Based on the same principles as the k-Nearest Neighbors



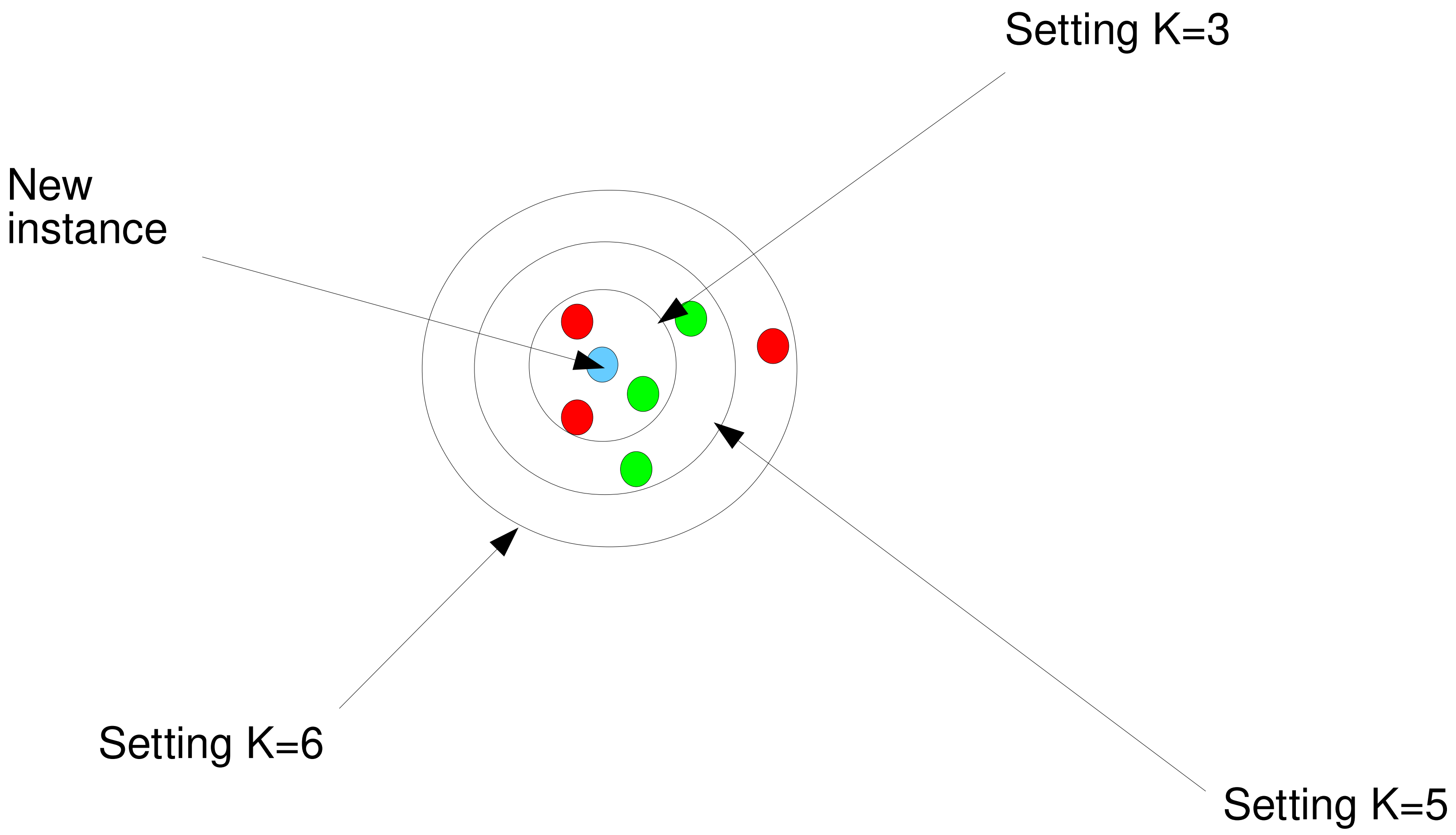
Distance-Weighted Nearest Neighbors

- Based on the same principles as the k-Nearest Neighbors



Distance-Weighted Nearest Neighbors

- Based on the same principles as the k-Nearest Neighbors



Distance-Weighted Nearest Neighbors

- It is based on Radial functions centered at the new instance a.k.a. query point

- Classification output:

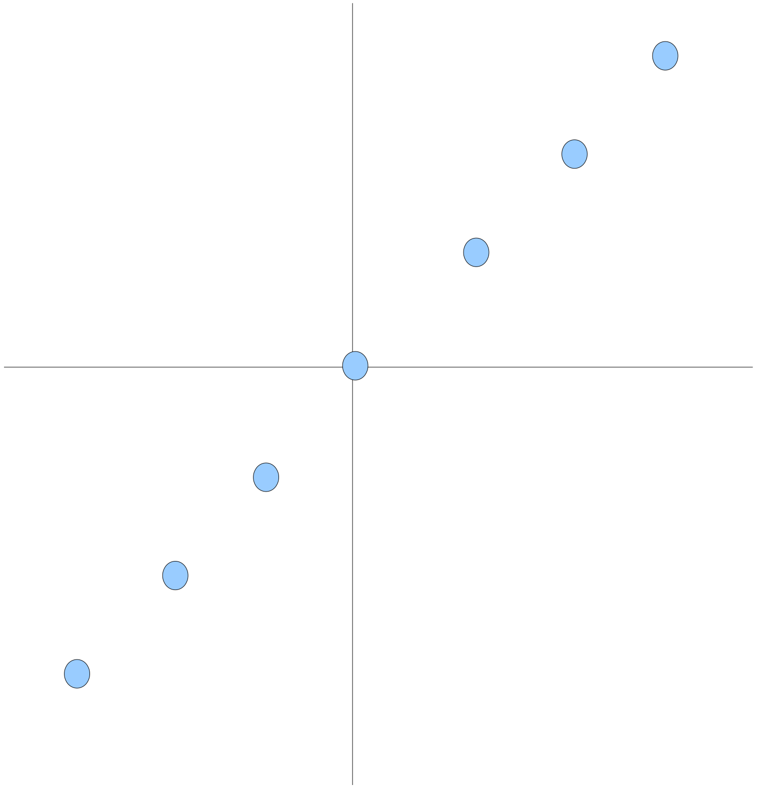
$$f(\mathbf{x}) = \frac{\sum_{i=1}^{n} w_i y_i}{\sum_{i=1}^{n} w_i}$$

- Given the weight function:

$$w_i = \exp - \frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{2\sigma^2}$$

Distance-Weighted Nearest Neighbors

- After implementing, test it on this simple example of an identity function:



Two main questions:

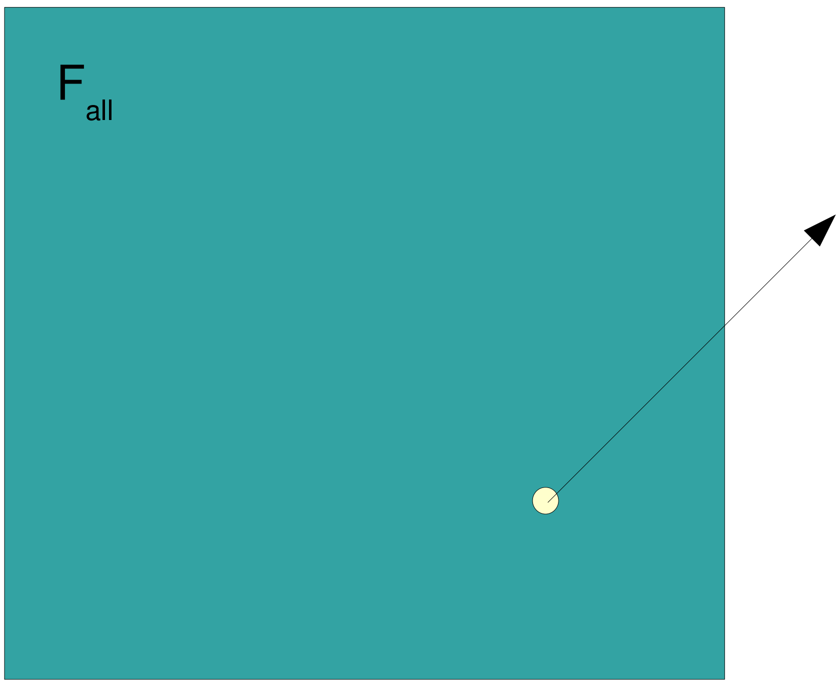
- What happens if sigma is too big?

- What happens if sigma is too small?

So, how can we define the best value for sigma?

Distance-Weighted Nearest Neighbors

- When sigma tends to infinity

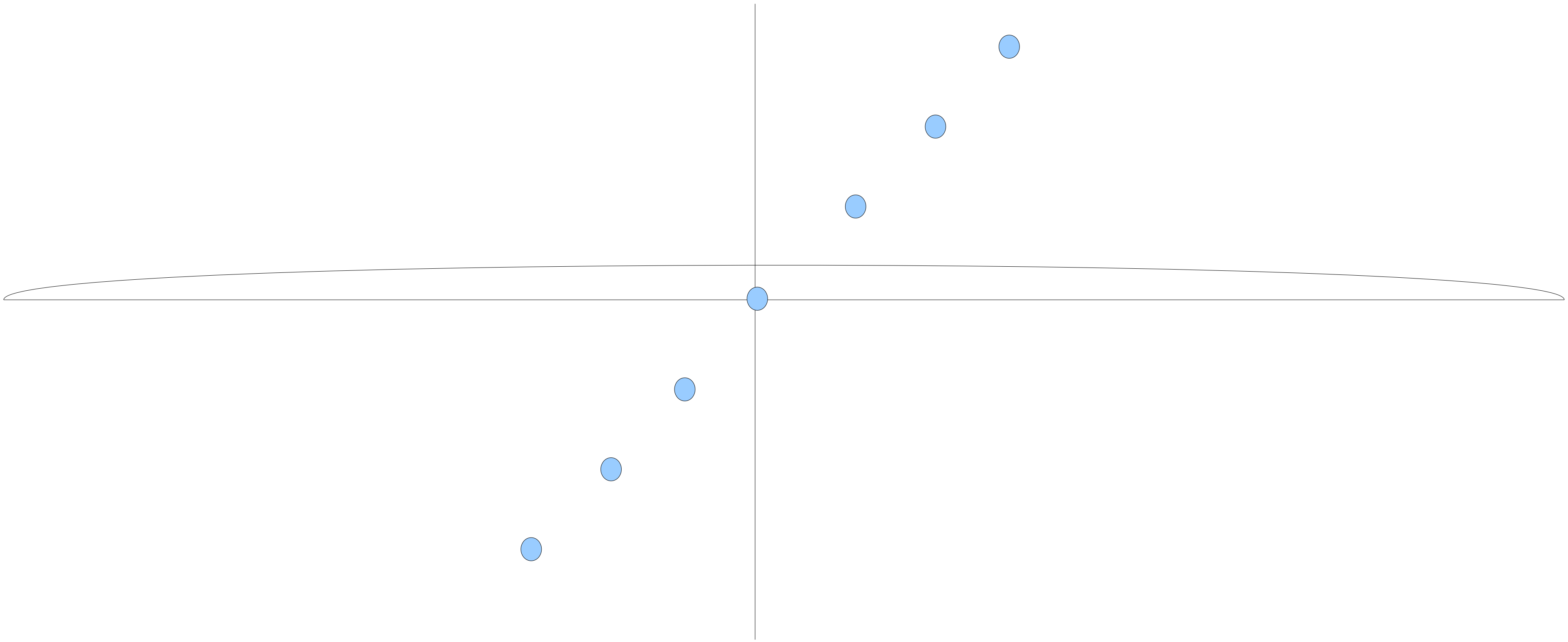


The space of admissible functions (bias) will contain a single function

In this case, the average function

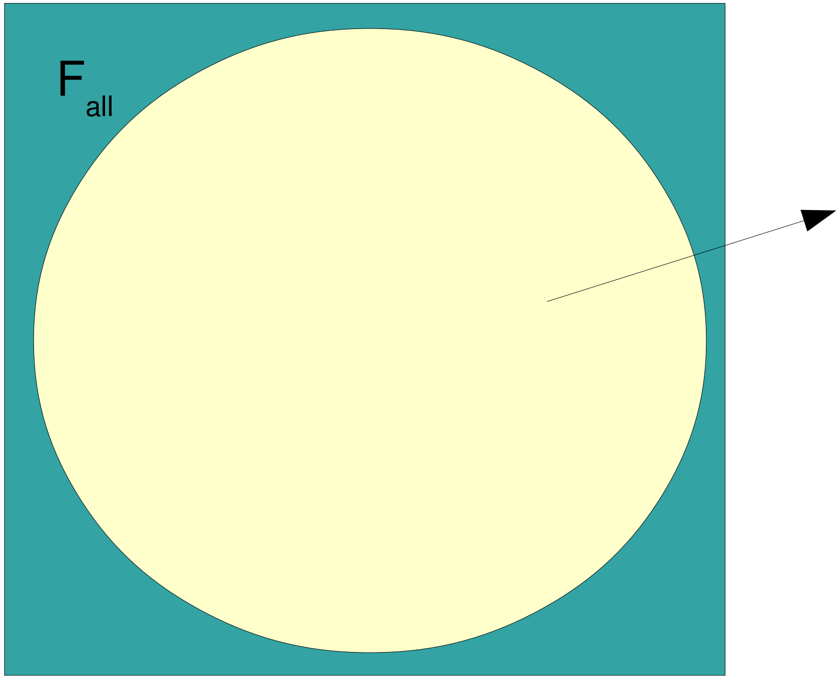
Distance-Weighted Nearest Neighbors

- When sigma tends to infinity



Distance-Weighted Nearest Neighbors

- When sigma tends to zero



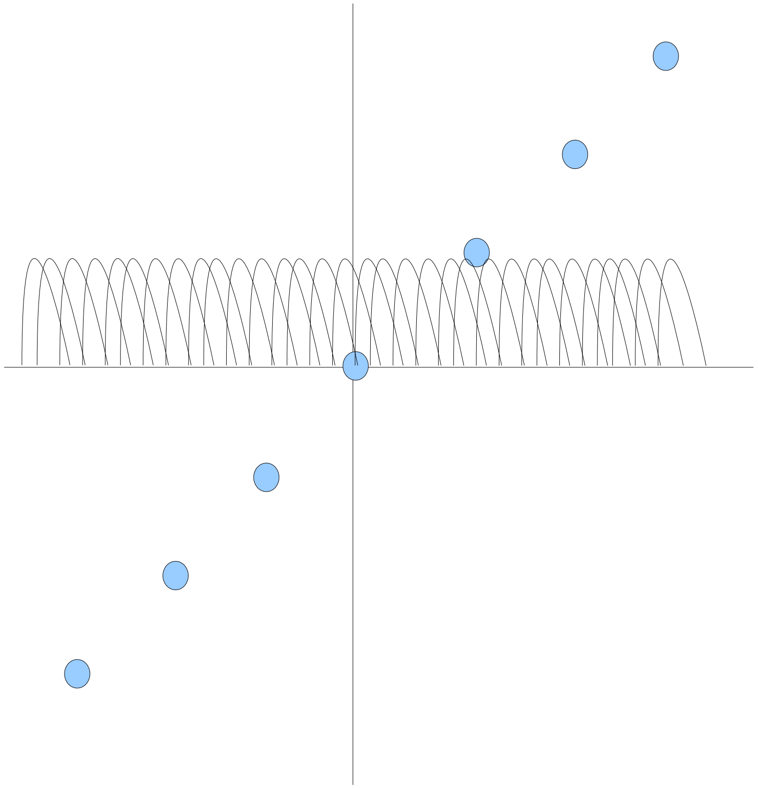
The space of admissible functions (bias) will tend to the whole space

What is the problem with that?

It will most probably contain at least one memory-based classifier

Distance-Weighted Nearest Neighbors

- When sigma tends to zero



Understanding the basics about SLT

- Vapnik formulated the great part of the Statistical Learning Theory
  - His basic idea was to prove how some supervised algorithm “learns”
  - That required some formalization
    - Concept of Generalization
    - Reduce the empirical risk as more examples are sampled
  - Took advantage of the Law of Large Numbers

Understanding the basics about SLT

- Vapnik took advantage of the Law of Large Numbers:

$$P\left(\left|\frac{1}{n}\sum_{i=1}^{n}\xi_i - E(\xi)\right| > \epsilon\right) \leq 2\exp(-2n\epsilon^2)$$

- But that works if and only if:
  - Data examples are independent from each other
  - Data examples must be sampled in an independent manner
  - The function to be estimated is independent of data
  - The data distribution must be fixed/static
    - It cannot change along time
- Main advantage:
  - We have an upper bound → lets see it!

Understanding the basics about SLT

- So, from the Law of Large Numbers:

$$P\left(\left|\frac{1}{n}\sum_{i=1}^{n}\xi_i - E(\xi)\right| > \epsilon\right) \leq 2\exp(-2n\epsilon^2)$$

- He defined the following:

$$P(|R_{\text{emp}}(f) - R(f)| > \epsilon) \leq 2\exp(-2n\epsilon^2)$$

- In which:

$$R_{\text{emp}}(f) = \frac{1}{n}\sum_{i=1}^{n}\ell(X_i, Y_i, f(X_i))$$

$$R(f) = E(\ell(X, Y, f(X))$$

Understanding the basics about SLT

- So, from the Law of Large Numbers:

$$P\left(\left|\frac{1}{n}\sum_{i=1}^{n}\xi_i - E(\xi)\right| > \epsilon\right) \leq 2\exp(-2n\epsilon^2)$$

- He defined the following:

$$P(|R_{\text{emp}}(f) - R(f)| > \epsilon) \leq 2\exp(-2n\epsilon^2)$$

- In which:

But that is valid for a single predefined function f

Understanding the basics about SLT

- Vapnik rewrote:

$$P(|R_{\mathrm{emp}}(f) - R(f)| > \epsilon) \leq 2\exp(-2n\epsilon^2)$$

- So, for all functions contained in the algorithm bias:

$$P\Big(|R(f_1) - R_{\mathrm{emp}}(f_1)| > \varepsilon \ \text{ or } \ |R(f_2) - R_{\mathrm{emp}}(f_2)| > \varepsilon \ \text{ or } \ \ldots \ \text{ or } \ |R(f_m) - R_{\mathrm{emp}}(f_m)| > \varepsilon \Big)$$

Understanding the basics about SLT

- Vapnik rewrote:

$$P(|R_{\text{emp}}(f) - R(f)| > \epsilon) \leq 2\exp(-2n\epsilon^2)$$

- So, for all functions contained in the algorithm bias:

$$P\Big(|R(f_1) - R_{\text{emp}}(f_1)| > \varepsilon \ \text{ or } \ |R(f_2) - R_{\text{emp}}(f_2)| > \varepsilon \ \text{ or } \ \dots \ \text{ or } \ |R(f_m) - R_{\text{emp}}(f_m)| > \varepsilon \Big)$$

- What is bounded as follows:

$$P\Big(|R(f_1) - R_{\text{emp}}(f_1)| > \varepsilon \ \text{ or } \ |R(f_2) - R_{\text{emp}}(f_2)| > \varepsilon \ \text{ or } \ \dots \ \text{ or } \ |R(f_m) - R_{\text{emp}}(f_m)| > \varepsilon \Big)$$
$$\leq \sum_{i=1}^{m} P(|R(f_i) - R_{\text{emp}}(f_i)| > \varepsilon)$$

Understanding the basics about SLT

- Vapnik rewrote:

$$P(|R_{\text{emp}}(f) - R(f)| > \epsilon) \leq 2\exp(-2n\epsilon^2)$$

- So

P

That is not difficult to see...

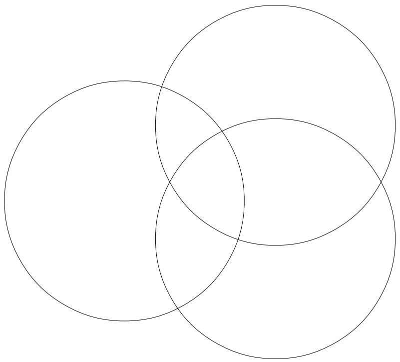
- What is bounded as

$$P\Big(|R(f_1) - R_{\text{emp}}(f_1)| > \varepsilon \;\text{ or }\; |R(f_2) - R_{\text{emp}}(f_2)| > \varepsilon \;\text{ or }\; \ldots \;\text{ or }\; |R(f_m) - R_{\text{emp}}(f_m)| > \varepsilon \Big)$$

$$\leq \sum_{i=1}^{m} P(|R(f_i) - R_{\text{emp}}(f_i)| > \varepsilon)$$

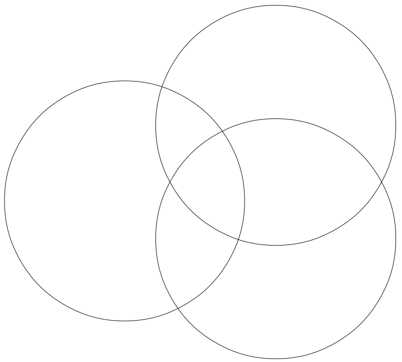
Understanding the basics about SLT

- Suppose we have a set of sets, which could intersect or not:
  - If they intersect:

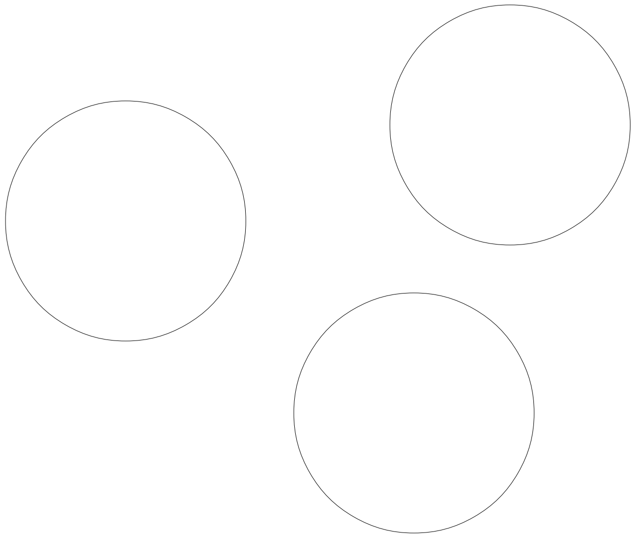


Understanding the basics about SLT

- Suppose we have a set of sets, which could intersect or not:
  - If they intersect:



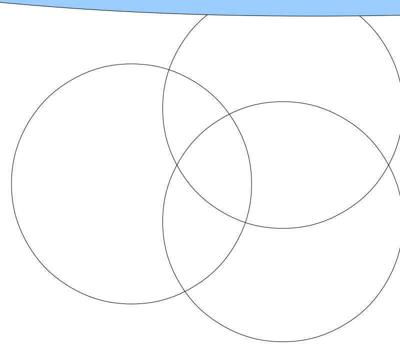
  - If they do not



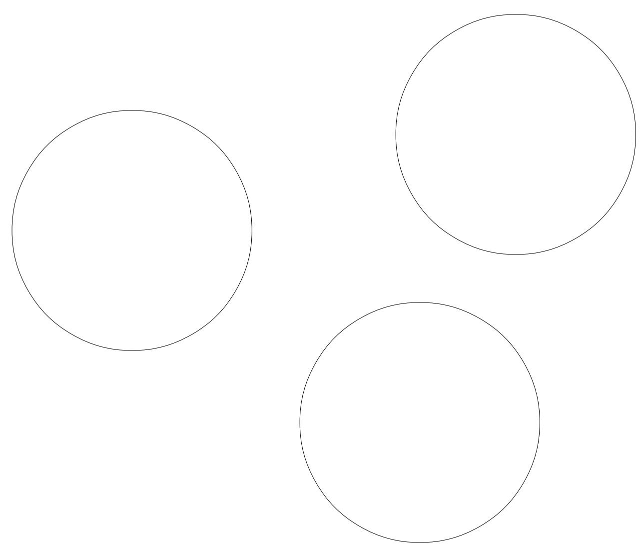
Understanding the basics about SLT

- Supp
  - I

The sum of the union of sets is always smaller than or equal to when they do not intersect!



  - If they do not



Understanding the basics about SLT

- So, for all functions within the algorithm bias:

$$P\Big(|R(f_1) - R_{\text{emp}}(f_1)| > \varepsilon \ \text{ or } \ |R(f_2) - R_{\text{emp}}(f_2)| > \varepsilon \ \text{ or } \ \ldots \ \text{ or } \ |R(f_m) - R_{\text{emp}}(f_m)| > \varepsilon \Big)$$

$$\leq \sum_{i=1}^{m} P(|R(f_i) - R_{\text{emp}}(f_i)| > \varepsilon)$$

Understanding the basics about SLT

- So, for all functions within the algorithm bias:

$$P\Big(|R(f_1) - R_{\text{emp}}(f_1)| > \varepsilon \ \text{ or } \ |R(f_2) - R_{\text{emp}}(f_2)| > \varepsilon \ \text{ or } \ \ldots \ \text{ or } \ |R(f_m) - R_{\text{emp}}(f_m)| > \varepsilon \Big)$$

$$\leq \sum_{i=1}^{m} P(|R(f_i) - R_{\text{emp}}(f_i)| > \varepsilon)$$

- Given every function is bounded as follows, according to the Law of Large Numbers:

$$P(|R_{\text{emp}}(f) - R(f)| > \epsilon) \leq 2\exp(-2n\epsilon^2)$$

Understanding the basics about SLT

- So, for all functions within the algorithm bias:

$$P\Big(|R(f_1) - R_{\mathrm{emp}}(f_1)| > \varepsilon \ \text{ or } \ |R(f_2) - R_{\mathrm{emp}}(f_2)| > \varepsilon \ \text{ or } \ \ldots \ \text{ or } \ |R(f_m) - R_{\mathrm{emp}}(f_m)| > \varepsilon \Big)$$

$$\leq \sum_{i=1}^{m} P(|R(f_i) - R_{\mathrm{emp}}(f_i)| > \varepsilon)$$

- Given every function is bounded as follows, according to the Law of Large Numbers:

$$P(|R_{\mathrm{emp}}(f) - R(f)| > \epsilon) \leq 2\exp(-2n\epsilon^2)$$

- So that Vapnik obtained:

$$\sum_{i=1}^{m} P(|R(f_i) - R_{\mathrm{emp}}(f_i)| > \varepsilon) \ \leq 2m\exp(-2n\varepsilon^2)$$

Understanding the basics about SLT

- This is one of his main results!

$$\sum_{i=1}^{m} P(|R(f_i) - R_{\mathrm{emp}}(f_i)| > \varepsilon) \;\leq 2m \exp(-2n\varepsilon^2)$$

  - Let us plot it!

Understanding the basics about SLT

- This is one of his main results!

$$\sum_{i=1}^{m} P(|R(f_i) - R_{\text{emp}}(f_i)| > \varepsilon) \leq 2m \exp(-2n\varepsilon^2)$$

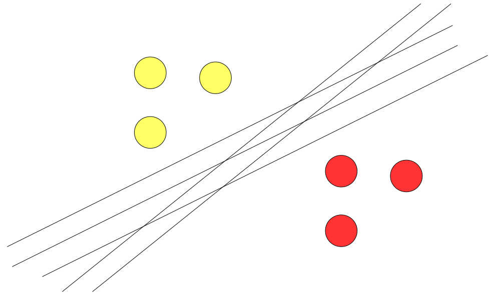
  - Let us plot it!
  - But how to define m?
    - Number of different classification/regression functions inside the algorithm bias

Understanding the basics about SLT

- This is one of his main results!

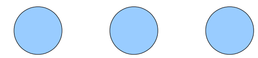
$$\sum_{i=1}^{m} P(|R(f_i) - R_{\text{emp}}(f_i)| > \varepsilon) \leq 2m \exp(-2n\varepsilon^2)$$

- Let us plot it!
- But how to define m?
  - Number of different classification/regression functions inside the algorithm bias
  - He had a clever idea (once more) of defining similar classifiers according to their outputs



Understanding the basics about SLT

- For example, consider 3 points in a two-dimensional plane as follows:

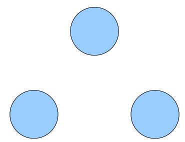


- Suppose linear functions are used to form classifiers:

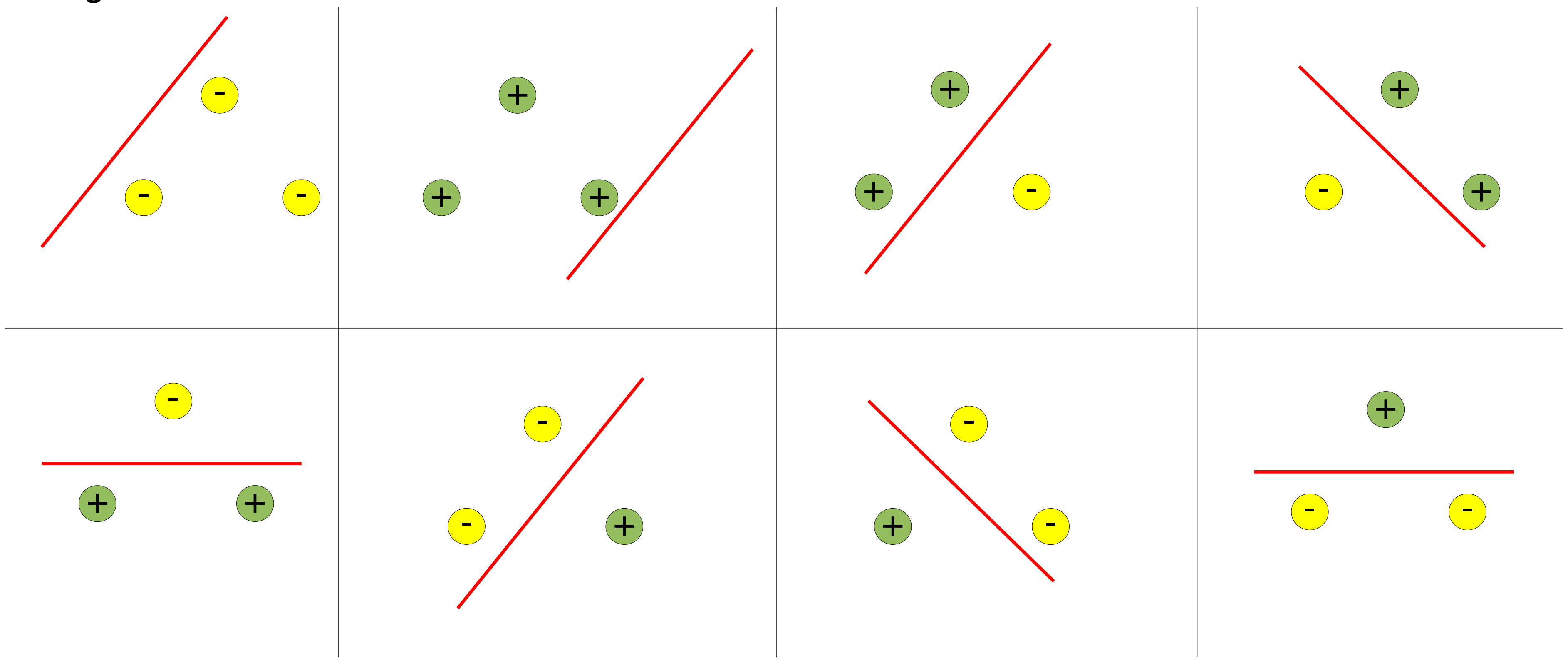
- We could shatter this sample in 4 different ways
  - But is there any other 3-point sample that we could shatter in more ways?

Understanding the basics about SLT

- Suppose we have the 3 points in different setting (still in R^2):



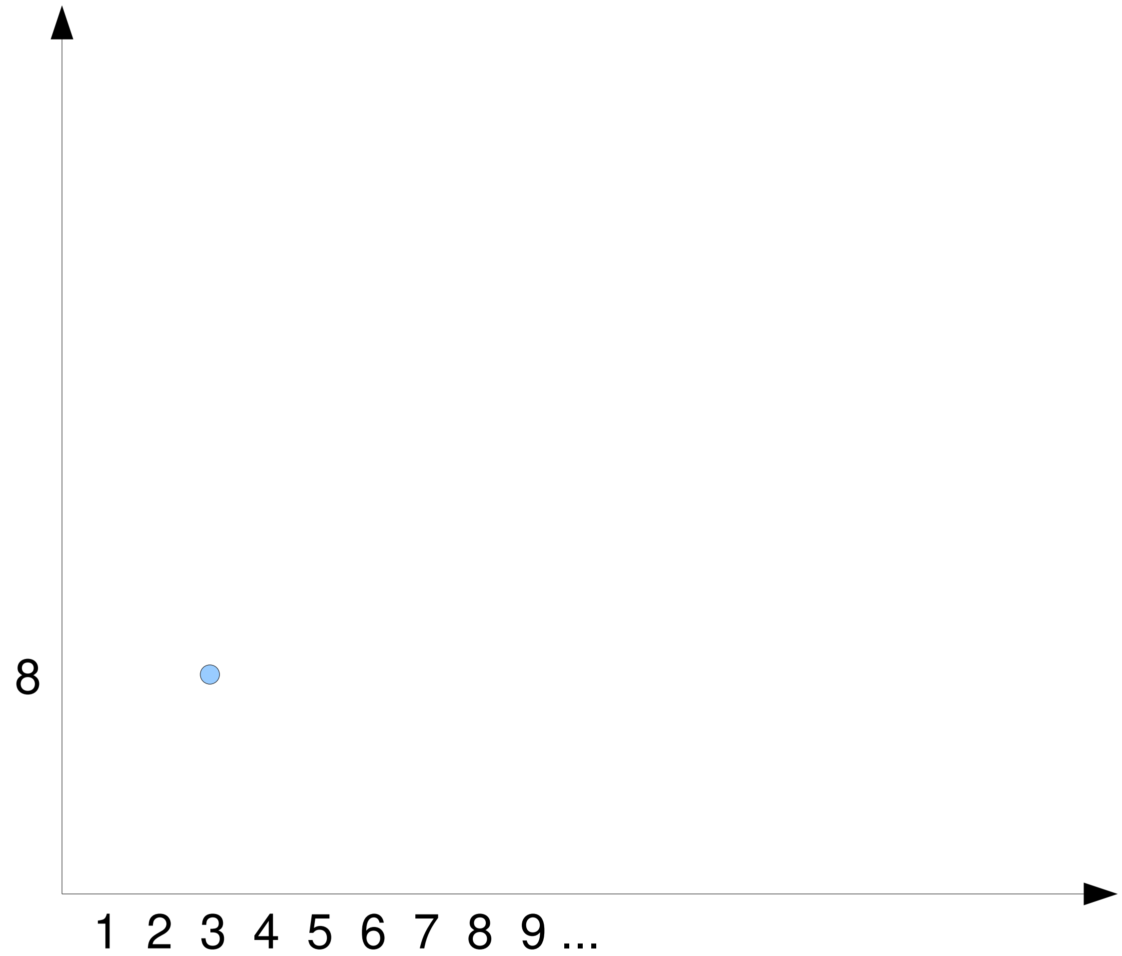
- Again consider F contains all linear functions:



- Observe F was capable of shattering this sample in all 2^n possible ways, what take us to the fact that F has a VC dimension at least equal to 3
  - Because there is at least one sample with 3 instances that can be shattered in all possible ways

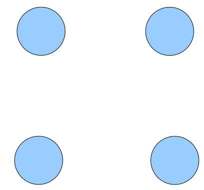
Understanding the basics about SLT

- In that sense, we conclude that for R^2:

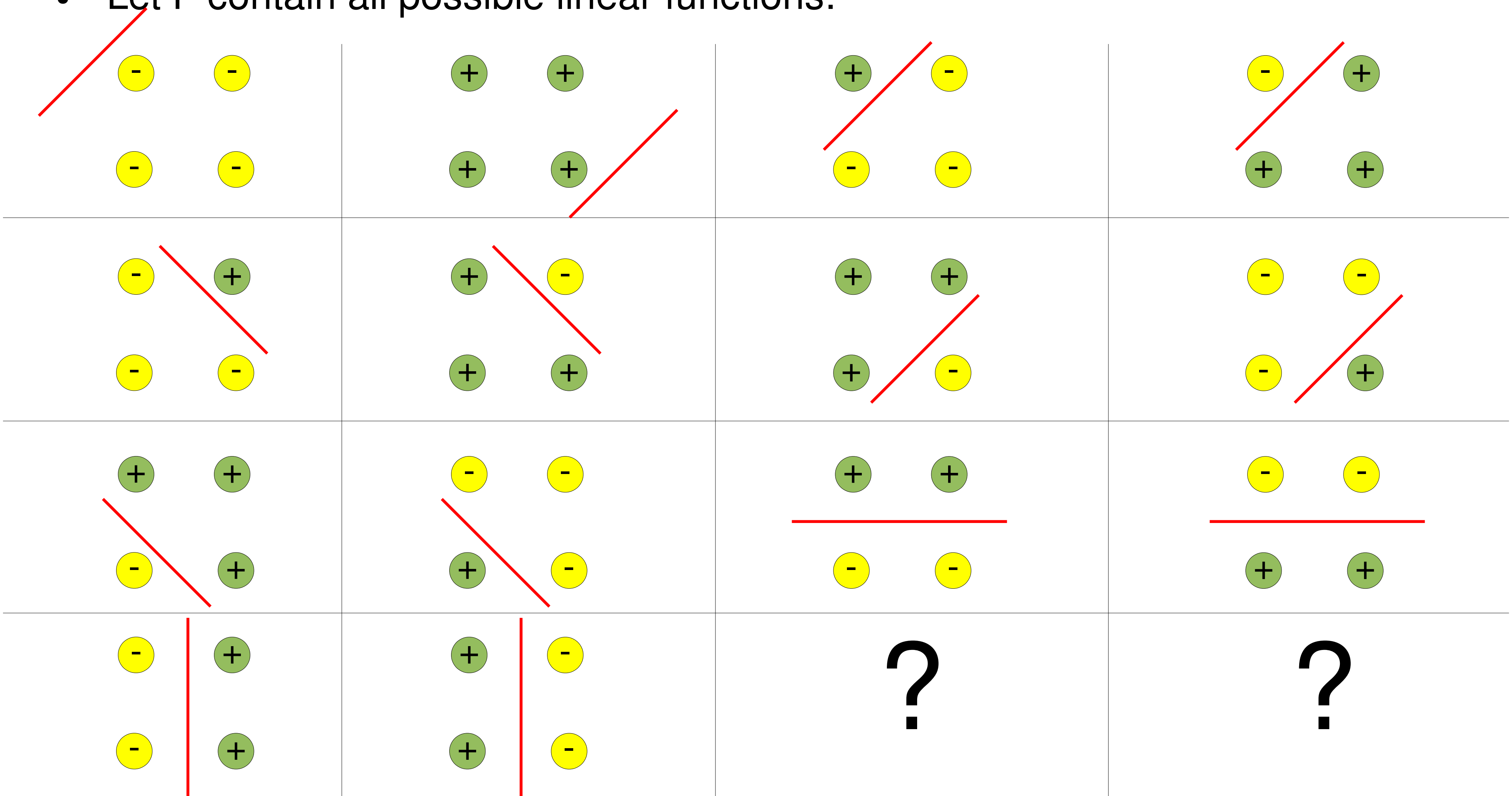


Understanding the basics about SLT

- Let's still consider in R^2:

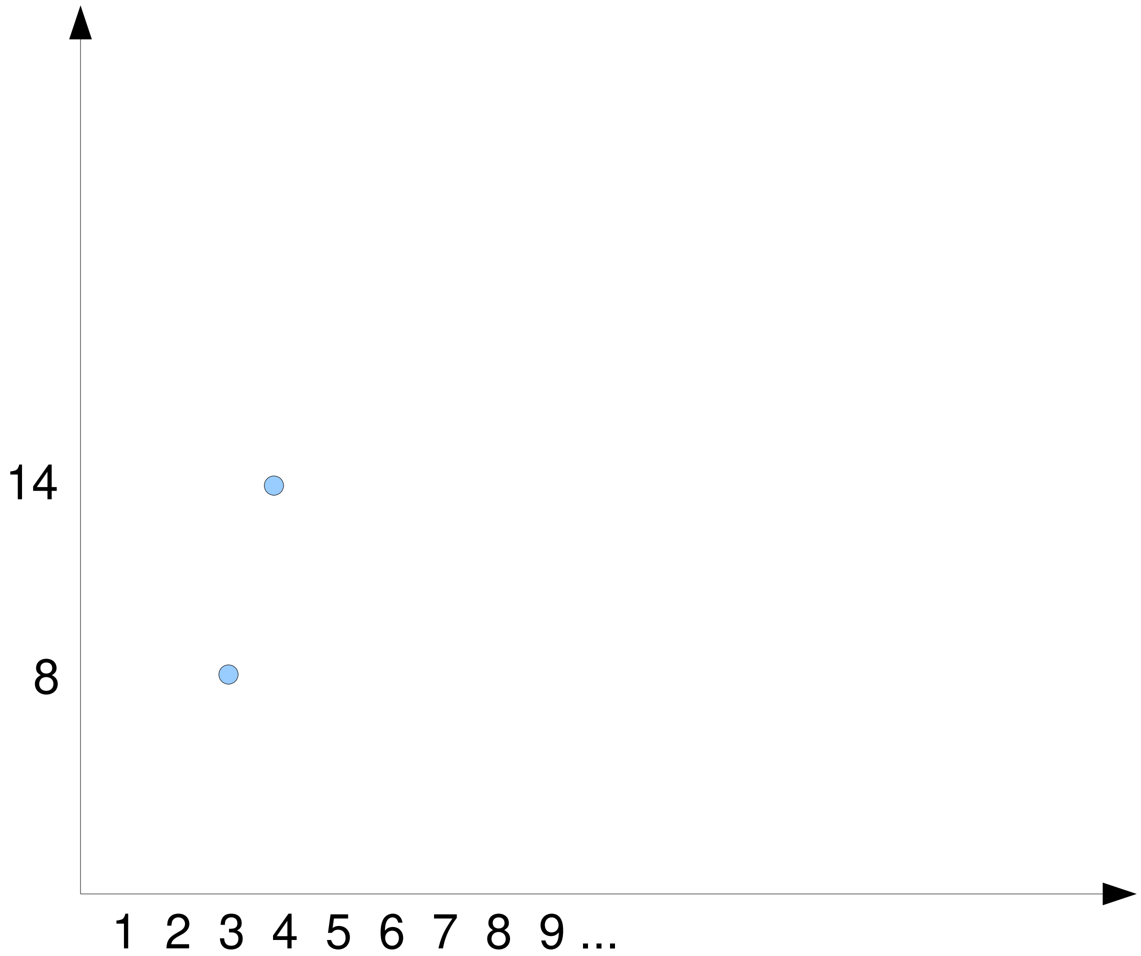


- Let F contain all possible linear functions:



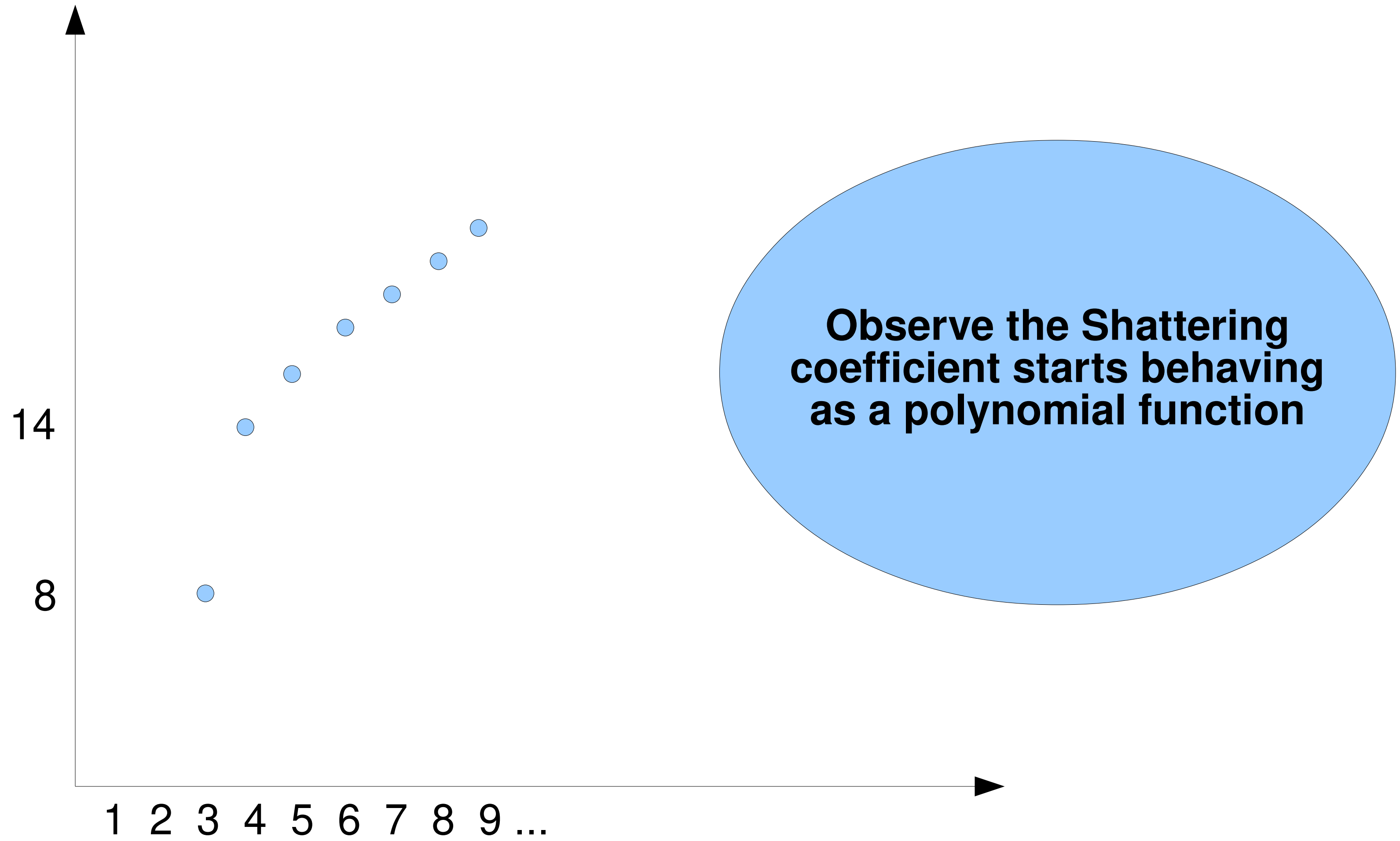
Understanding the basics about SLT

- In that sense, we conclude that for R^2:



Understanding the basics about SLT

- In that sense, we conclude that for R^2:



Understanding the basics about SLT

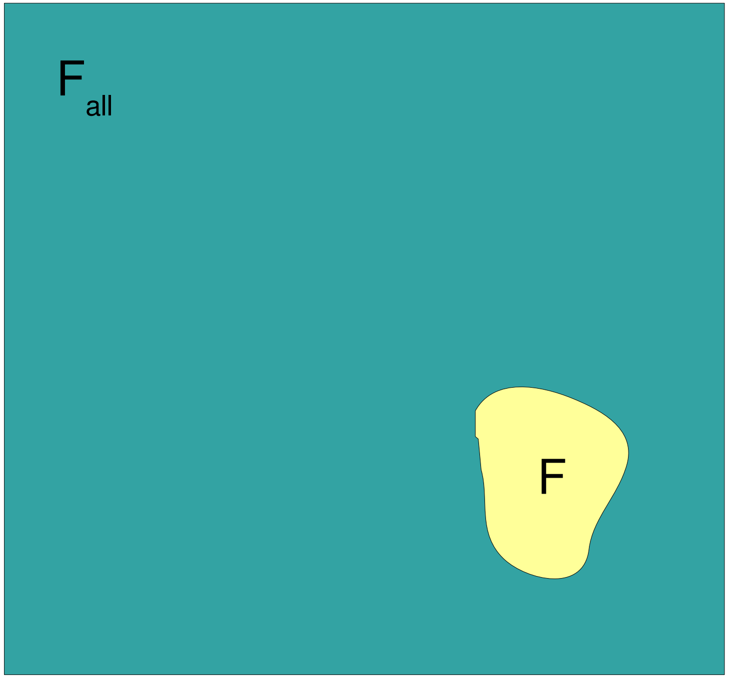
- In fact, Learning is only ensured if m(n) grows polynomially:

$$\sum_{i=1}^{m} P(|R(f_i) - R_{\text{emp}}(f_i)| > \varepsilon) \leq 2m \exp(-2n\varepsilon^2)$$

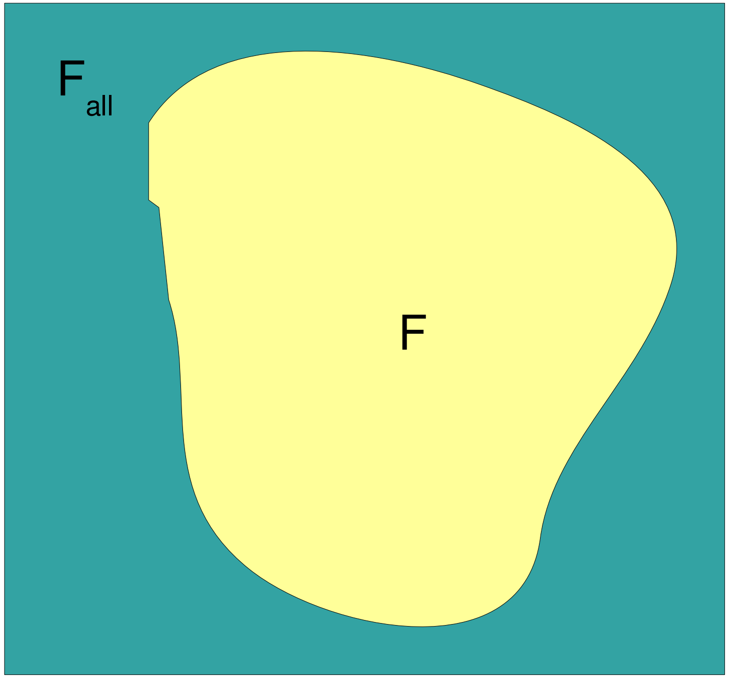
- Let us open the formulation and see what happens:
  - If it is polynomial
  - If it is exponential

Understanding the basics about SLT

- In this sense:



Polynomial Shattering coefficient



Exponential Shattering coefficient

References

- Vapnik, V., The Nature of Statistical Learning Theory, Springer, 2011
- Luxburg and Scholkopf, Statistical Learning Theory: Models, Concepts, and Results. Handbook of the History of Logic. Volume 10: Inductive Logic. Volume Editors: Dov M. Gabbay, Stephan Hartmann and John Woods, Elsevier, 2009
- Schölkopf, B., Smola, A. J., Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, MIT, 2002

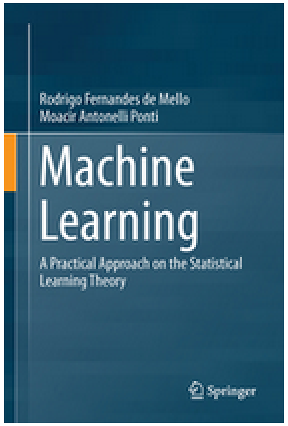
References

Springer

Search

Home Subjects Services Products Springer Shop About us

» Computer Science » Artificial Intelligence



Machine Learning

A Practical Approach on the Statistical Learning Theory

Authors: **Fernandes de Mello**, Rodrigo, **Antonelli Ponti**, Moacir

Machine Learning: A Practical Approach to the Statistical Learning Theory

Rodrigo Fernandes de Mello

Associate Professor

Universidade de São Paulo

Instituto de Ciências Matemáticas e de Computação

mello@icmc.usp.br

August 7th, 2019