SUPERVISED LEARNING

Fernando Pérez-Cruz

6th October 2004

FORMAT

- 3-hour lectures on Mondays. Room 305 Pearson
- Lectures slides and blackboard.
- Slides and extra material at:
  http://www.gatsby.ucl.ac.uk/∼fernando/SupLearn.html.
- E-mail: fernando@gatsby.ucl.ac.uk
- Office: 402 Alexandra House (Gatsby). Office: hours TBD.

EVALUATION

- Homework (50%) and Final Exam (50%).
- 6 Homework assignments.
- Deliver them on-time, penalty otherwise.
- Students from 4C55:
  - Best 4/6 homework marks.
  - Different exam.

MATERIAL

- Text Books:
  - C. Bishop, *Neural Networks for Pattern Recognition.* Claredom 1995.
  - D. MacKay, *Information Theory, Inference, and Learning Algorithms.* Cambridge University Press 2003.
  - B. Schölkopf and A. Smola, *Learning with Kernels.* MIT Press 2002.

- Additional Readings:
  - Duda, Hart and Stork, *Pattern Classification.* 2nd Edition. Wiley 2000
  - S. Haykin, *Neural Networks.* 2nd Edition. Prentice-Hall 1998.
  - T. Mitchell, *Machine Learning*. McGraw-Hill 1997.

REQUIREMENTS

- Algebra.
- Calculus.
- Optimisation.
- Probability and Statistics.
- Programming (Preferably Matlab).

COURSE OUTLINE

- Introduction to Machine Learning (3 lectures).
  - Why, Applications. Math and Optimisation review. Basic Tools.
- Bayesian Approach to Neural Networks (3 lectures).
  - Bayesian MLP. Approximations and Sampling. Gaussian Processes
- Support Vector Machines and Kernel Methods (3 lectures).
  - SVM for classification and Regression. Extensions: KPCA, KFD, ...
- Extensions (1 lecture).
  - Decision Trees, boosting ... . Reinforcement Learning.

LECTURERS

- Fernando Perez-Cruz (Coordinator).
- David J. C. Mackay.
- Iain Murray.
- Edward Snelson.
- Chu Wei.
- Nathaniel Daw.

Pattern Recognition

- Suppose you are a 4-year-old student, learning to read.
- First you need to know the letters.
- You teacher show you an "a" and tells you it is an a.
- with time you will learn to recognise as the a category:
- You have learn to generalised from the few examples of a's to any possible one.
- Can we teach a machine to do the same thing?

Novelty Detection

- Your are still at school, but by now you are 12.
- You got a test in which they show you four words:

  Renault Vespa Volkswagen Vauxhaul

  and they ask you choose the odd one out.
- There is no specific teacher. You need to assume something to answer.
- These problems can be ambiguous and present several answers, i.e:

  Beckham Button Owen Raul

- Can we teach a machine to do the same thing?

Machine Learning Basics

- We will always start from samples $(\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n)$ obtained from unknown $p(\mathbf{z})$ independently and identically distributed (i.i.d.).
- and we want to infer $p(\mathbf{z})$ or something about it.
- Inductive process. (Is it always possible?)
- Uncertainty, not deterministic.
- There is not (always) a correct answer.
- Prior Knowledge.

MACHINE LEARNING

- We need models to infer general information from data.

- Difficulties:
  - Noisy measures.
  - Complex and nonlinear interactions.
  - Have many variables (irrelevant).
  - Incomplete data.

- Move from the data to the model:
  "Do not fall in love with models, fall in love with data" (Hendry?)

APPLICATIONS

- Interdisciplinary Subject: Computer Science, Statistics, Math, Physics, Engineering, Psychology.

- Text/speech recognition.
- Speaker identification.
- Computer vision.
- Video labelling.
- Time series.
- Bioinformatics.
- Medical.
- Information retrieval.
- . . .
- Basically any you can think of.

Type of Learning

- Supervised Learning. There is an explicit teacher (inputs and outputs).
  - Objective: predict the output of any possible input.
  - Classification ($y \in \{1, 2, \ldots, \}$) or Regression $y \in \mathbb{R}$.
- Unsupervised Learning. There is only inputs, no teacher.
  - Objective: Explain x.
  - Density estimation, clustering, feature selection or detect anomalies.
- Reinforcement Learning

SUPERVISED LEARNING

- Start with samples $\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$ i.i.d. from $p(\mathbf{x}, y)$.

- Objective: predict y given any $\mathbf{x}$.

- Three approaches:
  - Model $p(\mathbf{x}, y)$.
  - Model $p(y|\mathbf{x})$.
  - Obtain "the best" y given $\mathbf{x} \Rightarrow y = f(\mathbf{x})$.

DETERMINISTIC APPROACH

- Obtain "the best" y given $\mathbf{x}$, with respect to what?

- We need a loss function between the truth (y) and its prediction ($f(\mathbf{x})$): $L(y, f(\mathbf{x}))$.

- We can compute the risk of a function $f \in \mathcal{F}$ as:

$$R(f) = \int_{\mathcal{Y}\times\mathcal{X}} L(y, f(\mathbf{x}))p(y, \mathbf{x})dyd\mathbf{x}$$

- to get the best prediction we will have to look for the best $f \in \mathcal{F}$, the one with lowest risk. But we do not know $p(y, \mathbf{x})$.

PROBABILISTIC APPROACH

- We are interested in knowing more about y, $p(y|\mathbf{x})$.

- If we knew $p(y|\mathbf{x})$, we can say that finding the best y is like choosing its mode (or maybe its mean).

- This information might be very useful in many applications:
  - Medical.
  - Financial.
  - Telecommunications.

Bayes Rule

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

REGRESSION

- Start with our training data set:

$$\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}, \qquad \mathbf{x}_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{bmatrix} \; (\mathbf{x}_i \in \mathbb{R}^d) \text{ and } \mathbf{y}_i \in \mathbb{R}.$$

- Find a function: $f(\cdot) : \mathbb{R}^d \longrightarrow \mathbb{R}$, such that $\forall \mathbf{x} \in \mathbb{R}^d \Rightarrow y = f(\mathbf{x})$.
  - Fit the training data.
  - Generalised to unseen patterns.

- Where do we look for $f(\cdot)$?

EXAMPLES

- Predict House price, from neighbourhood, size (m^2), number of rooms, built period, condition ...

- Cyclosporine blood concentration from anthropometrical, clinical and biochemical data of patients.

- Volatility (variance) in financial from observations of the times series and other relevant variables.

EMPIRICAL RISK MINIMIZATION

- Define a loss between the true output y and the predicted output $f(\mathbf{x})$:

$$L(y, f(\mathbf{x}))$$

- Empirical Risk of is the average over the training data set:

$$R_{emp}(f) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(\mathbf{x}_i))$$

- We want it to "fit the data". We can find

$$\widehat{f} = \operatorname*{argmin}_{f \in \mathcal{F}} R_{emp}(f) = \operatorname*{argmin}_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(\mathbf{x}_i))$$

EMPIRICAL RISK MINIMIZATION

- We can find this by equating to zero its derivative:

$$\frac{dR_{emp}}{df} = \sum_{i=1}^{n} \frac{dL(y_i, f(x_i))}{df} = 0$$

- To solve this equation, we need:
  - to know what $L(\cdot, \cdot)$ measures.
  - to know the form of $f(\cdot)$.

- Will this always give a good answer?

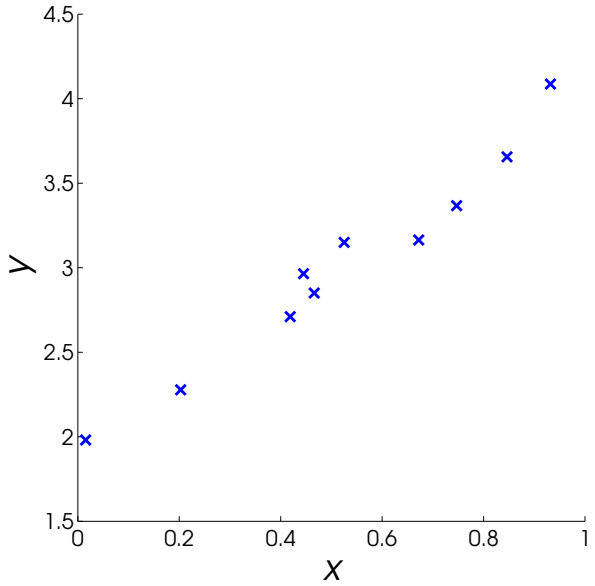
EXAMPLE: LINEAR LEAST SQUARE

- We chose a linear model for $f(\cdot)$:

$$f(\mathbf{x}) = \mathbf{x}^\top \mathbf{w} + b$$
$$\mathbf{w} \in \mathbb{R}^d \quad \text{and} \quad b \in \mathbb{R}$$

- We choose for the loss function the square loss:

$$L(y, f(\mathbf{x})) = \frac{1}{2}(y - f(\mathbf{x}))^2 = \frac{1}{2}(y - (\mathbf{x}^\top \mathbf{w} + b))^2$$



One Dimension

- The weight vector is a scalar: $\mathbf{w} = w \in \mathbb{R}$.

- We need to minimise:

$$\min_{w,b} R_{emp}(w,b) = \min_{w,b} \frac{1}{2n} \sum_{i=1}^{n} (y_i - wx_i - b)^2$$

- To obtain the minimum we need to find a point in which the gradient is zero:

$$\nabla_{w,b} R_{emp}(w,b) = \begin{bmatrix} \frac{\partial R_{emp}}{\partial w} \\ \\ \frac{\partial R_{emp}}{\partial b} \end{bmatrix} = \begin{bmatrix} -\frac{1}{n} \sum_{i=1}^{n} x_i (y_i - wx_i - b) \\ \\ -\frac{1}{n} \sum_{i=1}^{n} (y_i - wx_i - b) \end{bmatrix} = \begin{bmatrix} 0 \\ \\ 0 \end{bmatrix}$$

ONE DIMENSION SOLUTION

- We can now operate, leading to:

$$b = \frac{1}{n}\sum_{i=1}^{n} y_i - \frac{w}{n}\sum_{i=1}^{n} x_i$$

$$w\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} y_i x_i - b\sum_{i=1}^{n} x_i$$

- The *optimal* slope and threshold are equal to:

$$w^* = \frac{\sum_{i=1}^{n} y_i x_i - \frac{1}{n}\sum_{i=1}^{n} y_i \sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} x_i^2 + \frac{1}{n}\sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_i} \qquad b = \frac{1}{n}\sum_{i=1}^{n} y_i - \frac{w^*}{n}\sum_{i=1}^{n} x_i$$

LINEAR LEAST SQUARE SOLUTION

- Now we extended for $\mathbf{x}$ in any dimension. The equations are very similar:

$$\frac{\partial R_{emp}}{\partial \mathbf{w}} = -\frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i(y_i - (\mathbf{x}_i^\top \mathbf{w} + b)) = \mathbf{0}$$

$$\frac{\partial R_{emp}}{\partial b} = -\frac{1}{n}\sum_{i=1}^{n} (y_i - (\mathbf{x}_i^\top \mathbf{w} + b)) = 0$$

- This can be easily solved using matrices, i.e:

$$\sum_{i=1}^{n} \mathbf{x}_i y_i = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{n1} \\ x_{12} & x_{22} & \cdots & x_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1d} & x_{2d} & \cdots & x_{nd} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \mathbf{X}^\top \mathbf{y}$$

MATRIX NOTATION

- which can be expressed in matrix form as:

$$\begin{aligned}(\mathbf{X}^\top\mathbf{X})\mathbf{w} + \mathbf{X}^\top\mathbf{1}b &= \mathbf{X}^\top\mathbf{y} \\ \mathbf{1}^\top\mathbf{X}\mathbf{w} + nb &= \mathbf{1}^\top\mathbf{y}\end{aligned} \quad \Rightarrow \quad \begin{bmatrix}(\mathbf{X}^\top\mathbf{X}) & \mathbf{X}^\top\mathbf{1} \\ \mathbf{1}^\top\mathbf{X} & n\end{bmatrix}\begin{bmatrix}\mathbf{w} \\ b\end{bmatrix} = \begin{bmatrix}\mathbf{X}^\top\mathbf{y} \\ \mathbf{1}^\top\mathbf{y}\end{bmatrix}$$

  where $\mathbf{1} = [1, 1, \ldots, 1]^\top$, $n \times 1$ vector of ones.

- If we redefine $\overline{\mathbf{x}}_i = [\mathbf{x}_i 1]$ and $\overline{\mathbf{w}} = [\mathbf{w}b]$, the above systems can be simplified to:

$$(\overline{\mathbf{X}}^\top\overline{\mathbf{X}})\overline{\mathbf{w}} = \overline{\mathbf{X}}^\top\mathbf{y} \quad \Rightarrow \quad \overline{\mathbf{w}}^* = (\overline{\mathbf{X}}^\top\overline{\mathbf{X}})^{-1}\overline{\mathbf{X}}^\top\mathbf{y}$$

PROPERTIES

- The optimal solution $\mathbf{w}^*$ and b^* present several relevant properties.

- The mean error is zero ($e_i = y_i - (\mathbf{x}_i^\top \mathbf{w}^* + b^*)$)

$$\frac{1}{n}\sum_{i=1}^{n} e_i = 0 \qquad \longleftarrow \qquad \left(\frac{\partial R_{emp}}{\partial b} = -\frac{1}{n}\sum_{i=1}^{n}(y_i - (\mathbf{x}_i^\top \mathbf{w} + b))\right) \qquad (1)$$

- The error is not correlated with any components of the data:

$$\frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i e_i = 0 \qquad \longleftarrow \qquad \left(\frac{\partial R_{emp}}{\partial \mathbf{w}} = -\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i(y_i - (\mathbf{x}_i^\top \mathbf{w} + b))\right) \qquad (2)$$

NONLINEAR FUNCTIONS

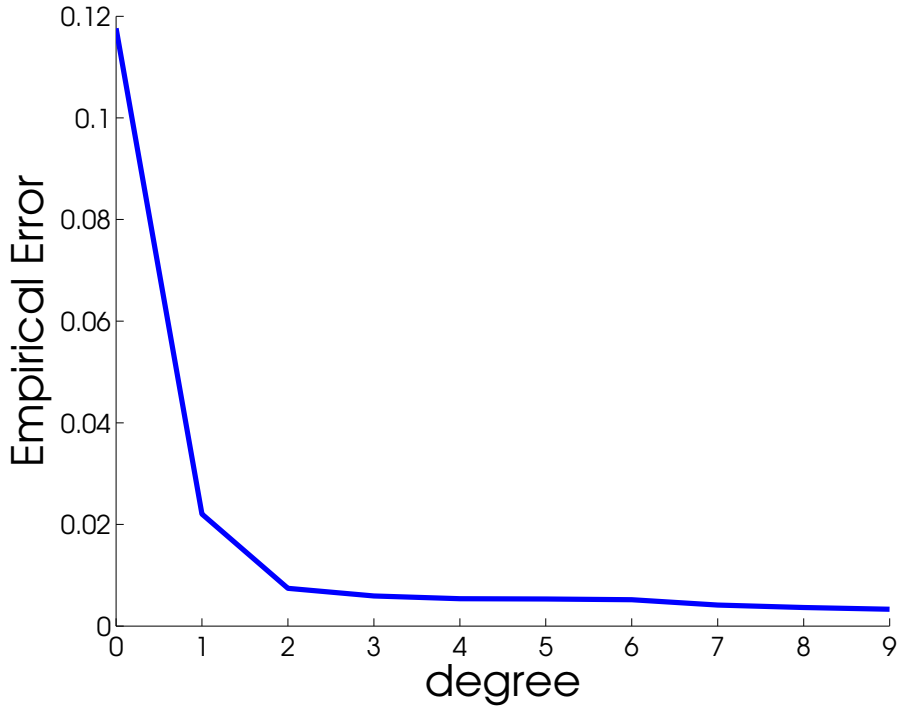
- The linear relation is quite limited.
- The relation between $\mathbf{x}$ and y needs not be linear.
- One way of expressing nonlinearities is using powers of $\mathbf{x}$.
- Example: $x_i \longrightarrow [1, x_i, x_i^2, x_i^3, \ldots]^\top$
- we can get linear, quadratic or cubic ... relations between x and y.
- In general we, can map $\mathbf{x} \rightarrow \boldsymbol{\phi}(\mathbf{x})$ and apply the least square regression.

ERROR

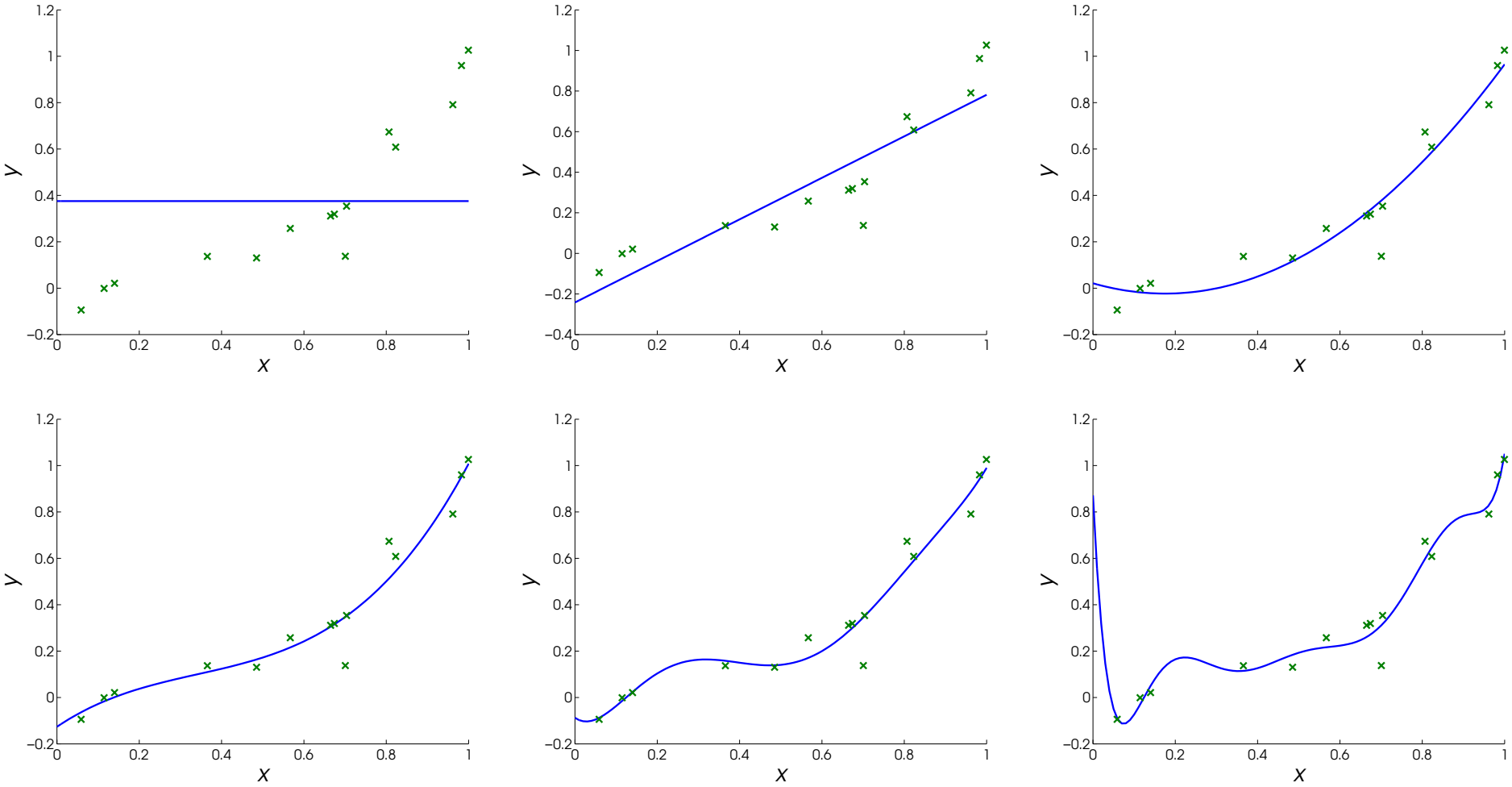
- We are going to fit a nonlinear relation between x and y using a increasing polynomial expansion of x.

- We present the Empirical Risk in the Figure

- Which is the best polynomial approximation

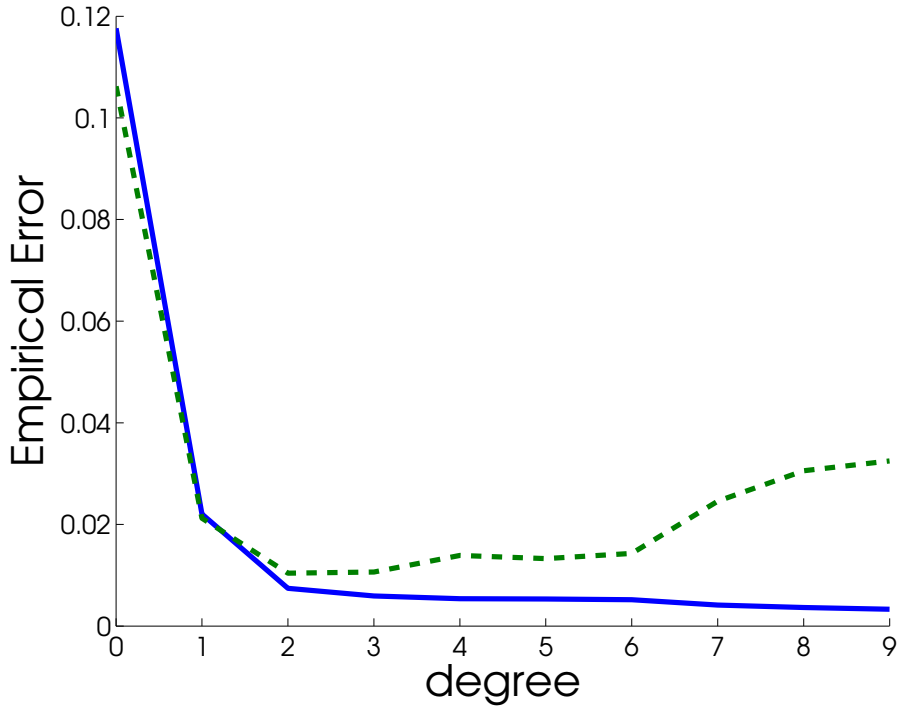


Different Solutions



ERROR OVER TEST SET

- We have computed now the error over a test set.
- We present the Empirical Risk in the Figure
- Which is the best polynomial approximation now
- Overfitting and underfitting.



OVERFITTING

- Occurs when we have large input space a very few data.
- In the previous example, we have increased the number of dimensions for a fix amount of data.
- If we used a 14^{th} order polynomial expansion we will be able to fit the training point without error.
- We will be memorising the data.
- We will not be able to generalise to unseen data.
- Curse of dimensionality (Bellman 1961).

SOLUTIONS

- Reduced complexity algorithms.
- Regularization.
- Capacity control.
- Prior knowledge.