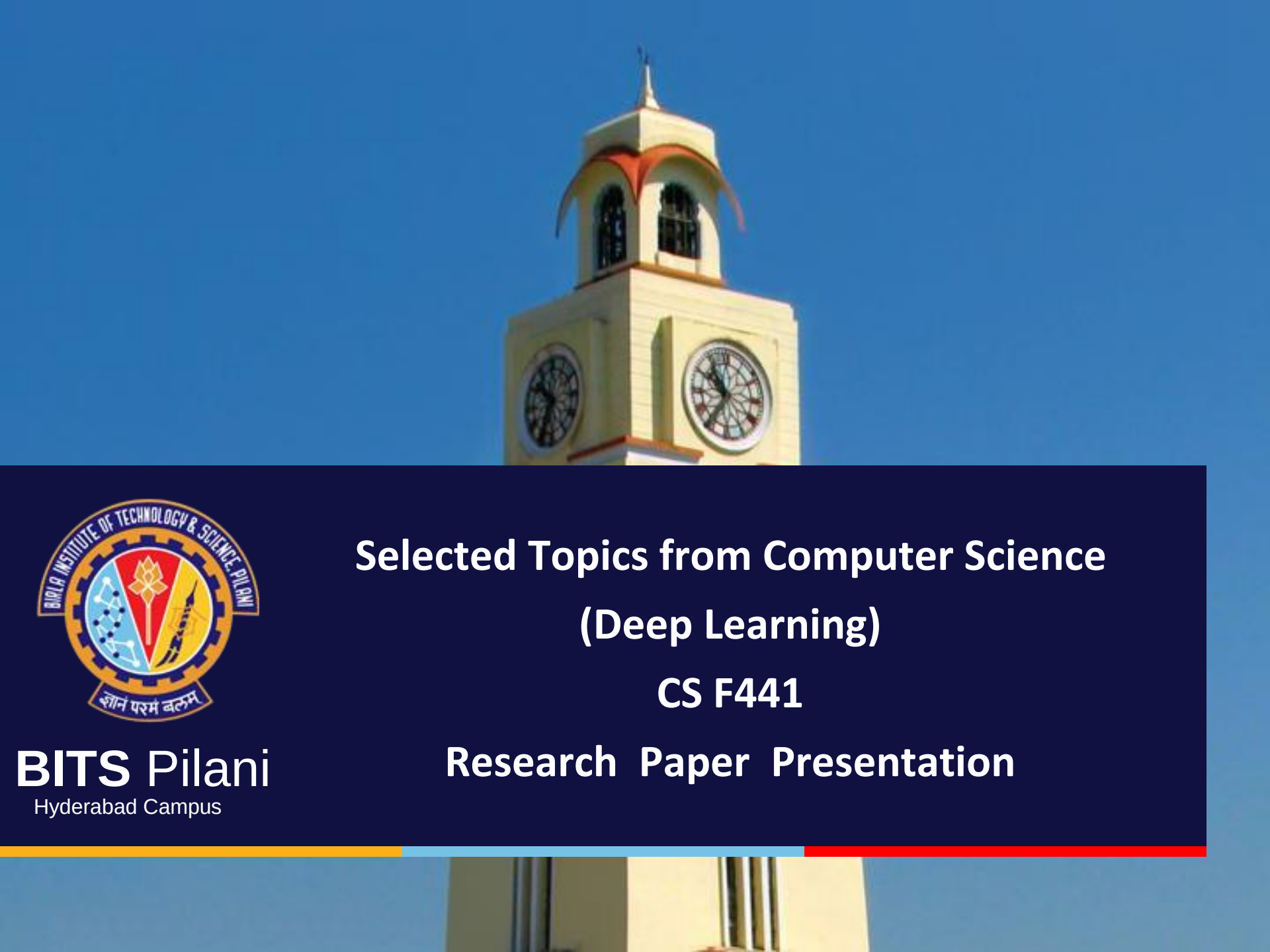


Selected Topics from Computer Science

(Deep Learning)

CS F441

Research Paper Presentation

BITS Pilani

Hyderabad Campus

Group Details (with Id No and Name)

Raj Kashyap Mallala	2017A7PS0025H
Varun Gumma	2017A7PS0165H
Abhinav Sukumar Rao	2018A7PS0172H

Research Paper Details

Name of the paper: Understanding Deep Learning Requires Rethinking Generalization (ICLR Best Paper Award)

Authors: Chiyuan Zhang (MIT), Samy Bengio (Google Brain), Moritz Hardt (Google Brain), Benjamin Recht (UCB), Oriol Vinyals (Google DeepMind)

Published in: ICLR 2017

Introduction to the research problem (in bullet points)

- The problem is that attributing regularization or the model's properties is not enough to explain why Deep learning models perform well in practice. Despite their massive size, successful deep artificial neural networks can exhibit a remarkably small difference between training and test performance.
- Good performance is noticed even when the labels are not true to the dataset i.e. the models perform well even in training even with random data.
- The effect of explicit regularization, finite sample expressivity, and finally implicit regularization are seen in this paper, and it attempts to breakdown the explanations currently being made on DL models.

Related work (in bullet points)

- *Hardt et al. (2016)* gave an upper bound on the generalization error of a model trained with stochastic gradient descent in terms of the number of steps gradient descent took.
- Universal approximation theorem for MLPs *Cybenko et al. (1989)*

Related work (in bullet points)

A few other major papers referred were:

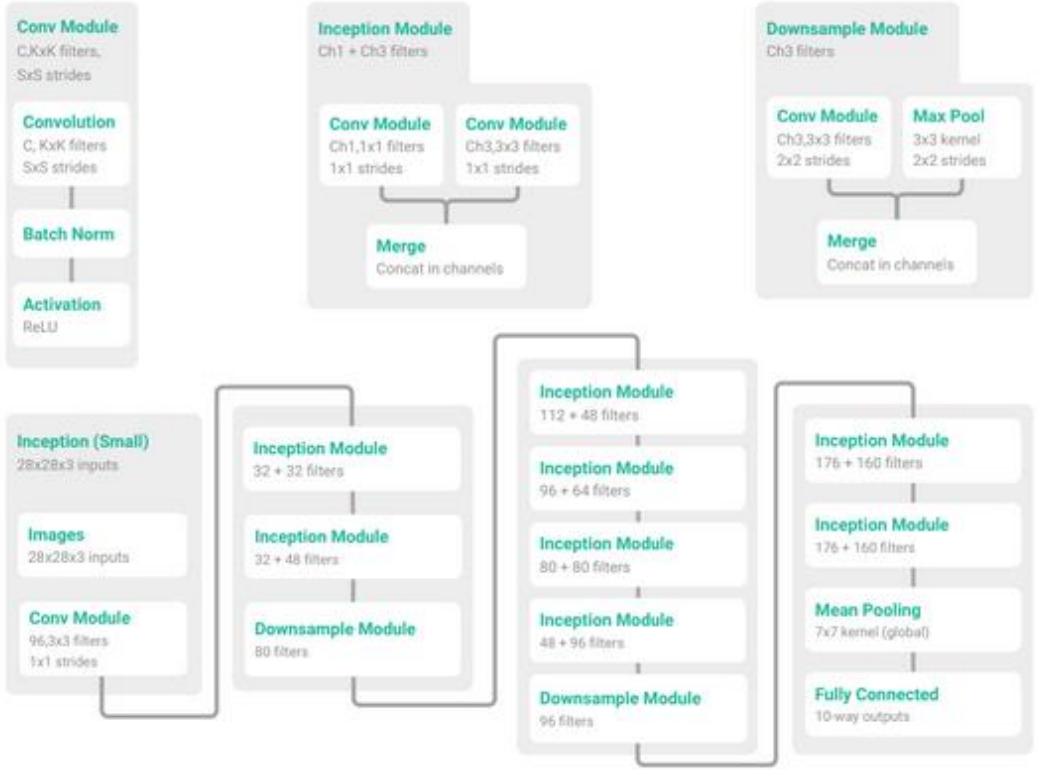
- Uniform stability (Bousquet & Elisseeff, 2002): Refers to how sensitive an algorithm is to the replacement of a single example. This is solely the property of the algorithm and does not deal with the data or its distribution.
- Dropout (Srivastava et al. 2014): Dropout causes certain neurons in each layer to randomly get deactivated during training. This forces the network to learn using different subsets of features. This ensures that no feature is not given a large weight. Also, dropout generates multiple models with varying configuration, which can act as ensembles.
- Batch normalization (Ioffe & Szegedy, 2015) is an operator that normalizes the layer responses within each mini-batch.

Proposed model / theory in the paper (in bullet points)

- The experimental setup involves:
  - A small version of an AlexNet, a model architecture that can be used to train higher resolution images as well.
  - “The small Alexnet is constructed by two (convolution 5x5→ max-pool 3x3 → local-response-normalization) modules followed by two fully connected layers with 384 and 192 hidden units, respectively. Finally a 10-way linear layer is used for prediction.”

Proposed model / theory in the paper (in bullet points)

- A small version of Inception, which attempts to use multiple kernel sizes for detecting both local and global features. The architecture is as follows:

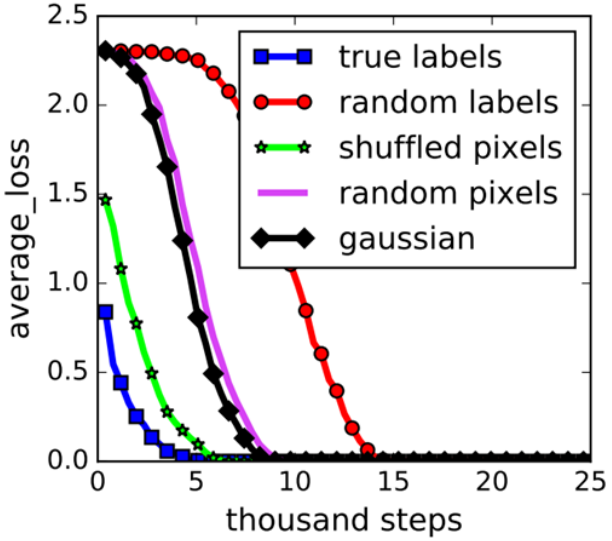


Proposed model / theory in the paper (in bullet points)

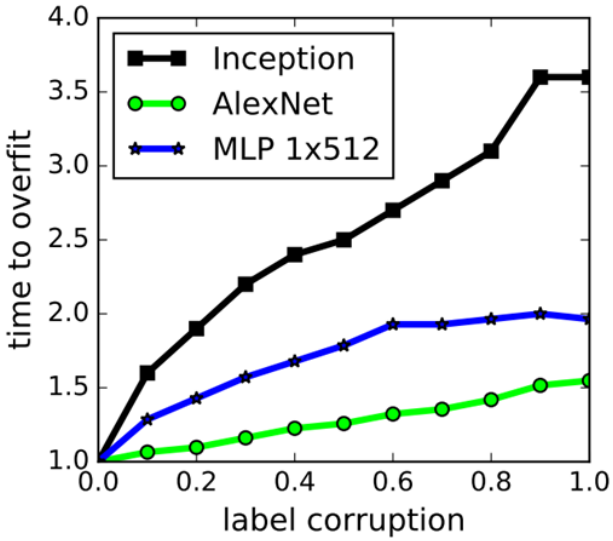
- Both of these models are trained on 2 Datasets, the dataset CIFAR 10 and the Imagenet dataset.
- Additionally, for CIFAR 10, 2 more models are considered an MLP with 3 hidden layers are 512 units each, and an MLP with 1 hidden layer and 512 units
- 3 kinds of regularizers are covered: Data augmentation, weight decay (L2), and dropout.
- Other implicit regularizers such as Batch normalization, and early stopping are also looked upon.

Experiments, if any (in bullet points)

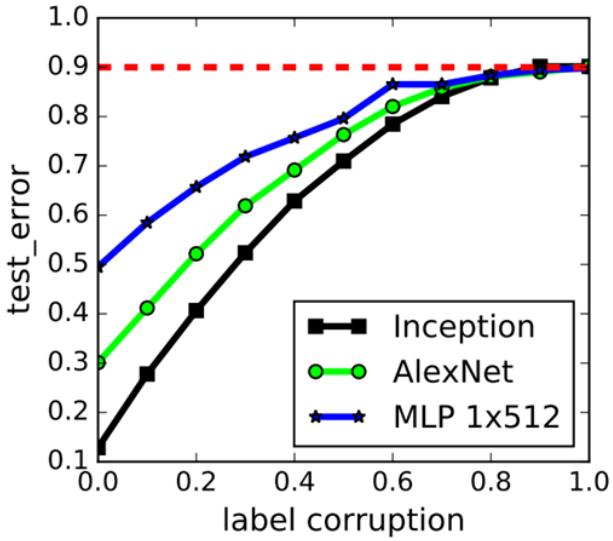
- They run SGD with momentum (0.9) CIFAR10 with 0.1 or 0.01 learning rate, which decays by a factor of 0.95 per epoch.
- ImageNet is trained on a asynchronous distributed SGD system.



(a) learning curves

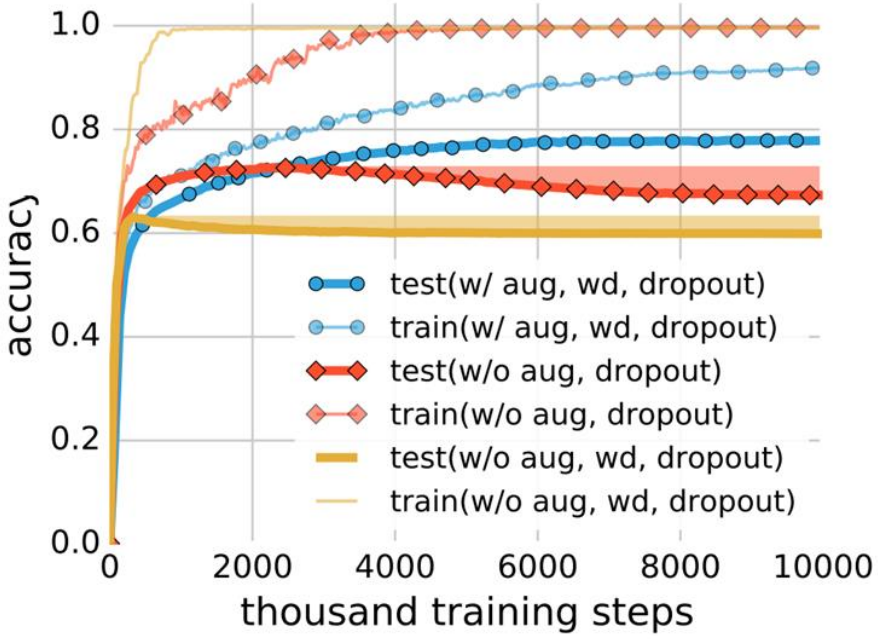


(b) convergence slowdown

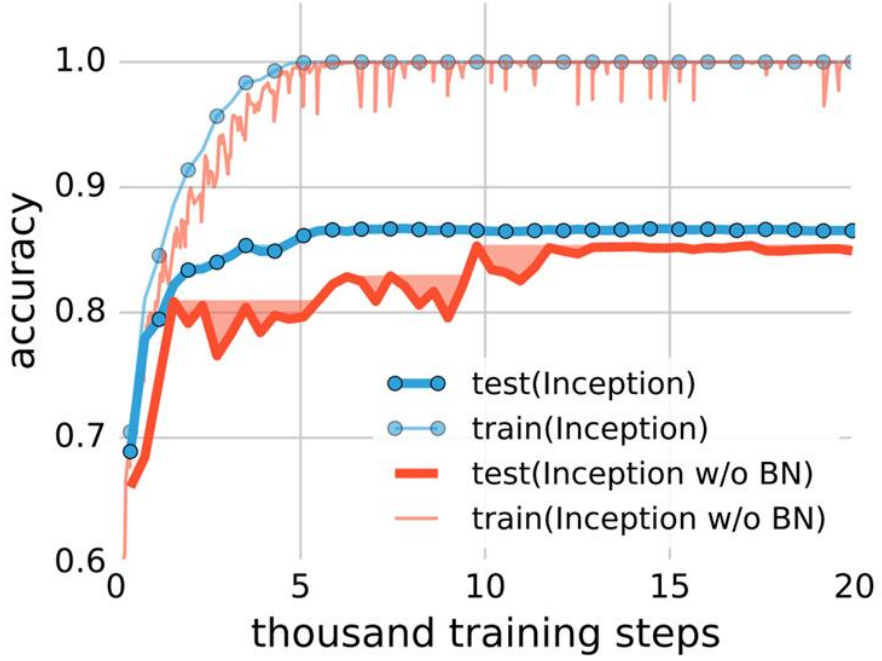


(c) generalization error growth

Experiments, if any (in bullet points)



(a) Inception on ImageNet



(b) Inception on CIFAR10

Conclusion / major outcomes (in bullet points)

- The experiments they conducted emphasize that the effective capacity of several successful neural network architectures is large enough to shatter the training data. (shattering: arrange them into classes with no error)
- Deep Networks have the capacity to fit even random labels or random noise to given labels, i.e. they can memorize the data.
- This means there is no generalization by the network in this case and any prediction is no better than a random prediction and hence the accuracy is just 1/N (where N is the number of classes)

Conclusion / major outcomes (in bullet points)

- Another major observation is that several properties of the training (convergence and speed of training) with random data/random labels/corrupted labels still are unaffected.
- A two layer Neural Network with ReLU activation with 2n+d parameters can represent any function on a sample of size n with d dimension. (Finite Sample Expressivity)
- While using the kernel method there's no improvement using any explicit or implicit regularizers. Which could mean that while Regularization can improve generalization, it's not the only sole reason, since even upon removal, the models generalize very well with the method.
- For linear models (Linear/Logistic Regression), SGD can act as a implicit regularizer. Small datasets with Gaussian Kernels can generalize well without regularization.

Conclusion / major outcomes (in bullet points)

- They also claim that minimum norm is not the criterion for the generalization performance, and just is a part of generalization.
- Regularization is necessary when there are more parameters to be learnt that the data points. Regularizers confine the learning to a subset of the hypothesis space with manageable complexity such that there is no loss in generalization.

Thank You!!

Footnote: https://www.youtube.com/watch?v=O42vde4tbG0

Some Math

Consider
→ d features, n data points
→ $d \geq n$
→ infinite solutions, overfit model.
→ ∴ $Xw = Y$ → true, as some solution obviously exists.

Using SGD:

$$w_0 = w - \eta \epsilon_1 x_1 - \eta \epsilon_2 x_2 \cdots \eta \epsilon_n x_n$$

$$= w - \sum_{i=1}^{1} \eta \epsilon_i x_i = w + \sum_{i=1}^{n} -\eta \epsilon_i x_i$$

$$= w + \sum_{i=1}^{n} \alpha_i x_i = w + x^T \alpha$$

If $w = [0]$ → initially w is a zero vector.

∴ $w_0 = x^T \alpha$ → some constants.

∴ $Xw_0 = Y$ (satisfies)

Some math contd..

$\therefore X X^T \alpha = Y \quad ; \quad K\alpha = Y \; ; \quad \boxed{\alpha = YK^{-1}}$

using L2 norm :

$$L = \frac{1}{2}(\hat{Y} - Y)^2 + \frac{\lambda}{2} w^T w$$

$$\frac{\partial L}{\partial w} = X^T (X w^* - Y) + \lambda w^* = 0$$

$$\therefore (X^T X + \lambda I) w^* = X^T Y$$

$$\therefore \quad w^* = (X^T X + \lambda I)^{-1} (X^T Y)$$

$$= (X^T X + \lambda I)^{-1} (X^T X X^T \alpha)$$

if $\lambda \simeq 0$; (very low of λ)

$$w^* = (X^T X)^{-1} (X^T X)(X^T \alpha) = X^T \alpha = w_0$$

Some math contd..

∴ this implies, weights obtained using SGD are same as the optimal weights obtained from norm equations with L_2 penalty with a very small regularizing parameter value.

— X —

$b_1 < x_1 < b_2 < x_2 < b_3 \cdots b_n < x_n$

$A_{n \times n} = [\max\{x_i - b_j, 0\}]_{ij}$ has full rank

Consider $C(x) = \sum_{j=1}^{n} w_j \max[(a^T x) - b_j, 0]$

$a \in \mathbb{R}^d$
$b, w \in \mathbb{R}^n$

$\mathbb{R}^n \to \mathbb{R}$

fix $S: \{z_1, z_2, \cdots z_n\}$ distinct & $y \in \mathbb{R}^n$

find a, b, w so $y_i = C(z_i)$

To do this: choose a, b such that $x_i = \langle a, z \rangle \ \forall i$ follows $b_1 < x_1 < b_2 < x_2 \cdots b_n < x_n$

now if, $y_i = C(z_i)$ $i = 1 \cdots n$

we can say: $C(z) = Aw = y$

$[\max\{\langle a, z_i \rangle - b_j\}, 0]_{ij} \, w$

has full rank

$\Rightarrow$ Solve $Aw = y$ to get your weights