SMAI-M20-L30:Neural Network Learning

C. V. Jawahar

IIIT Hyderabad

October 26, 2020

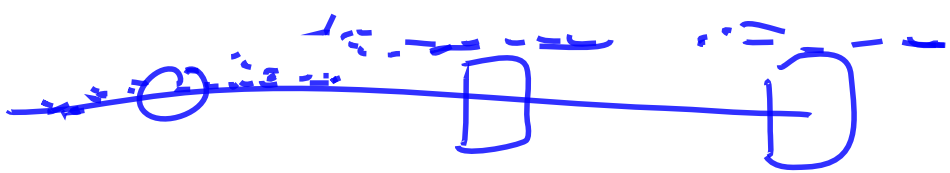
Announcements

- Quiz 2: Tentatively for next week. (say wed.)
- Refresh your programming and Jupyter notebooks.

Class Review

SVM and Kernels: Revisiting the illustrative problems we already solved.

- if samples scale in magntitude what happens to the solutions?
- if more samples come around a sample, what happens to the number of support vectors and α?
- ExOR and Paritty problems
- Does addition of new samples lead to new SVs?

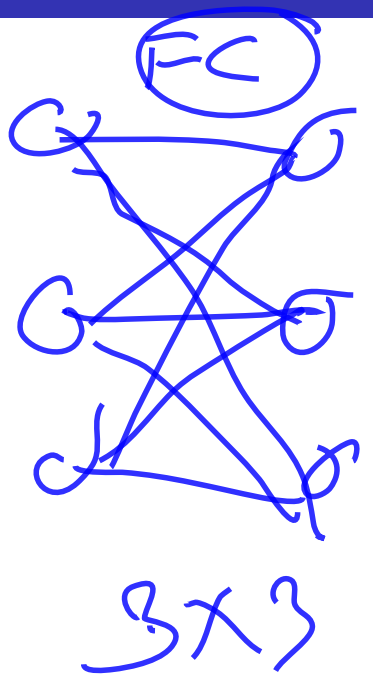


Blank

Recap:

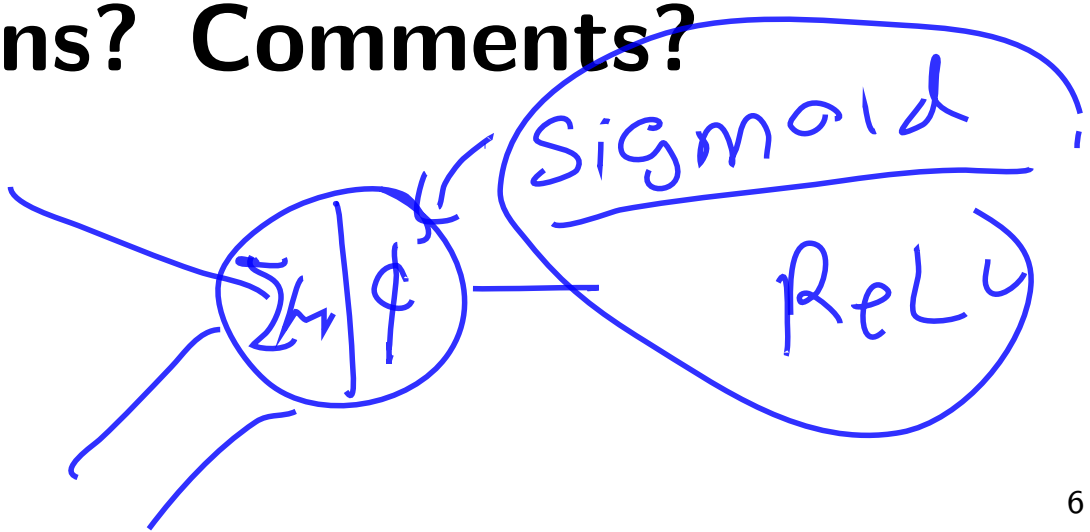
- **Supervised Learning:** Formulation, Conceptual Issues, Concerns etc.
- **Classifiers:** (i) Nearest Neighbour, (ii) Notion of a Linear Classifier (iii) Perceptrons (iv) Bayesian Optimal Classifier (v) Logistic Regression (vi) Multiclass classification architectures (v) SVMs (hard margin, soft margin, kernel)
- **Dimensionality Reduction and Applications:** (i) Feature Selection and Extraction (ii) PCA (iii) LDA (iv) Eigen face
- **Matrix Factorization and Applications:** (i) SVD, (ii) Eigen Decomposition (iii) Matrix Completion (iv) LSI (v) Recommendations
- **Neural Network Architectures and Learning** (i) Neuron model, Single Layer Perceptrons (ii) SLP(iii) MLP. (iv) Backpropagation
- **Other Topics:**
  - Linear Regression
  - Probabilistic View, Bayesian View, MLE
  - Gradient Descent: Stochastic and Batch GD
  - Loss Functions and Optimization
  - Eigen Vector based optimization
  - Kernel Functions and Kernel Matrix

This Lecture:

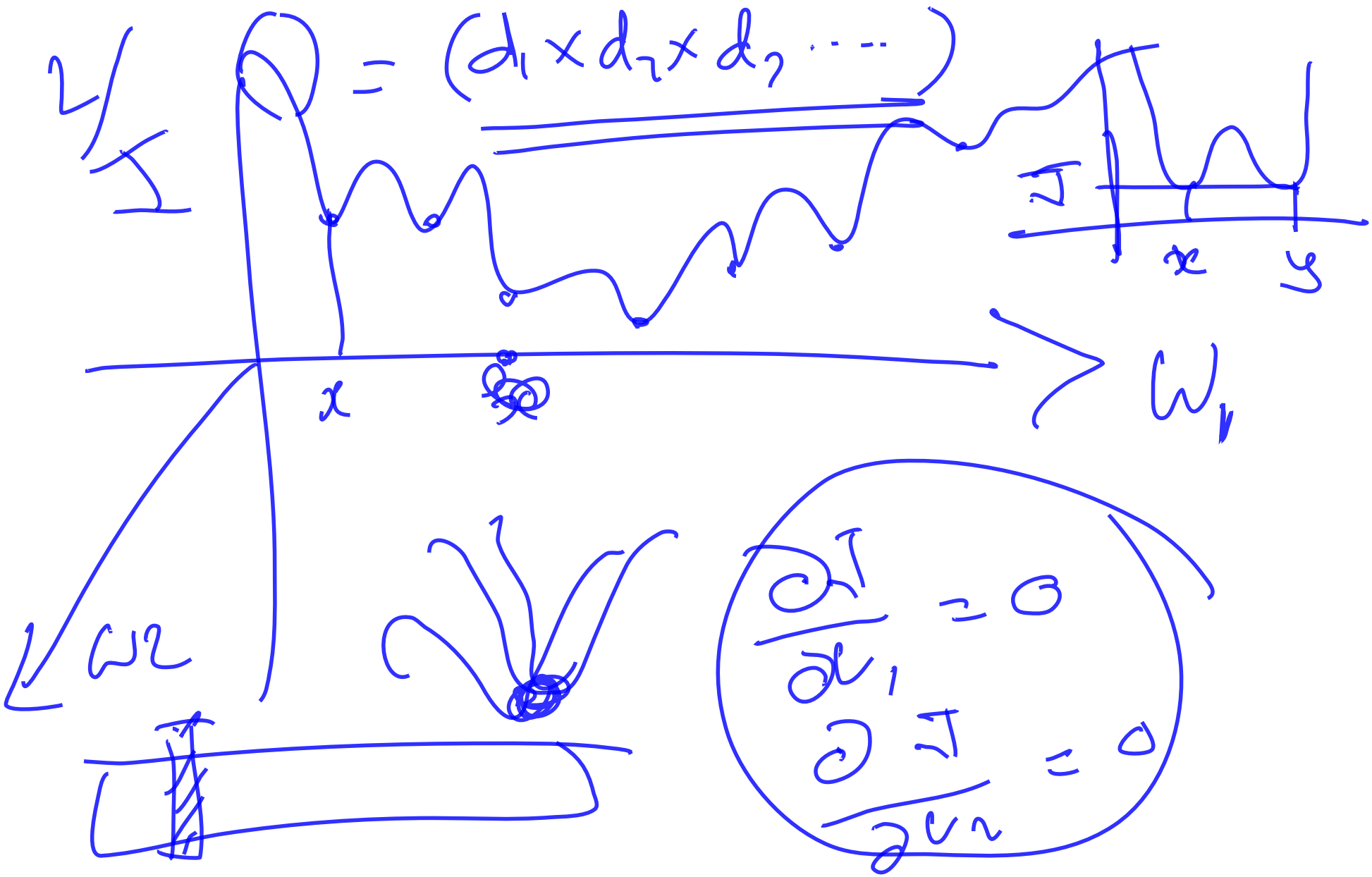


1. Acctivations in Neural Networks
   1. Sigmoid and Tanh
   2. ReLU and L-ReLu
2. Learning in NNs
   1. Local minima and non-convexity
   2. Plateaus and saddle points
   3. Vanishing and Exploding Gradients

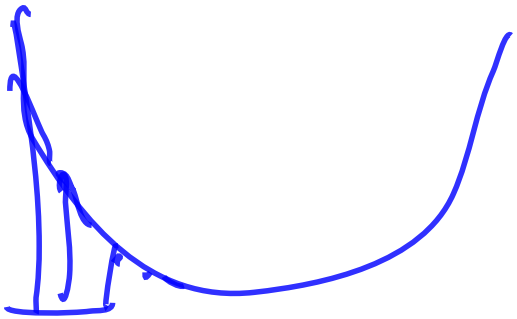
Questions? Comments?



Blank

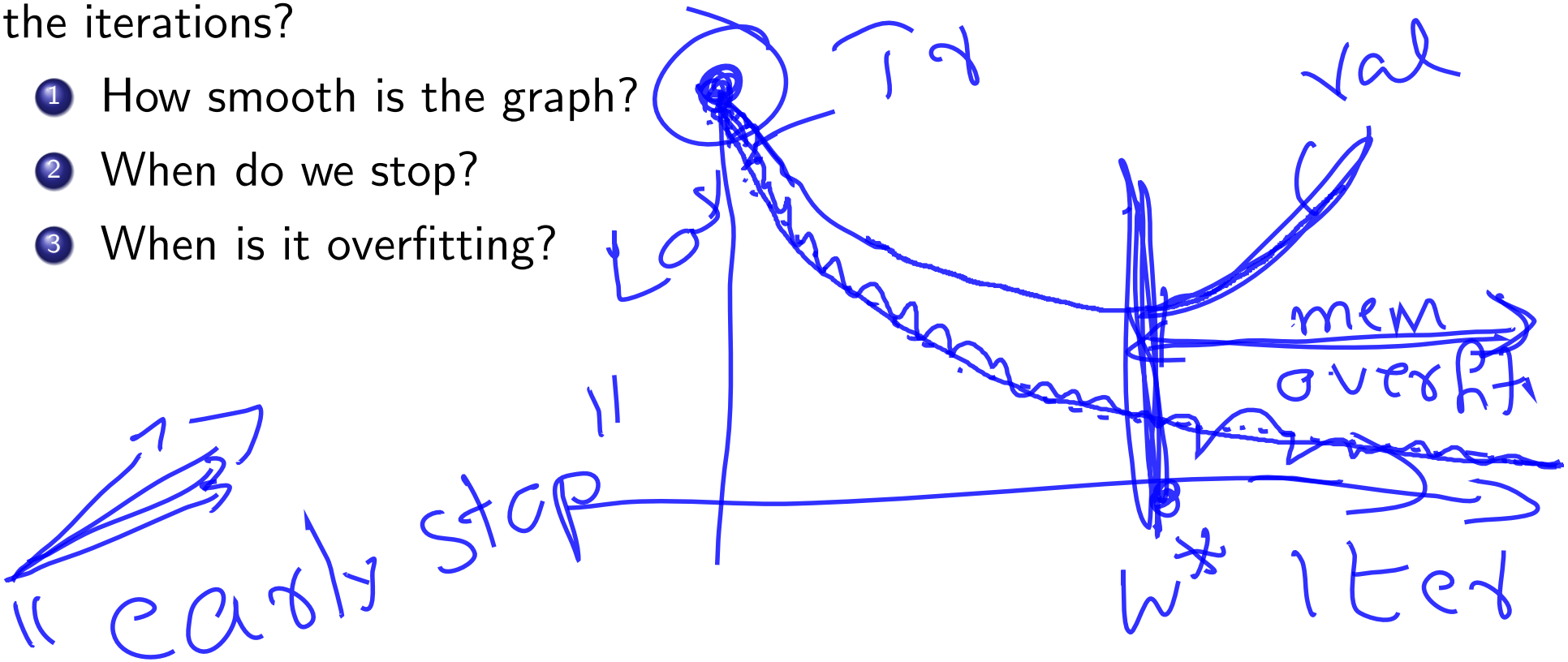

$\Omega = (d_1 \times d_2 \times d_3 \cdots)$
J
x
y
x
ω_1
ω_2
$\frac{\partial J}{\partial x_1} = 0$
$\frac{\partial J}{\partial x_2} = 0$

Discussions Point - I

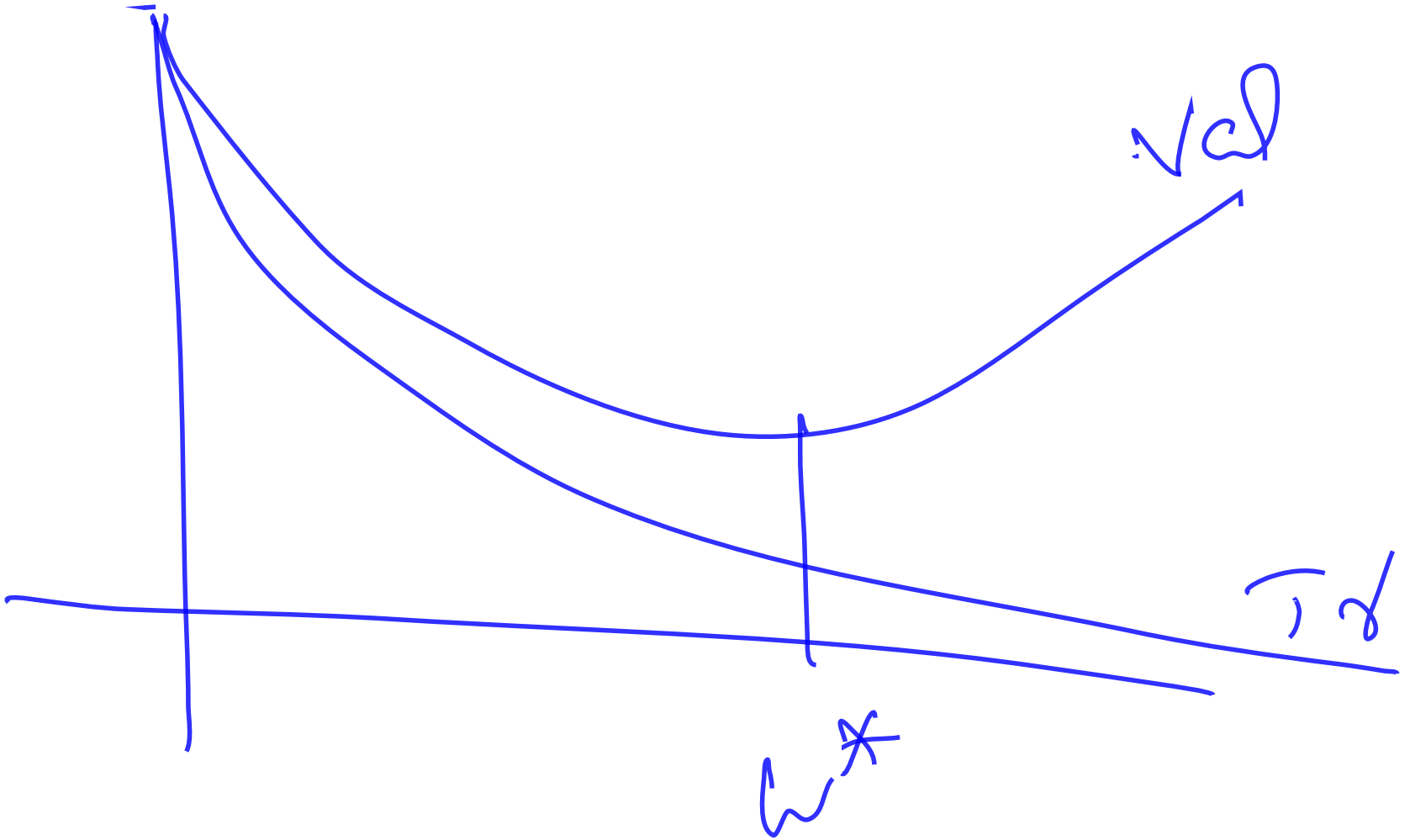


What happens to the NN Loss graphs on "Train" and "Val/Test" during the iterations?

1. How smooth is the graph?
2. When do we stop?
3. When is it overfitting?



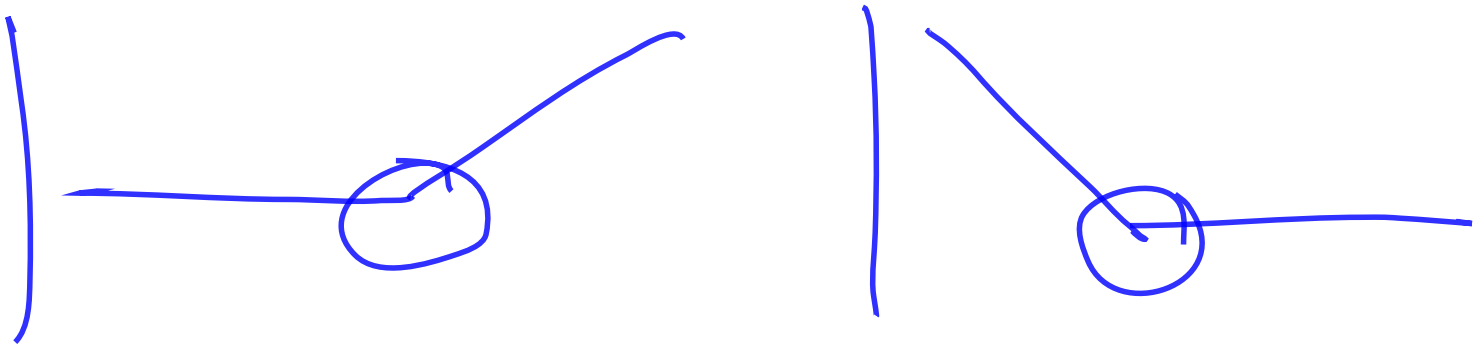
Blank



Blank

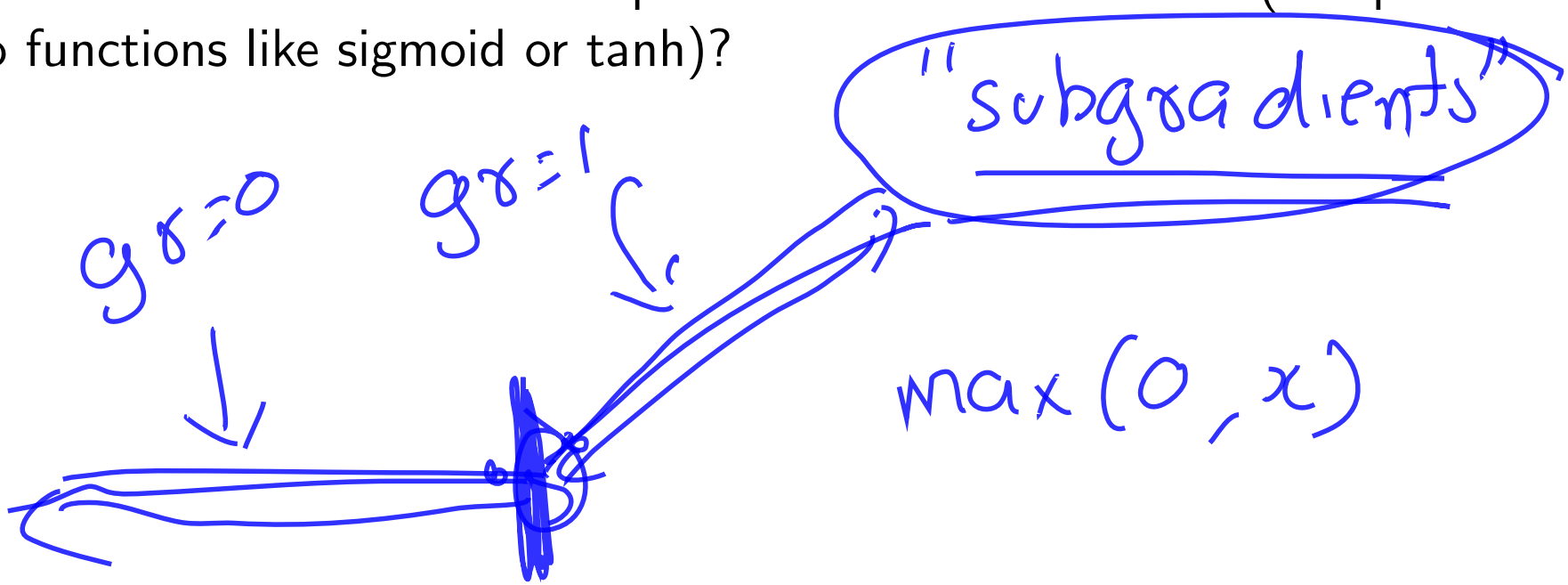
Blank

Discussions Point -II

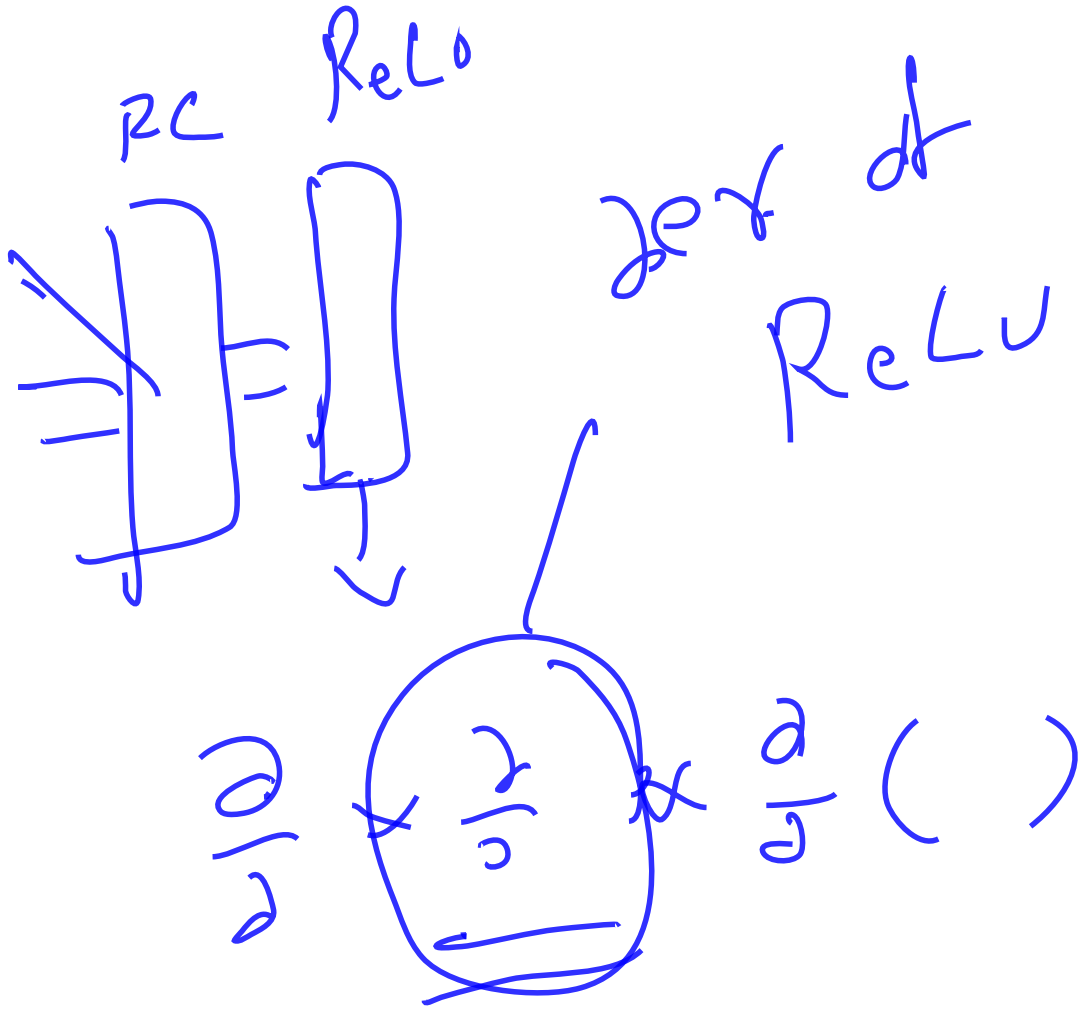


We know the functions like ReLu and Hinge Loss used in NN/ML.

- What makes these functions special for Gradient Descent (compared to functions like sigmoid or tanh)?



Blank



Blank

Discussion Point - III

(Advanced; Out of Syllabus!!) We know that a new kernel can be defined in terms of existing kernels:

$$\sum_{i=1}^{K} \alpha_i \kappa(\cdot, \cdot)$$

Then why don't we formulate the overall learning problem in SVM, including that of learning these α_i

1. Discuss why it is a good idea?
2. How do we use it for "fusing" different features?
3. Why do we limit to $\sum$?

See some of the works relevant[1] and [2]. Read later.

[1] http://manikvarma.org/pubs/varma07c.pdf
[2] https://cvit.iiit.ac.in/images/ConferencePapers/2009/Rakesh09More.pdf

Blank

Blank

Discussion Point - IV

(Advanced; Out of Syllabus!!)
We know that linear SVMs are superefficient (compared to K-SVMs).
Can we find a $\phi()$ corresponding to a Kernel and solve the problem as

$$\mathbf{w}^T \phi(x)$$

Indeed, this may become difficult for many kernels (eg. RBFs). **why?**
Can we find a finite dimensional approximation of $\phi()$? How does it help in speeding up SVM with no major reduction in accuracy?
read [3] and [4] later.

1. Discuss why it is a good idea?
2. Suggest an application where speed matters (eg. in the reference is that ofobject detection).

[3] https://cvit.iiit.ac.in/images/ConferencePapers/2010/Sreekanth10Generalized.pdf
[4] https://www.robots.ox.ac.uk/ vgg/publications/2011/Vedaldi11/vedaldi11.pdf

Blank

Blank

What Next:?

1. NN Architectures and NN Learning
2. Programming for Deep Learning.