Neural Networks for Signal Processing-I

Prof. Shayan Srinivasa Garani

Department of Electronic System Engineering

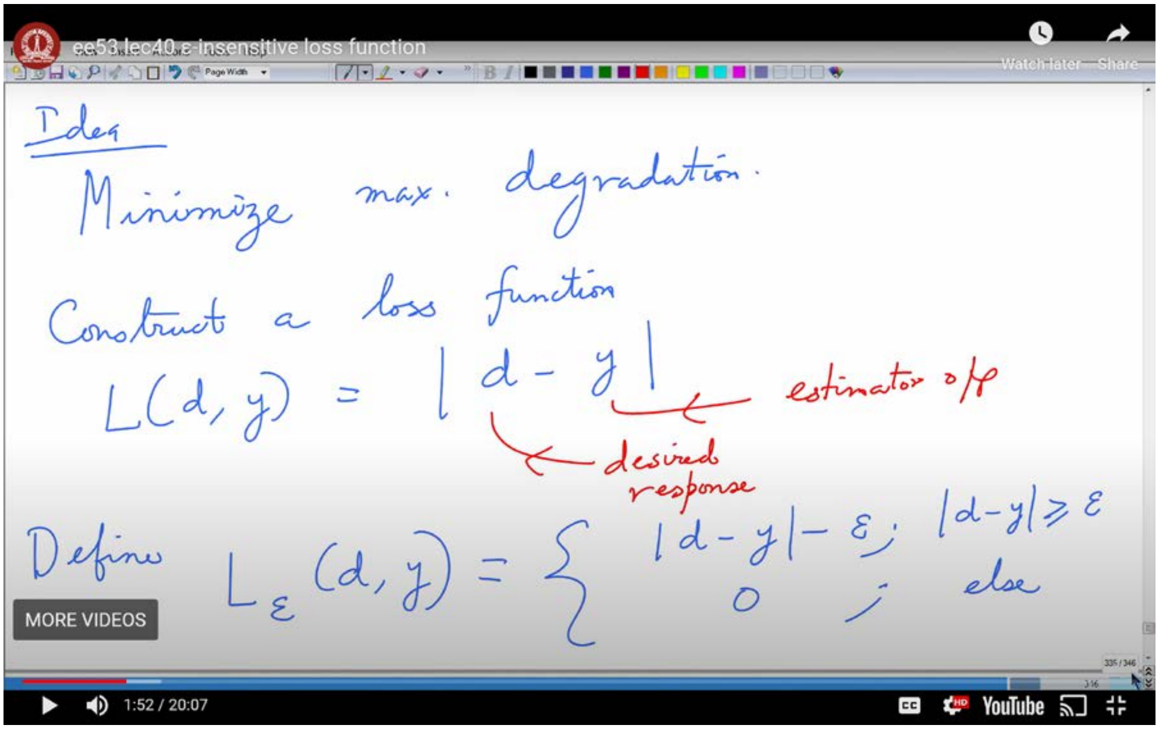
Indian Institute of Science, Bengaluru

Lecture – 40

ε-Insensitive Loss Function

Support Vector Machines (SVMs) can indeed be applied to non-linear regression problems. To effectively set up and solve these regression problems using SVMs, it's essential to understand the concept of epsilon-insensitive loss functions. Let's explore how this can be done.

(Refer Slide Time: 01:52)



Idea

Minimize max. degradation.

Construct a loss function

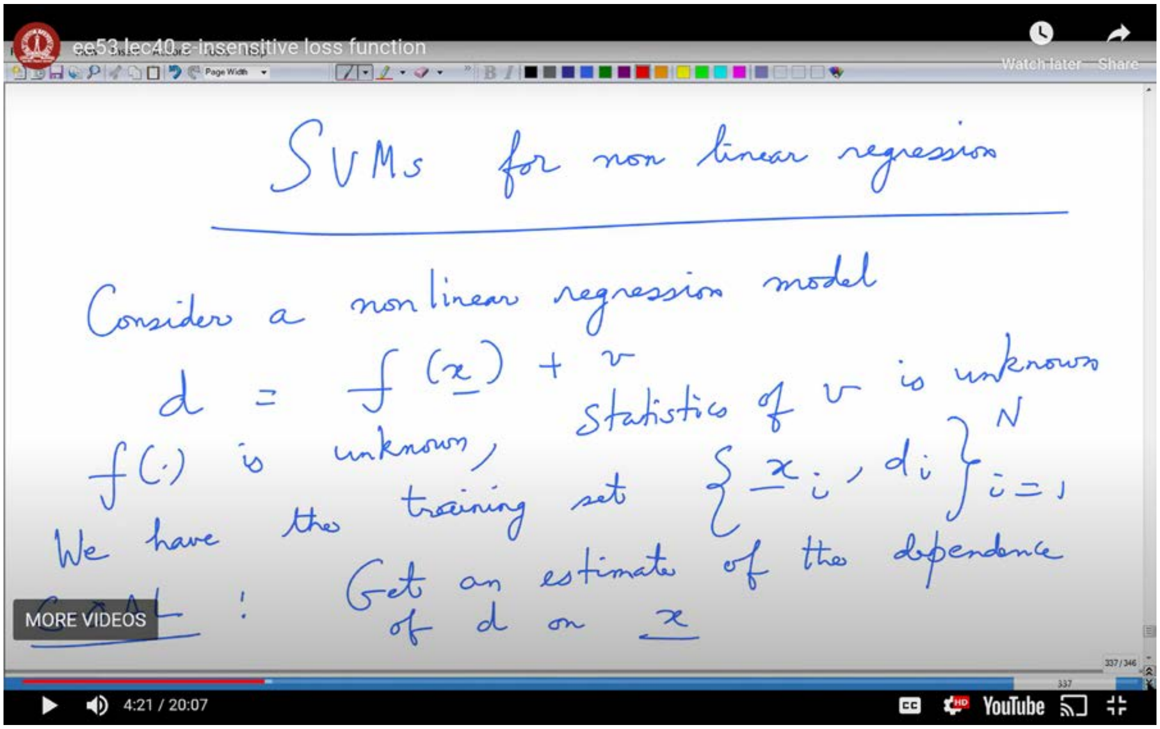
$$L(d, y) = |\underbrace{d}_{\text{desired response}} - \underbrace{y}_{\text{estimator o/p}}|$$

Define $L_\varepsilon(d, y) = \begin{cases} |d-y| - \varepsilon ; & |d-y| \geq \varepsilon \\ 0 ; & \text{else} \end{cases}$

Our goal is to develop a robust estimator for regression that is insensitive to small fluctuations in the model, essentially a non-parametric approach where specific parameters are not predefined.

One major issue with traditional estimators, such as the least squares estimator, is its sensitivity to outliers. This sensitivity can lead to poor performance, especially with distributions that have long tails. The squared error metric tends to amplify errors, particularly when outliers are present. In contrast, using absolute error can mitigate this problem. Therefore, the objective here is to minimize the maximum degradation in performance, which can be framed as a min-max optimization problem.

(Refer Slide Time: 04:21)



SVMs for non linear regression

Consider a non linear regression model

$d = f(\underline{x}) + v$ statistics of v is unknown

$f(\cdot)$ is unknown,

We have the training set $\{\underline{x}_i, d_i\}_{i=1}^{N}$

Goal : Get an estimate of the dependence of d on $\underline{x}$

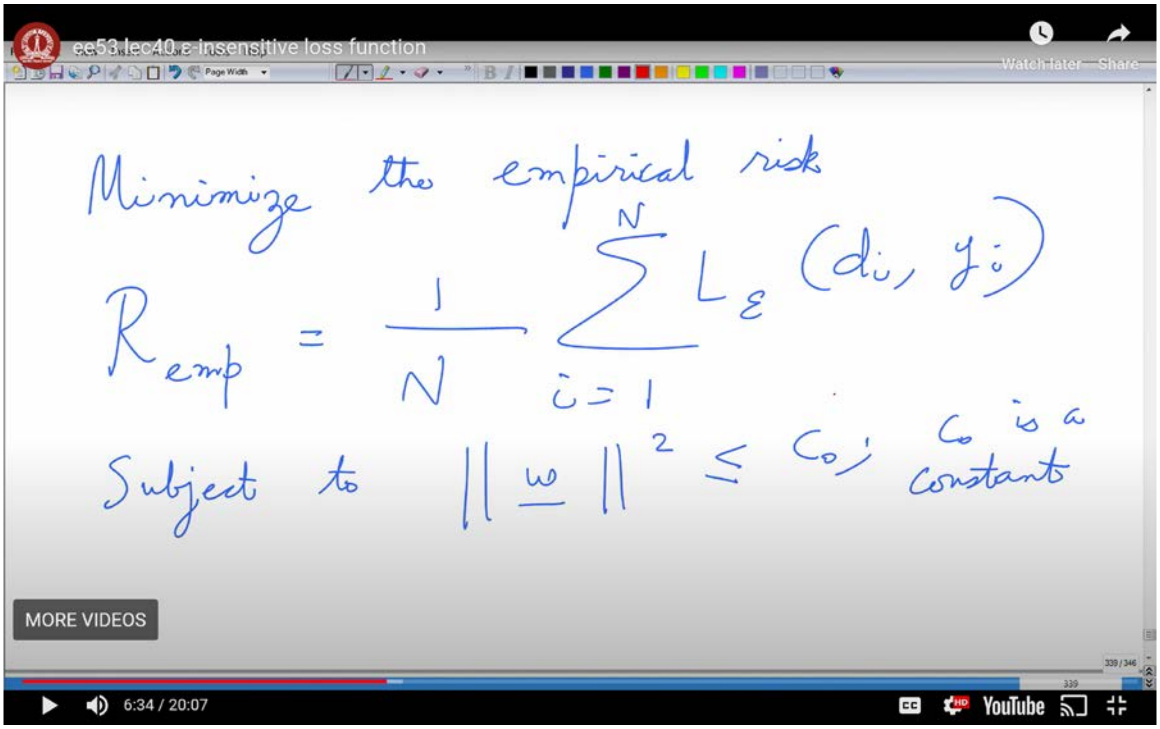
To address this, we construct a loss function L(d, y), where d is the desired response and y is the estimator's output. We define this loss function as |d - y|. From this, we derive the epsilon-insensitive loss function. The epsilon-insensitive loss function operates as follows: if the absolute difference |d - y| is within an epsilon tolerance, the loss is zero; otherwise, the loss is |d - y| - ϵ. This can be visualized as a "bathtub" function. Within the interval from -ϵ to +ϵ, the loss is zero; outside this interval, the loss increases linearly with |d - y| - ϵ. This defines an epsilon-insensitive region where the estimator is robust to small deviations, effectively disregarding errors within the range of -ϵ to +ϵ.

Having established the epsilon-insensitive loss function, we can now apply this to SVMs for non-linear regression problems. Consider the non-linear regression model:

$$d = f(x) + v$$

where f is an unknown function, x is the input vector, v represents noise with unknown statistics, and d is the observed response. Given a training set consisting of pairs (x_i, d_i) for i = 1 to n, our goal is to estimate the function f that describes the dependence of d on x.

(Refer Slide Time: 06:34)



Minimize the empirical risk

$$R_{emp} = \frac{1}{N} \sum_{i=1}^{N} L_{\varepsilon}(d_i, y_i)$$

Subject to $\| \underline{w} \|^2 \leq C_0$; C_0 is a constant

The approach involves using the epsilon-insensitive loss function to handle the non-linear regression problem effectively with SVMs, thereby ensuring robustness against small errors and deviations in the model.

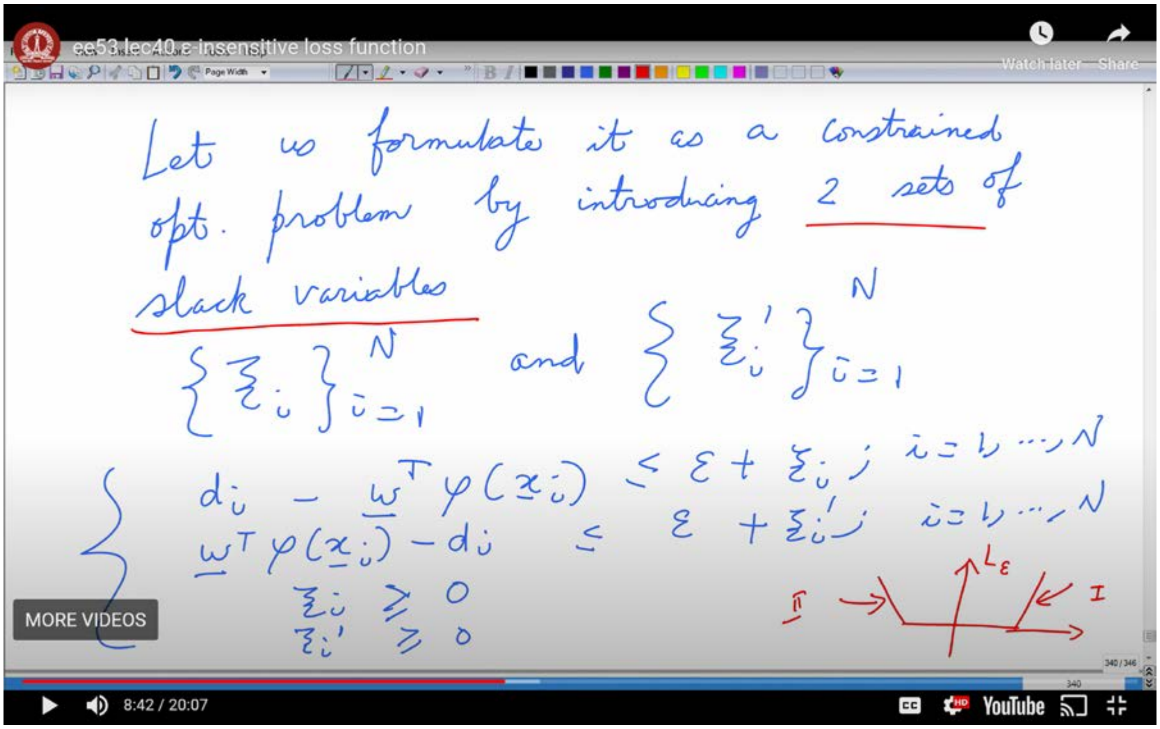
In tackling non-linear regression problems with SVMs, we use a linear combination of non-linear basis functions. This approach leverages the idea of incorporating non-linear functions over points in the input space to guide our formulation.

Let's consider y defined as:

$$y = \sum_{j=0}^{m} w_j \Phi_j(x)$$

Here, y represents the output, where w_j are the weights and $\varphi_j(x)$ are the non-linear basis functions applied to the input x. This can be seen as the inner product between the weight vector w and the vector of non-linear basis functions φ(x). In this context, w is a weight vector including w_0 to w_{m-1}, with w_0 often representing the bias term. The function φ(x) comprises $\varphi_0(x)$ to $\varphi_{m-1}(x)$, where $\varphi_0(x) = 1$ to account for the bias.

(Refer Slide Time: 08:42)



Let us formulate it as a constrained opt. problem by introducing 2 sets of slack variables

$\{\xi_i\}_{i=1}^{N}$ and $\{\xi_i'\}_{i=1}^{N}$

$$\begin{cases} d_i - \underline{w}^T \varphi(\underline{x}_i) \le \varepsilon + \xi_i \,; & i = 1, \dots, N \\ \underline{w}^T \varphi(\underline{x}_i) - d_i \le \varepsilon + \xi_i' \,; & i = 1, \dots, N \\ \xi_i \ge 0 \\ \xi_i' \ge 0 \end{cases}$$

Our goal is to minimize the empirical risk given by:

$$\frac{1}{n} \sum_{i=1}^{n} \text{loss}(d_i, y_i)$$

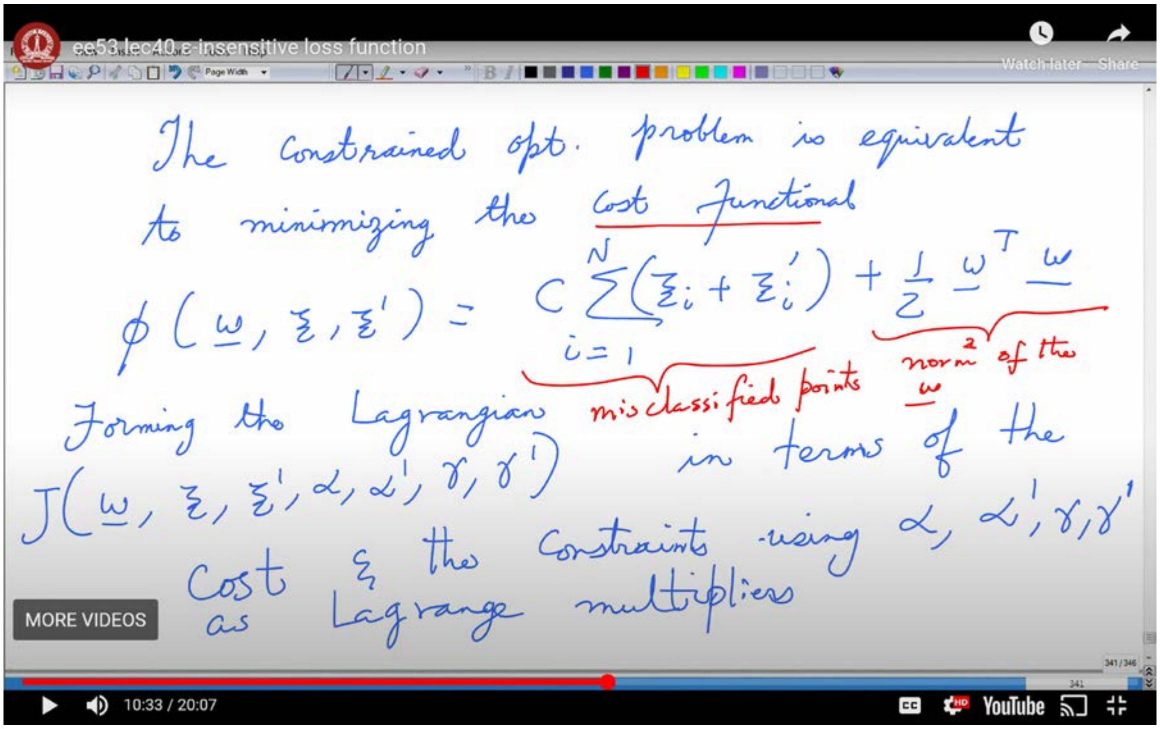
where $\text{loss}(d_i, y_i)$ is the epsilon-insensitive loss function. We aim to minimize this risk subject to the constraint that the norm of the weight vector is less than or equal to a fixed constant C_0.

To solve this optimization problem, we introduce two sets of slack variables, ζ_i and ζ_i', for i = 1 to n. These slack variables help handle violations of the constraints. The constraints are as follows:

1. $\zeta_i \geq 0$ and $\zeta_i' \geq 0$ (non-negativity of slack variables).

2. Consistent with the epsilon-insensitive loss function, d - y should be greater than ϵ or d - y - ϵ should be positive. This translates into constraints that ensure the deviation from the epsilon-insensitive interval is properly accounted for.

(Refer Slide Time: 10:33)



The constrained opt. problem is equivalent to minimizing the cost functional

$$\phi(\underline{w}, \xi, \xi') = \underbrace{C\sum_{i=1}^{N}(\xi_i + \xi_i')}_{\text{misclassified points}} + \underbrace{\frac{1}{2}\underline{w}^T\underline{w}}_{\text{norm}^2 \text{ of the } \underline{w}}$$

Forming the Lagrangian

$$J(\underline{w}, \xi, \xi', \alpha, \alpha', \gamma, \gamma')$$

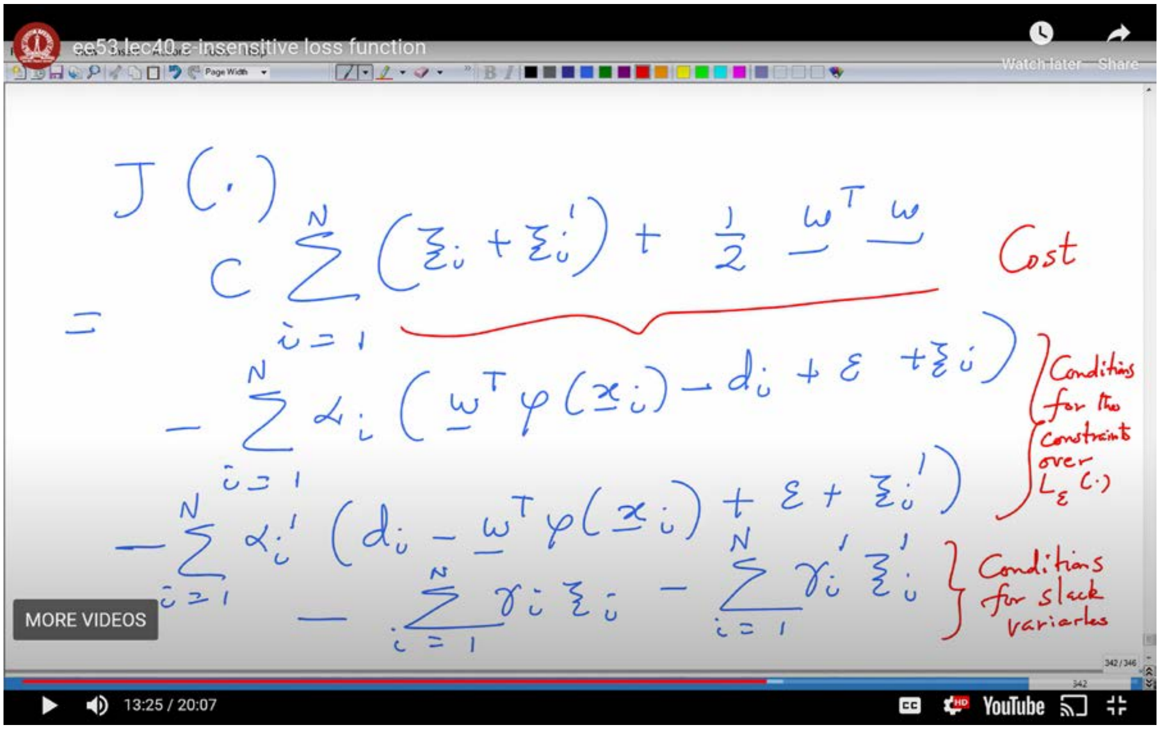
in terms of the cost & the constraints using $\alpha, \alpha', \gamma, \gamma'$ as Lagrange multipliers

We express these constraints with inequalities:

- For the case where d - y exceeds ϵ, and
- For the case where y - d exceeds ϵ.

Next, we formulate the cost functional. The first term of the cost functional is designed to minimize the number of misclassified points. This idea is similar to our approach for linearly separable patterns, where we formulated an average risk and set up an optimization problem. The second term of the cost functional is the squared norm of the weight vector, which we seek to minimize to maximize the margin of separation.

(Refer Slide Time: 13:25)



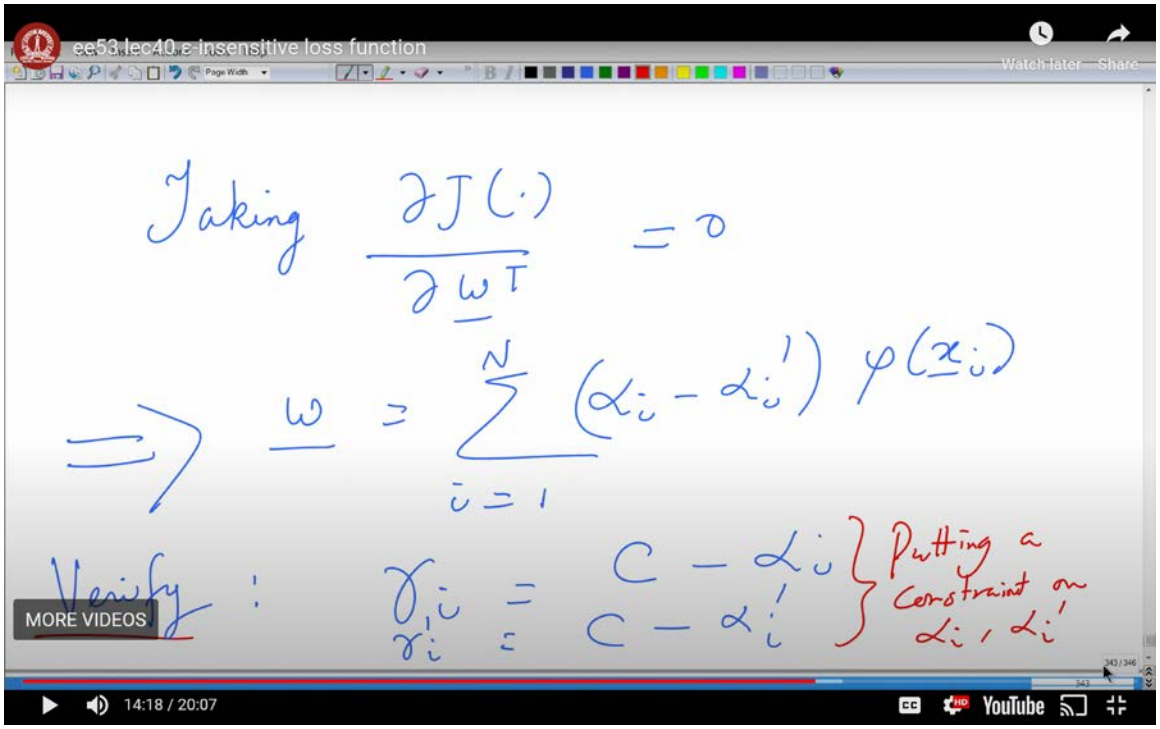
$$
\begin{aligned}
J(\cdot) = {} & \underbrace{C\sum_{i=1}^{N}(\xi_i+\xi_i') + \frac{1}{2}\,\underline{w}^T\underline{w}}_{\text{Cost}} \\
& \left.\begin{aligned}
& - \sum_{i=1}^{N}\alpha_i\left(\underline{w}^T\varphi(\underline{x}_i) - d_i + \varepsilon + \xi_i\right) \\
& - \sum_{i=1}^{N}\alpha_i'\left(d_i - \underline{w}^T\varphi(\underline{x}_i) + \varepsilon + \xi_i'\right)
\end{aligned}\right\} \text{Conditions for the constraint over } L_\varepsilon(\cdot) \\
& \left.- \sum_{i=1}^{N}\gamma_i\xi_i - \sum_{i=1}^{N}\gamma_i'\xi_i'\right\} \text{Conditions for slack variarles}
\end{aligned}
$$

Thus, the cost functional includes both minimizing the empirical risk and the norm of the weight vector. We aim to minimize this cost subject to constraints involving slack variables, Lagrange parameters, and the non-linear transformation of the input vectors. The Lagrangian for this problem incorporates the weight vector w, the slack variables ζ and ζ', and potentially all variables i from 1 to n can be combined into a vector for more efficient computation.

Similarly, we introduce the Lagrange multipliers α and α', as well as γ and γ', which correspond to the constraints on the slack variables. These slack variables must be non-negative, hence the need for Lagrange multipliers to enforce this constraint.

To proceed, we need to set up the Lagrangian for the optimization problem and then optimize with respect to the various variables within it. This involves taking partial derivatives of the Lagrangian with respect to all variables, setting those derivatives to zero, and solving the resulting equations.

(Refer Slide Time: 14:18)



Taking $\frac{\partial J(\cdot)}{\partial \underline{w}^T} = 0$

$$\Rightarrow \underline{w} = \sum_{i=1}^{N} (\alpha_i - \alpha_i') \varphi(\underline{x}_i)$$

Verify : $\left.\begin{array}{l} \gamma_i = C - \alpha_i \\ \gamma_i' = C - \alpha_i' \end{array}\right\}$ Putting a constraint on α_i, α_i'

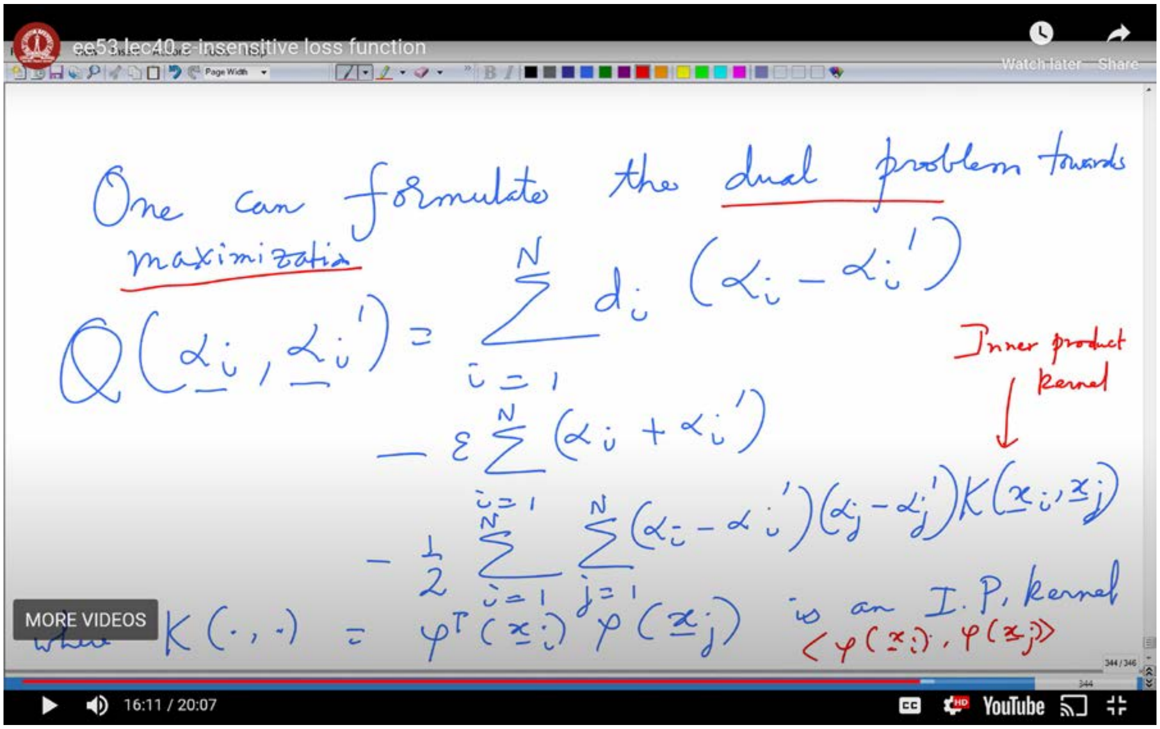
The cost functional J is quite complex, consisting of six terms. The first two terms pertain to the cost itself, while the next two terms are related to the constraints involving the Lagrange multipliers α and α', which link the weight vector w, the basis functions φ, and the responses d for the two regions in the epsilon-insensitive loss function. The final two terms address the constraints on the slack variables.

In essence, the last four terms of J are constraints related to the epsilon-insensitive loss function, while the first two terms correspond to the cost component.

To optimize this Lagrangian, we take the derivative with respect to the weight vector w and set it to zero. This yields the equation:

$$w = \sum_{i=1}^{n} (\alpha_i - \alpha_i')\phi(x_i)$$

(Refer Slide Time: 16:11)



One can formulate the dual problem towards maximization

$$Q(\underline{\alpha}_i, \underline{\alpha}_i') = \sum_{i=1}^{N} d_i (\alpha_i - \alpha_i') - \varepsilon \sum_{i=1}^{N} (\alpha_i + \alpha_i') - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} (\alpha_i - \alpha_i')(\alpha_j - \alpha_j') K(\underline{x}_i, \underline{x}_j)$$

Inner product kernel

where $K(\cdot, \cdot) = \varphi^T(\underline{x}_i) \varphi(\underline{x}_j)$ is an I.P. kernel $\langle \varphi(\underline{x}_i), \varphi(\underline{x}_j) \rangle$

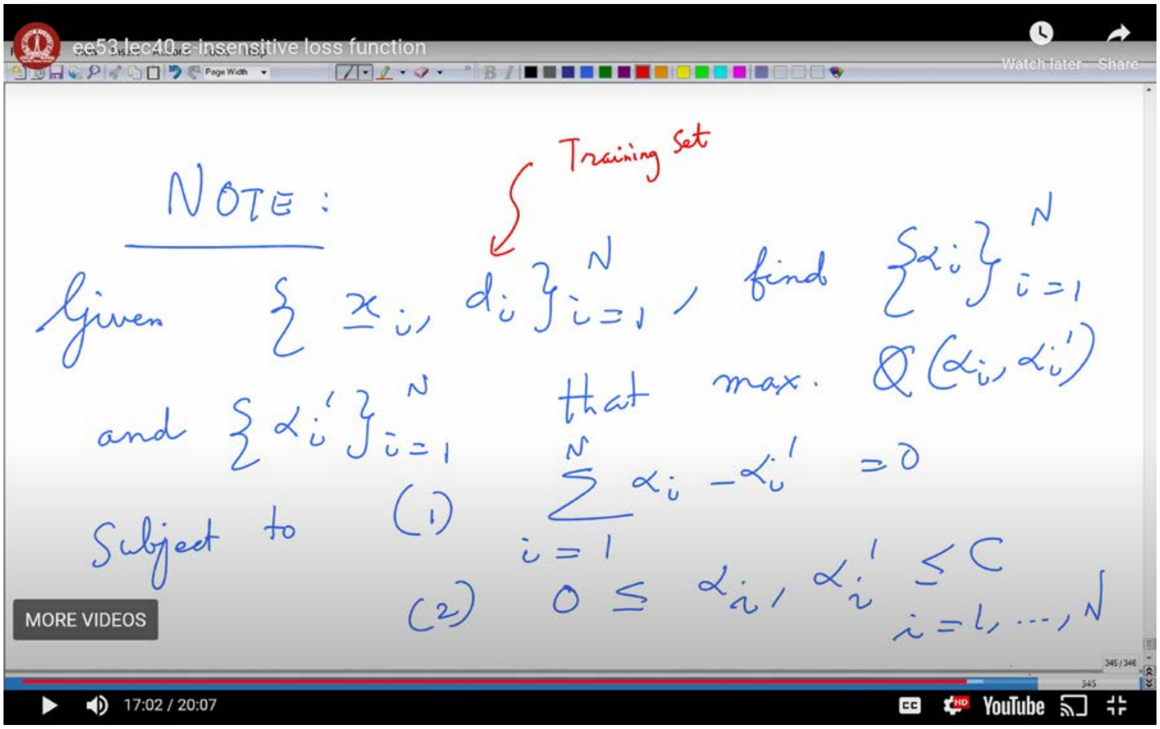
This result shows that the multipliers γ_i and γ_i' are related to α_i and α_i' through a constant C, which imposes constraints on α_i and α_i'. If the connection isn't immediately clear, it will be explored further in the course homework.

Next, we formulate the dual problem, which involves maximizing a dual cost function Q over the Lagrange multipliers α_i and α_i'. This dual function includes the inner product kernel $k(x_i, x_j)$. The formulation of this dual problem is straightforward once we compute the partial derivatives of the Lagrangian from the primal problem with respect to the parameters.

I integrate this into the primal problem and then rearrange it to obtain the dual problem. In the dual problem, we aim to maximize a function, which is essentially the negative of the cost from the primal problem. This kernel function represents the inner product between two functions, specifically the inner product of $\varphi(x_i)$ and $\varphi(x_j)$.

Given this dual problem, it's crucial to understand its significance. We are provided with the training data set, which consists of pairs (x_i, d_i) where i ranges from 1 to n. Our task is to determine the Lagrange multipliers α_i and α_i' that maximize the dual cost function while satisfying the specified constraint equations.

(Refer Slide Time: 17:02)



NOTE :

Training set

Given $\{\underline{x}_i, d_i\}_{i=1}^{N}$, find $\{\alpha_i\}_{i=1}^{N}$ and $\{\alpha_i'\}_{i=1}^{N}$ that max. $Q(\alpha_i, \alpha_i')$

Subject to (1) $\sum_{i=1}^{N} \alpha_i - \alpha_i' = 0$

(2) $0 \le \alpha_i, \alpha_i' \le C$
$i = 1, \dots, N$

Once we have computed the optimal weight vector w, we can then derive the regression function f given by w^T φ(x). This function can be compactly expressed as:

$$f(x) = \sum_{i=1}^{n} (\alpha_i - \alpha_i')k(x_i, x)$$

Here, the only free parameters are ϵ and C. ϵ is related to the epsilon-insensitive loss function, and C is a scaling constant used for summing the slack variables. These parameters are not optimized within the framework but are set as part of the problem setup.

After formulating and solving the optimization problem using an optimization solver, we obtain the optimal hyperplane. From this hyperplane, we derive the optimal weight vector, which allows us to compute the regression function.

This problem, particularly in the context of non-linear regression, involves multiple constraints, two related to the epsilon-insensitive loss function and two concerning the slack variables. Additionally, the cost function includes terms for minimizing the weight vector norm and the misclassification error. Consequently, solving this problem involves handling a somewhat complex optimization task.

With this module, you should now have a clearer understanding of how to set up and solve these optimization problems in general. Next, we will revisit the XOR problem and explore how SVMs can be applied to solve it through an illustrative example.