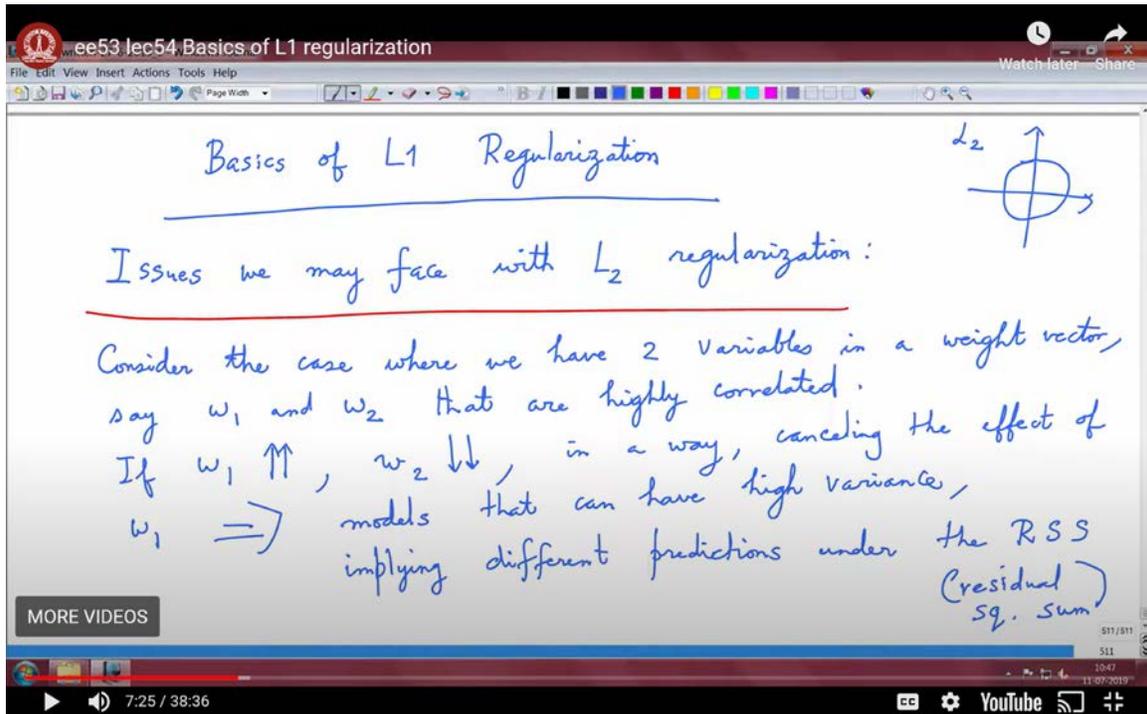


Neural Networks for Signal Processing-I
Prof. Shayan Srinivasa Garani
Department of Electronic System Engineering
Indian Institute of Science, Bengaluru

Lecture – 54
Basics of L1 Regularization

Let's delve into the basics of L_1 regularization and some related algorithms. Previously, we discussed the necessity of setting up a cost functional with regularization constraints and explored Tikhonov regularization, which transforms an ill-posed problem into a well-posed one. But what type of constraints should we impose? The choice depends on the problem's geometry.

(Refer Slide Time: 07:25)



The screenshot shows a video player interface for a lecture slide. The slide title is "Basics of L1 Regularization". To the right of the title is a hand-drawn diagram of a circle with a vertical and a horizontal axis, labeled L_2 . Below the title, the text reads: "Issues we may face with L_2 regularization:". This is followed by a paragraph: "Consider the case where we have 2 variables in a weight vector, say w_1 and w_2 that are highly correlated. If $w_1 \uparrow$, $w_2 \downarrow$, in a way, canceling the effect of $w_1 \Rightarrow$ models that can have high variance, implying different predictions under the RSS (residual sq. sum)". The video player shows a progress bar at 7:25 / 38:36 and a YouTube logo at the bottom right.

For instance, if our cost function involves minimizing the sum of the squares of the weights, we're essentially working with the induced L_2 norm. Minimizing this norm aligns with our

goal in the L_2 sense. However, this may not always be the constraint we're interested in. Sometimes, we might want to focus on minimizing the number of non-zero elements in our weight vector, which leads us to L_1 regularization. This approach is useful when we aim to achieve sparsity in our model.

The choice of regularity constraints is therefore influenced by the problem's geometry and practical considerations. With this in mind, we'll now formally introduce L_1 regularization, discuss its motivation, and explore how to set up both unconstrained and constrained optimization frameworks. We'll also examine some algorithms related to these concepts.

(Refer Slide Time: 10:56)

ee53 lec54 Basics of L1 regularization

We considered the regression problem earlier

$$X := [x_{ij}]_{n \times k} \quad \underline{x}_i \in \mathbb{R}^k$$

$\underline{y}_{n \times 1}$ data response

$$\underline{w} = (X^T X + \lambda I)^{-1} X^T \underline{y} \quad (\text{min-norm soln})$$

Cost functional: $\|X \underline{w} - \underline{y}\|_2^2 + \lambda \|\underline{w}\|_2^2$

(a) \underline{w} must not be too large

Balance the large variables in \underline{w} meeting the target cost.

MORE VIDEOS

10:56 / 38:36

Firstly, let's address some issues associated with L_2 regularization. Consider a weight vector with two variables, w_1 and w_2 . Suppose w_1 and w_2 are highly correlated. If w_1 increases, w_2 might decrease in such a way that it cancels out the effect of w_1 . This correlation can lead to models with high variance, meaning that predictions can vary significantly under different conditions, particularly when evaluated using the Residual Sum of Squares (RSS) metric.

The L_2 norm has a specific geometric interpretation: it operates within a ball. When regularizing with L_2 , you constrain the optimization to lie inside or outside a circular boundary. This geometric constraint influences how the problem is approached and solved within the L_2 framework.

The challenge we're facing is that L_2 regularization does not account for model sparsity. This means that variables which may be irrelevant to the problem or those that are highly correlated can lead to one variable offsetting the effect of another. To address this, we need a different geometric approach, which is where L_1 regularization comes into play.

(Refer Slide Time: 16:36)

L_2 norm

- 1) does not account for the parsimony of the model
i.e., sparsity constraints are not taken into account.
- 2) L_2 models may have non-zero values associated with
inconsequential variables.

Costs involving L_1 penalty impose
sparsity constraints $\|w\|_1$

MORE VIDEOS

16:36 / 38:36

Let's revisit our regression problem. Given a data matrix X , which is $n \times k$ where each vector x_i belongs to R^k , and a data response vector y which is $n \times 1$, our goal is to find the best fit line for this data while adhering to regularity constraints. The solution to this line corresponds to the minimum norm solution, which we have previously discussed.

In the L_2 norm setup, we aim to minimize $\|Xw - y\|_2^2$, subject to a constraint that the norm of the weight vector w should not be too large. This approach balances the weight variables,

ensuring they do not exceed certain limits while meeting the target cost. Essentially, we want to keep the weights controlled to avoid any variable from becoming excessively large.

However, the problem with L_2 regularization is that it can lead to issues when dealing with correlated variables. For instance, if one variable increases while another decreases in a way that cancels out the effect, it can result in models with high variance, leading to inconsistent predictions under different conditions. While L_2 regularization handles these issues smoothly because it results in a convex optimization problem, it does not necessarily lead to a sparse or parsimonious model.

(Refer Slide Time: 19:56)

ee53.lec54.Basics of L1 regularization

1) If the data matrix $X_{(n \times k)}$ has irrelevant features, L_1 seems to be better than $L_2 \Rightarrow$ low variance feature selection

2) L_1 can yield a better variable/attribute selection

- a) Simplification of models for interpretability.
- b) Shorter training times
- c) Avoid the problem of overfitting \Rightarrow avoid the curse of dimensionality

MORE VIDEOS

19:56 / 38:36

YouTube

Let's consider practical constraints, such as in neural networks where we use backpropagation and other algorithms. If we encounter dead neurons or insignificant synaptic weights, keeping these weights is impractical due to computational complexity and memory constraints. Storing and computing with all these weights is both memory-intensive and burdensome. Hence, L_2 regularization, which does not encourage sparsity, may not be ideal for such practical scenarios.

To recap, L₂ norm does not account for model sparsity, meaning it does not address the issue of having many non-zero values in inconsequential variables. For instance, in the case of analyzing DNA strands, we might have hundreds of features, but only a subset of these features may be relevant for inferring gene expression. Identifying which variables to retain and which to discard is crucial and highly specific to the problem at hand.

Thus, L₁ regularization is motivated by the need to handle such issues, promoting sparsity in the model and thus addressing the limitations of L₂ regularization.

Let's dive into the scenario where we want to impose L₁ regularization constraints on the weight vector, focusing on the L₁ penalty which promotes sparsity. In L₁ regularization, you take the absolute value of each weight, sum these absolute values, and ensure that this sum is less than a pre-specified number.

(Refer Slide Time: 24:02)

Unconstrained formulation

$$\min_w \left\| Xw - y \right\|_2^2 + \lambda \left\| w \right\|_1 \quad \text{--- (1)}$$

$\lambda \left\| w \right\|_1$ is the L_1 norm

Clearly (1) has issues of differentiability @ the origin

$$\left\| w \right\|_1 = |w_1| + |w_2| + \dots + |w_k|$$

$|x|$ is not differentiable @ $x=0$.

In practical terms, if our data matrix X contains irrelevant features, L₁ regularization is often more effective than L₂. This is because L₁ regularization promotes feature selection

by driving some coefficients to exactly zero, which in turn simplifies the model. The benefits of better variable or attribute selection through L₁ regularization are significant:

1. Model Simplification for Interpretability: A simpler model is easier to interpret, allowing us to make clearer observations and understand the data better.
2. Reduced Training Times: Simplified models often lead to shorter training times, enhancing efficiency.
3. Mitigation of Overfitting and Curse of Dimensionality: By removing irrelevant features, L₁ regularization helps in avoiding overfitting and managing high-dimensional data more effectively.

These advantages highlight why L₁ regularization is preferable in certain scenarios over L₂.

Now, let's set up the unconstrained formulation. We aim to minimize the squared L₂ norm of $Xw - y$ over all possible choices of w , subject to the constraint imposed by L₁ regularization. Specifically, we are interested in minimizing:

$$\|Xw - y\|_2^2$$

subject to the L₁ norm of w being less than or equal to a certain bound. Let's call this Equation 1.

Previously, with L₂ regularization, we minimized $\|Xw - y\|_2^2$ subject to the norm of w being minimized, which led to a closed-form solution. For L₁ regularization, the situation differs. Here, the L₁ norm of w introduces sparsity constraints, making the geometry of the problem distinct.

In L₁ regularization, the critical aspect is the absolute value function, which impacts the differentiability of the objective function. Specifically, the L₁ norm is:

$$\text{norm}_{l_1} = |w_1| + |w_2| + \dots + |w_k|$$

Differentiating this norm with respect to each coordinate presents challenges because the absolute value function is not differentiable at zero. This lack of differentiability complicates finding a closed-form solution, unlike the smooth quadratic cost function in L_2 regularization.

(Refer Slide Time: 27:05)

Constrained Formulation

$$\min_w \|Xw - y\|_2^2 \quad \text{s.t.} \quad \|w\|_1 \leq t \quad \text{--- (2)}$$

↑
Chosen value

Non-differentiable constraints are converted to a set of linear constraints

⇒ Feasible region is a polyhedron!

MORE VIDEOS

27:05 / 38:36

In the constrained formulation, the only difference from the unconstrained case is that we now want to bound the L_1 norm of w within a specific limit. For example, we might constrain the norm of w such that:

$$\|w\|_1 \leq t$$

This subtle difference has significant implications for setting up and solving our optimization problem.

Let's consider the optimization problem where we want to minimize the squared Euclidean norm $\|Xw - y\|_2^2$ subject to the constraint that the L_1 norm of w is less than or equal to t , where t is a chosen constant. In this scenario, we are interested in the absolute values of

the weights in the weight vector w , and we require that their sum is constrained to be less than or equal to t . Essentially, this means:

$$|w|_1 = |w_1| + |w_2| + \dots + |w_k| \leq t$$

(Refer Slide Time: 29:24)

The image shows a screenshot of a video lecture slide. The slide has a white background with handwritten text in blue and red. At the top, the word "LASSO" is written in red and underlined. To its right, the word "ALGORITHMS" is written in blue and underlined. Below "LASSO", the full name "Least Absolute Selection and Shrinkage Operator." is written in blue, with each word underlined. Below this, the text "Consider solving the problem (2)." is written in blue. Underneath that, "Suppose we have 2 variables in w " is written in blue. At the bottom left of the slide, there is a button labeled "MORE VIDEOS". The video player interface at the bottom shows a play button, a volume icon, and a progress bar indicating 29:24 / 38:36. The YouTube logo and other interface elements are also visible.

This forms what we call a set of non-differentiable constraints, which we will refer to as Equation 2. To handle these non-differentiable constraints, we convert them into a set of linear constraints, which means that the feasible region for this problem becomes a polyhedron.

For a deeper understanding of this geometry, I recommend reviewing the lecture on norms from the course *Mathematical Methods and Techniques for Signal Processing*. This lecture provides detailed insights into the geometry of normed spaces, which will help clarify these concepts.

Now, with the regularization problem set up, we can discuss algorithms designed to solve such optimization problems. One prominent algorithm used in this context is LASSO,

which stands for Least Absolute Shrinkage and Selection Operator. This technique, popular in statistics, was developed by Tibshirani and is specifically designed to address problems involving L_1 regularization.

(Refer Slide Time: 31:27)

The screenshot shows a video player interface for a lecture titled "ee53 lec54 Basics of L1 regularization". The main content is handwritten text and a diagram. The text reads: "By considering the sign of w_1 and w_2 ". Below this, four inequalities are listed, grouped by a large right-facing curly bracket: $w_1 + w_2 \leq t$, $w_1 - w_2 \leq t$, $-w_1 + w_2 \leq t$, and $-w_1 - w_2 \leq t$. To the right of the inequalities is a 2D coordinate system with a vertical axis labeled w_2 and a horizontal axis labeled w_1 . A circle containing the letter 'I' is drawn in the lower-left quadrant of the axes. Below the diagram, the text "Home Work" is written and underlined, followed by the instruction: "Plot the constraints on $w_1 - w_2$ plane and show the feasible region." The video player controls at the bottom show a play button, a volume icon, a progress bar at 31:27 / 38:36, and the YouTube logo.

To motivate the use of LASSO, consider the constraint optimization problem given by Equation 2. Suppose we have two variables, w_1 and w_2 . We have four possible constraints based on the signs of w_1 and w_2 :

1. $w_1 + w_2 \leq t$

2. $w_1 - w_2 \leq t$

3. $-w_1 + w_2 \leq t$

4. $-w_1 - w_2 \leq t$

These constraints can be visualized in the w_1 - w_2 plane, resulting in a feasible region that can be plotted based on these inequalities. You will see lines with slopes of 1 and -1, creating a geometric shape in the plane.

I encourage you to plot these constraints yourself as an exercise. As a graduate student, you should be able to draw the feasible region defined by these inequalities and understand how the solution to the original cost function $\|Xw - y\|_2^2$ fits within this feasible region.

(Refer Slide Time: 34:32)

Any minimizer to the RSS subject to (1)
will minimize the cost (2)

Problem: If we have 'k' variables, we have 2^k constraints \Rightarrow exponential increase in complexity.

Over \mathbb{R}^{40} , 2^{40} are possible (Infeasible for optimization)

MORE VIDEOS

34:32 / 38:36

To address the problem of minimizing the squared residual sum under L_1 regularization constraints, we need to solve for the L_2 norm subject to the linear constraints that arise from the L_1 regularization. Specifically, we aim to minimize the L_2 norm squared of the residuals $\|Xw - y\|_2^2$ subject to the constraints imposed by the L_1 regularization. These constraints are captured by Equation 2, which defines the cost formulation we are optimizing.

Here's the challenge: With k variables, there can be 2^k possible linear constraints. This exponential growth in the number of constraints quickly becomes impractical. For example, in a 40-dimensional space, the number of constraints is 2^{40} , which is infeasible

for most optimization algorithms, even with advanced computational resources like cloud computing or multi-core processors.

(Refer Slide Time: 37:30)

ee561 lec54 Basics of L1 regularization

TIBSHIRANI'S APPROACH

Constraint Set = ϕ (In practice 't' can be small)

while ($\|w\|_1 \leq t$)

- Add sign(w) and fold this into the constraint set.
- Opt $\|Xw - y\|_2^2$ subject to the constraints.

MORE VIDEOS end while

37:30 / 38:36

To manage this complexity, consider the following approach inspired by Tibshirani's method. Start with an empty constraint set and iterate through a loop. While the L_1 norm of the weight vector w is less than or equal to t , add the signs of the weight vector w to the constraint set. Then, optimize the L_2 norm of the residuals $\|Xw - y\|_2^2$ subject to these constraints.

During each iteration of the loop:

1. Add the constraints based on the signs of w to the constraint set.
2. Optimize the L_2 norm under the updated constraints.

This process is repeated until all constraints are satisfied. Despite the theoretical feasibility, this method is computationally intensive due to the exponential number of constraints and the necessity to update constraints dynamically.

In practice, t can be small, but solving the problem efficiently requires careful handling of constraints and optimization iterations. As you add new constraints, the initial conditions of your previous solutions may no longer be applicable, leading to complex and potentially inefficient algorithms.

By carefully managing these iterations and constraints, we can ultimately find a weight vector w that satisfies all the constraints, providing a solution to the optimization problem. We'll discuss some practical considerations for this approach shortly.