Me: Solve question 2 for me

Gemini 3 pro: Show Thinking

Considering Optimization Problems

I'm currently focused on extracting the core problem statement from the image. It looks like we're dealing with a simplified Adam optimizer, denoted as $\mathcal{O}$, without weight decay. My next step is to understand the implications of this simplification for solving the complete problem outlined in Question 2, based on the image.

Clarifying Adam Update

I've been re-reading the provided text, and I've come to understand the Adam update rule more precisely. My analysis shows that, with a single data point and no weight decay, the Adam step essentially takes the sign of the gradient for each coordinate. I'm exploring how this simplified behavior impacts convergence in our example problem, specifically using the equation $[1, 0.1, 0.01]\theta = 1$ with a quadratic loss. I'm also working on setting an initial point for the gradient descent.

Determining SGD Solution

I'm now fully immersed in dissecting the problem posed in part (a). The challenge lies in pinpointing the specific θ to which vanilla SGD converges. It's a matter of identifying the minimum norm solution within the infinite set of solutions that satisfy the constraint. The core approach hinges on recognizing that the converged θ will be a scalar multiple of the data vector x. I'm crunching the math to find this scalar, and thus, the final solution.

Analyzing SGD Convergence

I've determined the analytical solution for θ_{SGD}. My calculations show that for the given data vector, the minimum norm solution is $\theta_{SGD} = \frac{1}{1.0101} [1, 0.1, 0.01]^T$. This result stems directly from the fact that SGD converges to a scalar multiple of the data vector. I've now moved on to investigating Adam's behavior.

Exploring Adam Dynamics

I'm now delving into the dynamics of Adam for this specific problem. My initial thought was considering the solution as a "minimum norm" solution, which led me to investigate the geometric transformation. It seems more like solving the problem in a transformed coordinate system with normalized features. I'm focusing on Adam's inherent feature scaling and how it impacts convergence. Specifically, I am thinking about how Adam, which essentially does steepest descent in L_∞ norm, arrives at the solution. I'm carefully examining the update rule with the moving averages removed. I will investigate why Adam's updates are essentially equivalent to steepest descent in the L_∞ norm with an equivalent scaling in the original space.

Revising Adam's Approach

I've revised my understanding of Adam's approach. In this scenario, with the simplified update, Adam moves in the direction of [1, 1, 1] until the hyperplane is reached. I've now confirmed the solution will be of the form $\theta = k [1, 1, 1]^T$ since each coordinate's update will be $\pm \delta$. This means k is $1/ \sum x_i$, solidifying the direct connection to the plane and equal-magnitude updates in each dimension. The solution is thus $[0.90909091, 0.90909091, 0.90909091]^T$ and I can now explain that Adam's convergence in this instance relies on its adaptation to feature scaling, effectively treating all features equally due to the sign-based update.

Connecting Feature Rescaling

I'm now fully connecting the concepts of feature rescaling and the Adam solution. I've calculated the required scaling matrix to normalize the second moment of each feature, leading to a new feature vector of all ones. Running SGD on this rescaled problem yields a solution that is directly linked to the original Adam result. I'm confirming that the scaling inherent in Adam, due to its sign-based updates, is equivalent to this explicit feature rescaling, clarifying the underlying mechanism for the convergence to a $[1, 1, 1]$ direction. I am now exploring the implications of the solution on the original unscaled features.

Confirming Final Theta

I'm now deriving the final solution for θ from the rescaled features, $\tilde{\theta}$. My calculations show that to convert $\tilde{\theta}$ (from rescaled space back to original units, I need to apply a scaling to θ determined by $x^T \theta = \tilde{x}^T \tilde{\theta}$, where $\tilde{x} = S x$, and then relate $\theta_i = \tilde{\theta}_i s_i$. With this I've confirmed that $\theta = \frac{1}{3} [1, 10, 100]^T$. I've arrived at the complete solution and understand the exact impact of Adam's transformation of the feature space.

Validating Solution's Equivalence

I'm now completely satisfied that the solution can be validated against various angles. I've re-examined the Adam behavior from an alternative angle, confirming that Adam's implicit feature scaling perfectly aligns with an explicit feature scaling approach. This leads to the same solution as the L_∞ optimization in the original space. I'm solidifying my understanding of this connection, and exploring different methods to double-check these conclusions.

Deriving Original Theta

I'm focusing now on the correct way to transform the solution back into the original space after the feature rescaling. My calculations showed that the solution for $\tilde{\theta}$ is

[1/3, 1/3, 1/3], and I am using this to find the corresponding θ. After some revisions and careful consideration, I have now confirmed the proper transformations and calculations that will arrive at the final solution for θ.

I've been working on the solution. I had a discrepancy between using σ_i and the scaling factor s_i. I've carefully reviewed the feature scaling transformation from the previous step. Specifically, if $\tilde{x} = S x$, then $x^T S \tilde{\theta} = x^T \theta$. I've determined that S is a diagonal matrix where each diagonal value is the inverse of the respective original feature value, allowing us to find $\theta_i = \tilde{\theta}_i / x_i$. This approach leads to $\theta = [1/3, 10/3, 100/3]^T$. The result satisfies the initial constraint.

Validating Feature Transformation

I'm now fully immersed in validating the feature transformation and its impact on the Adam solution. The core issue lies in ensuring the transformation from rescaled features to the original space maintains the correct relationship to the solution space. I'm focusing on the scaling matrix and its inverse to confirm that both approaches yield identical θ results. I'm now certain that any discrepancy results from the mis-application of the correct scaling matrix to transform θ.

Synthesizing Solution Insights

I'm now consolidating the solution and underlying principles into a clear presentation. The key is to elucidate the connection between Adam's implicit feature scaling and the resulting equal-coordinate weight distribution. I'll outline the steps of feature rescaling, SGD's behavior in the transformed space, and the subsequent inverse transformation back to the original space to derive the final θ. I will then present the analytical solution concisely.

Refining Solution Approach

I'm now zeroing in on Part (a) of Question 2. I need to formulate the minimum norm solution explicitly. For part (b), I'll detail the update simplification inherent in the Adam optimizer under the given conditions. And finally, for Part (c), I'm structuring a robust approach that clearly demonstrates the variable substitution. This will help simplify the problem into a clear solution.

Clarifying Question's Scope

I'm now fully focused on the first image, Question 2. I realize the second image may be irrelevant. The question specifies "all parts of question 2," so I will solely work on the constraints from the first image. The goal is to articulate the analytical solutions for θ from images, and provide a clear explanation for the transformation from one scenario to another.

Here are the solutions for all parts of Question 2.

Problem Setup

We are solving a linear regression problem with a single training point ($n = 1$).

- **Input:** $x = [1, 0.1, 0.01]^T$
- **Target:** $y = 1$
- **Constraint:** $x^T\theta = 1 \implies 1\theta_1 + 0.1\theta_2 + 0.01\theta_3 = 1$
- **Loss:** $f(\theta) = (1 - x^T\theta)^2$
- **Initialization:** $\theta_0 = \mathbf{0}$

Part (a): Vanilla SGD Convergence

Analysis:
When Stochastic Gradient Descent (SGD) is initialized at zero and applied to a linear least-squares problem with more parameters than data points (underdetermined system), it converges to the minimum L_2 norm solution.
Mathematically, we are looking for the vector θ that minimizes $\|\theta\|_2^2$ subject to $x^T\theta = 1$.

Derivation:
The solution must lie in the span of the data vector x. Thus, $\theta = k \cdot x$ for some scalar k.
Substitute this into the constraint:

$$x^T(kx) = 1 \implies k(x^Tx) = 1 \implies k = \frac{1}{\|x\|_2^2}$$

So the solution is:

$$\theta_{SGD} = \frac{x}{\|x\|_2^2}$$

Calculation:

1. Calculate the squared norm of x:

   $$\|x\|_2^2 = 1^2 + 0.1^2 + 0.01^2 = 1 + 0.01 + 0.0001 = 1.0101$$

2. Scale the vector x:

$$\theta = \frac{1}{1.0101}\begin{bmatrix}1\\0.1\\0.01\end{bmatrix}$$

Answer (a):

$$\theta = \begin{bmatrix}\frac{1}{1.0101}\\\frac{0.1}{1.0101}\\\frac{0.01}{1.0101}\end{bmatrix} \approx \begin{bmatrix}0.9900\\0.0990\\0.0099\end{bmatrix}$$

Part (b): Simplified Adam Convergence

Analysis:
The simplified Adam optimizer uses a scaling matrix M_t based on the inverse of the absolute gradient values (since $n = 1$, the "root mean square" of a single gradient component is just its absolute value).

- Gradient for component i: $g_i \propto -x_i(1 - x^T\theta)$.
- Scaling factor: $M_{ii} = 1/|g_i| \propto 1/|x_i|$ (ignoring the shared error term).
- Update step: $\Delta\theta_i \propto -M_{ii}g_i \propto \frac{x_i}{|x_i|} = \text{sign}(x_i)$.

Since all components of x are positive ($1, 0.1, 0.01$), the update direction is always $[1, 1, 1]^T$. Initialized at 0, the parameters will grow equally until they satisfy the constraint.

Derivation:
$\theta = k[1, 1, 1]^T$. We solve for k using the constraint $x^T\theta = 1$:

$$1(k) + 0.1(k) + 0.01(k) = 1$$

$$k(1.11) = 1 \implies k = \frac{1}{1.11}$$

Answer (b):

$$\theta = \begin{bmatrix}\frac{1}{1.11}\\\frac{1}{1.11}\\\frac{1}{1.11}\end{bmatrix} \approx \begin{bmatrix}0.9009\\0.9009\\0.9009\end{bmatrix}$$

Part (c): Feature Rescaling + SGD

Analysis:
This approach involves pre-processing the data so that all features have the same scale (unit second-moment), solving the problem in that scaled space using SGD, and then converting back.
Step 1: Rescaling
We want each feature x_i to have unit second moment ($x_i^2 = 1$).

- Original $x = [1, 0.1, 0.01]^T$.
- Scaling factors $S = \text{diag}(1, 10, 100)$.
- Rescaled input $\tilde{x} = S x = [1, 1, 1]^T$.

Step 2: Solve with SGD in rescaled space
We find the minimum L_2 norm solution for $\tilde{x}^T\tilde{\theta} = 1$.
Since $\tilde{x} = [1, 1, 1]^T$, the symmetry implies all weights $\tilde{\theta}_i$ must be equal.

$$\tilde{\theta}_1 + \tilde{\theta}_2 + \tilde{\theta}_3 = 1 \implies \tilde{\theta}_i = \frac{1}{3}$$

So, $\tilde{\theta} = [1/3, 1/3, 1/3]^T$.
Step 3: Convert back to original units
The relationship between the predictions must remain invariant: $x^T\theta = \tilde{x}^T\tilde{\theta}$.
Substituting $\tilde{x} = Sx$:
$x^T\theta = (Sx)^T\tilde{\theta} = x^T S^T\tilde{\theta} = x^T(S\tilde{\theta})$
Thus, $\theta = S\tilde{\theta}$.
Calculations:

- $\theta_1 = 1 \cdot (1/3) = 1/3$
- $\theta_2 = 10 \cdot (1/3) = 10/3$
- $\theta_3 = 100 \cdot (1/3) = 100/3$

Answer (c):

$$\theta = \begin{bmatrix} 1/3 \\ 10/3 \\ 100/3 \end{bmatrix} \approx \begin{bmatrix} 0.333 \\ 3.333 \\ 33.333 \end{bmatrix}$$