

Introduction to Machine Learning - CS725
Instructor: Prof. Ganesh Ramakrishnan
Lecture 9 - Optimization for Regression and
Machine Learning

Recap: Foundations with Gradient Vector

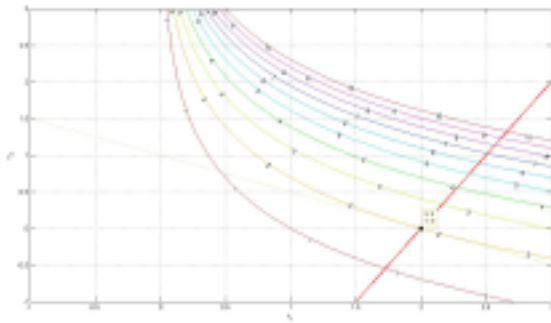
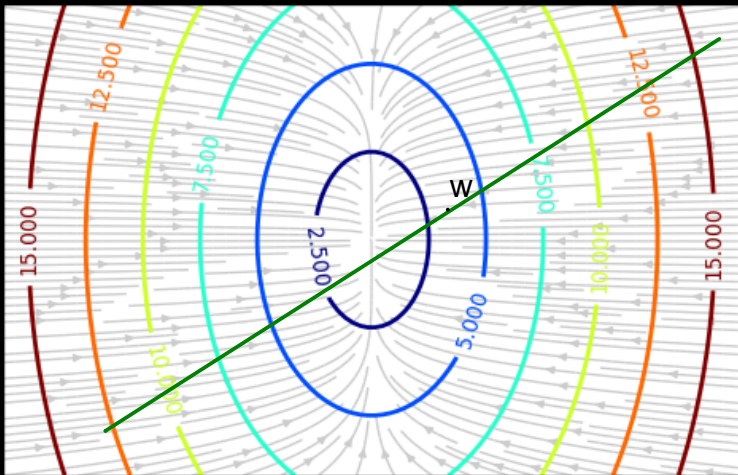


Figure 1: The level curves along with the gradient vector at $(2, 0)$. Note that the gradient vector is perpendicular to the level curve $x_1 e^{x_2} = 2$ at $(2, 0)$



Curvature along
this slice (v) at pt w is

$$v^T \nabla^2 f(w) v$$

Curvature should be
positive along every
slice v ...

That is, the Hessian
 $\nabla^2 f(w)$ should
be positive definite

Recap: Foundations with Gradient Vector

- Magnitude (euclidean norm) of gradient vector at any point indicates maximum value of directional derivative at that point
- Thus, at the point of minimum of a differentiable minimization objective (such as ridge regression),
- Ridge Regression: Find \mathbf{w} such that

$$\mathbf{w}^* = \arg \min_{\mathbf{w}} \|\Phi \mathbf{w} - \mathbf{y}\|^2 + \lambda \|\mathbf{w}\|^2 \quad (1)$$

$$= \arg \min_{\mathbf{w}} (\mathbf{w}^T \Phi^T \Phi \mathbf{w} - 2\mathbf{w}^T \Phi^T \mathbf{y} + \mathbf{y}^T \mathbf{y} + \lambda \|\mathbf{w}\|^2) \quad (2)$$

Foundations: Necessary condition 1 (Solving Ridge)

- If $\nabla f(\mathbf{w}^*)$ is defined & \mathbf{w}^* is local minimum/maximum, then $\nabla f(\mathbf{w}^*) = 0$ (A necessary condition) (Cite : Theorem 60 of [CS725/notes/classNotes/BasicsOfConvexOptimization.pdf](#))
- Given that

$$f(\mathbf{w}) = (\mathbf{w}^T \Phi^T \Phi \mathbf{w} - 2\mathbf{w}^T \Phi^T \mathbf{y} + \mathbf{y}^T \mathbf{y} + \lambda \|\mathbf{w}\|^2)$$

\implies

- We would have

Gradient must vanish at solution

.....
Trick: First work out the solution to the 1-d case
 \implies

\implies

Foundations: Necessary condition 1 (Solving Ridge)

- If $\nabla f(\mathbf{w}^*)$ is defined & \mathbf{w}^* is local minimum/maximum, then $\nabla f(\mathbf{w}^*) = 0$ (A necessary condition) (Cite : Theorem 60)

CS725/notes/classNotes/BasicsOfConvexOptimization.pdf

- Given that

$$f(\mathbf{w}) = (\mathbf{w}^T \Phi^T \Phi \mathbf{w} - 2\mathbf{w}^T \Phi^T \mathbf{y} + \mathbf{y}^T \mathbf{y} + \lambda \|\mathbf{w}\|^2) \quad (3)$$

$$\implies \nabla f(\mathbf{w}) = 2\Phi^T \Phi \mathbf{w} - 2\Phi^T \mathbf{y} + 2\lambda \mathbf{w} \quad (4)$$

- We would have

$$\nabla f(\mathbf{w}^*) = 0 \quad (5)$$

$$\implies 2(\Phi^T \Phi + \lambda I) \mathbf{w}^* - 2\Phi^T \mathbf{y} = 0 \quad (6)$$

$$\implies \underline{\mathbf{w}^* = (\Phi^T \Phi + \lambda I)^{-1} \Phi^T \mathbf{y}} \quad (7)$$

Can we insist on more for a minimum to hold?

At $x < 0$, curvature is downwards, and at $x > 0$ it is upwards.. $x = 0$ is inflection point

Eg: In a horse saddle, you sit on points on inflection..

Consider $f(x) = x^3$



At $x = 0$, $f'(x) = 3x^2 = 0$, though $x = 0$ is NOT a point of minimum!

Can we insist on more for a minimum to hold?

Gradient points in direction of increasing values of the function.

IMAGINE 1-D CAS: (Gradient) Derivative of gradient will correspond to increasing values of the rate of change

$$f(x) = x^3$$

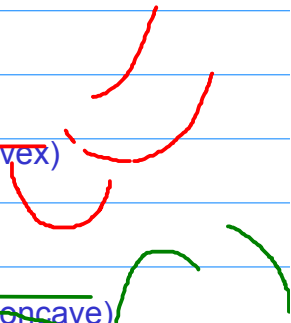
$$f'(x) = 3x < 0 \text{ if } x < 0 \text{ and } > 0 \text{ if } x > 0$$

~~positive $f''(x)$ corresponds to curvature upwards (convex)~~

At point of min, convex curvature is NECESSARY

~~negative $f''(x)$ corresponds to curvature downwards (concave)~~

At point of max, concave curvature is NECESSARY



Foundations: Necessary Condition 2 (Solving Ridge)

Positive semi-definiteness is necessary condition

- Is $\nabla^2 f(\mathbf{w}^*)$ positive definite ? (Sufficient for local minimum)
i.e. $\forall \mathbf{x} \neq 0$, is $\mathbf{x}^T \nabla^2 f(\mathbf{w}^*) \mathbf{x} > 0$? (Note : Any positive definite matrix is also positive semi-definite)
(Cite : Section 3.12 & 3.12.1)¹

Trick: Computing Hessian can be thought of as computing gradient on the transpose of the gradient vector (a row) since every time you apply gradient, you are generating a column vector.

.....
This gives you a matrix (Hessian)

¹CS725/notes/classNotes/LinearAlgebra.pdf

Foundations: Necessary Condition 2 (Solving Ridge)

- Is $\nabla^2 f(\mathbf{w}^*)$ positive definite ? (Sufficient for local minimum)
i.e. $\forall \mathbf{x} \neq 0$, is $\mathbf{x}^T \nabla^2 f(\mathbf{w}^*) \mathbf{x} > 0$? (Any p.d matrix is also p.s.d)
(Cite : Section 3.12 & 3.12.1)²

$$\nabla^2 f(\mathbf{w}^*) = 2\Phi^T \Phi + 2\lambda I \quad (8)$$

$$\implies \mathbf{x}^T \nabla^2 f(\mathbf{w}^*) \mathbf{x} = 2\mathbf{x}^T (\Phi^T \Phi + \lambda I) \mathbf{x} \quad (9)$$

$$= 2 \left((\Phi + \sqrt{\lambda} I) \mathbf{x} \right)^T \Phi \mathbf{x} \quad (10)$$

$$= 2 \left\| (\Phi + \sqrt{\lambda} I) \mathbf{x} \right\|^2 \geq 0 \quad (11)$$

Adding a lambda forces
the matrix to be most p.d
(Tutorial 3+4 problem 8)

Example of linearly correlated features

- Example where Φ doesn't have a full column rank,

$$\Phi = \begin{bmatrix} x_1 & x_1^2 & x_1^2 & x_1^3 \\ x_2 & x_2^2 & x_2^2 & x_2^3 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & x_n^2 & x_n^2 & x_n^3 \end{bmatrix} \quad (12)$$

- This is the simplest form of linear correlation of features, and it is not at all desirable.
- Effect of a nonzero λ with such Φ is that

Example of linearly correlated features

- Example where Φ doesn't have a full column rank,

$$\Phi = \begin{bmatrix} x_1 & x_1^2 & x_1^2 & x_1^3 \\ x_2 & x_2^2 & x_2^2 & x_2^3 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & x_n^2 & x_n^2 & x_n^3 \end{bmatrix} \quad (12)$$

- This is the simplest form of linear correlation of features, and it is not at all desirable.
- Effect of a nonzero λ with such Φ is that it tends to make the Hessian more positive definite (prob 8 of Tut 3+4)

Do Closed-form solutions Always Exist?

- Linear regression and Ridge regression both have closed-form solutions

- For linear regression,

$$w^* = (\Phi^T \Phi)^{-1} \Phi^T y$$

- For ridge regression,

$$w^* = (\Phi^T \Phi + \lambda I)^{-1} \Phi^T y$$

(for linear regression, $\lambda = 0$)

- What about optimizing the formulations (constrained/penalized) of Lasso (L_1 norm)? And support-based penalty (L_0 norm)? **Also requires tools of Optimization/duality**

Gradient Descent Algorithm

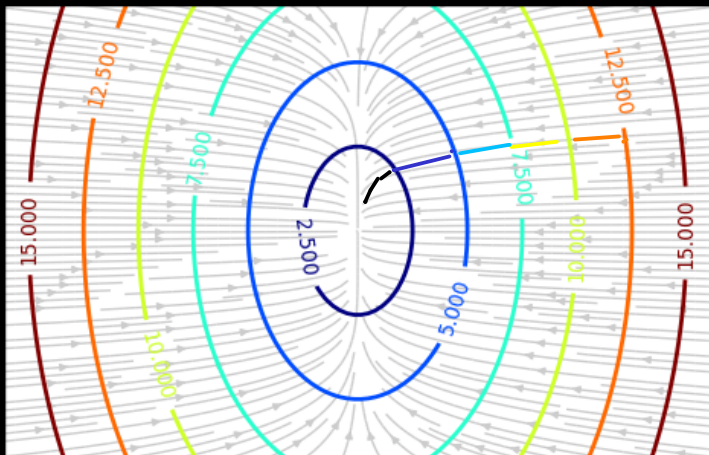
Find starting point $\mathbf{w}^{(0)} \in \mathcal{D}$

- $\Delta \mathbf{w}^k = -\nabla_{\varepsilon}(\mathbf{w}^{(k)})$ **Compute the gradient direction at current level**
- Choose a step size $t^{(k)} > 0$ using exact (or backtracking³) ray search. **Determined how much to move along this step..**
- Obtain $\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \underline{t}^{(k)} \underline{\Delta \mathbf{w}^{(k)}}$.
- Set $k = k + 1$. **until** stopping criterion (such as $\|\nabla_{\varepsilon}(\mathbf{w}^{(k+1)})\| \leq \epsilon$) is satisfied

Exact line search: $t(k) = \operatorname{argmin} \|\nabla_{\varepsilon}(\mathbf{w}^{(k)} + t \Delta \mathbf{w}^{(k)})\|$

~~It is a one dimensional optimization problem, often easy to solve~~

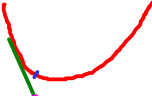
³optional



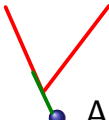
1) Direction of decrease or increase (one is negative of other) changes across level curves and (2) as rate of change changes, we have to revise the amount of movement/descent at that point (step length)

(Optional) Subgradients

- An equivalent condition for convexity of $f(\mathbf{x})$:


$$\forall \mathbf{x}, \mathbf{y} \in \text{dmn}(\mathbf{f}), \quad \underline{\mathbf{f}(\mathbf{y})} \geq \underline{\mathbf{f}(\mathbf{x}) + \nabla^\top \mathbf{f}(\mathbf{x})(\mathbf{y} - \mathbf{x})}$$

- $\mathbf{g}_f(\mathbf{x})$ is a *subgradient* for a function f at \mathbf{x} if


$$\forall \mathbf{y} \in \text{dmn}(\mathbf{f}), \quad \mathbf{f}(\mathbf{y}) \geq \mathbf{f}(\mathbf{x}) + \underline{\mathbf{g}_f(\mathbf{x})}^\top (\mathbf{y} - \mathbf{x})$$

- Any convex (even non-differentiable) function will have a subgradient at any point in the domain!
- If a convex function f is differentiable at \mathbf{x} then $\nabla f(\mathbf{x}) = \mathbf{g}_f(\mathbf{x})$
- \mathbf{x} is a point of minimum of (convex) f if and only if $\mathbf{0}$ is a subgradient of f at \mathbf{x}

(Sub)Gradient Descent Algorithm

For Lasso, Subgradient descent is possible, though we will show another method..

Find starting point $\mathbf{w}^{(0)} \in \mathcal{D}$

- $\Delta \mathbf{w}^{(k)} = -\nabla \varepsilon(\mathbf{w}^{(k)})$
- Choose a step size $t^{(k)} > 0$ using exact or backtracking ray search.
- Obtain $\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + t^{(k)} \underline{\Delta \mathbf{w}^{(k)}}$.
- Set $k = k + 1$. **until** stopping criterion (such as $\|\nabla \varepsilon(\mathbf{w}^{(k+1)})\| \leq \epsilon$) is satisfied

(Sub)Gradient Descent Algorithm

Exact line search algorithm to find $t^{(k)}$

- The line search approach first finds a descent direction along which the objective function f will be reduced and then computes a step size that determines how far \mathbf{x} should move along that direction.
- In general,

$$\underline{t^{(k)} = \arg \min_t f(\mathbf{w}^{(k+1)})} \quad (13)$$

- Thus,

(Sub)Gradient Descent Algorithm

Exact line search algorithm to find $t^{(k)}$

- The line search approach first finds a descent direction along which the objective function f will be reduced and then computes a step size that determines how far \mathbf{x} should move along that direction.
- In general,

$$t^{(k)} = \arg \min_t f \left(\mathbf{w}^{(k+1)} \right) \quad (13)$$

- Thus, for L_2 regularized least squared regression

$$t^{(k)} = \arg \min_t \epsilon \left(\mathbf{w}^{(k)} + 2t \left(\Phi^T \mathbf{y} - \Phi^T \phi \mathbf{w}^{(k)} - \lambda \mathbf{w}^{(k)} \right) \right) \quad (14)$$

Illustration of (Sub)Gradient Descent Algorithm

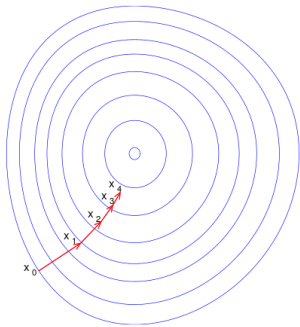


Figure 2: A red arrow originating at a point shows the direction of the negative gradient at that point. Note that the (negative) gradient at a point is orthogonal to the level curve going through that point. We see that gradient descent leads us to the bottom of the bowl, that is, to the point where the value of the function F is minimal. Source: Wikipedia


Gradient Descent and LS Regression (Tutorial 3+4)

Consider solving the (L_2 regularized) Least Squares Linear Regression problem using the gradient descent algorithm. And let us say $w^{(0)} = 0$ and that the step length $t^{(k)}$ is computed using exact line search for each value of k . In how many steps will the gradient descent algorithm converge? What would be your answer if we had a different initialization for $w^{(0)}$

Subgradients and Lasso

$$\mathbf{w}_{Lasso} = \arg \min_{\mathbf{w}} \|\Phi \mathbf{w} - \mathbf{y}\|_2^2 + \|\mathbf{w}\|_1$$

- The unconstrained form for Lasso has no closed form solution
- But it can be solved using a generalization of gradient descent called proximal subgradient descent⁴

⁴<https://www.cse.iitb.ac.in/~cs725/notes/classNotes/lassoElaboration.pdf> 

Iterative Soft Thresholding Algorithm for Solving Lasso

Proximal Subgradient Descent for Lasso

- Let $\varepsilon(\mathbf{w}) = \|\phi\mathbf{w} - \mathbf{y}\|_2^2$
- **Proximal Subgradient Descent Algorithm:**
Initialization: Find starting point $\mathbf{w}^{(0)}$ My step without the 1-norm
 - Let $\hat{\mathbf{w}}^{(k+1)}$ be a next gradient descent iterate for $\varepsilon(\mathbf{w}^k)$
 - Compute $\mathbf{w}^{(k+1)} = \underset{\mathbf{w}}{\operatorname{argmin}} \|\mathbf{w} - \hat{\mathbf{w}}^{(k+1)}\|_2^2 + \lambda \mathbf{t} \|\mathbf{w}\|_1$ by setting course correction subgradient of this objective to $\mathbf{0}$. This results in (see <https://www.cse.iitb.ac.in/~cs725/notes/classNotes/lassoElaboration.pdf>)
 - 1 ...
 - 2 ...
 - 3 ...Hint: First solve in one dimension...
- Set $k = k + 1$, **until** stopping criterion is satisfied (such as no significant changes in \mathbf{w}^k w.r.t $\mathbf{w}^{(k-1)}$)

Iterative Soft Thresholding Algorithm (Proximal Subgradient Descent) for Lasso

- Let $\varepsilon(\mathbf{w}) = \|\phi\mathbf{w} - \mathbf{y}\|_2^2$
- **Iterative Soft Thresholding Algorithm:**
Initialization: Find starting point $\mathbf{w}^{(0)}$
 - Let $\hat{\mathbf{w}}^{(k+1)}$ be a next iterate for $\varepsilon(\mathbf{w}^k)$ computed using any (gradient) descent algorithm
 - Compute $\mathbf{w}^{(k+1)} = \underset{\mathbf{w}}{\operatorname{argmin}} \|\mathbf{w} - \hat{\mathbf{w}}^{(k+1)}\|_2^2 + \lambda t \|\mathbf{w}\|_1$ by:
 - 1 If $\hat{w}_i^{(k+1)} > \lambda t/2$, then $w_i^{(k+1)} = -\lambda t/2 + \hat{w}_i^{(k+1)}$
 - 2 If $\hat{w}_i^{(k+1)} < -\lambda t/2$, then $w_i^{(k+1)} = \lambda t/2 + \hat{w}_i^{(k+1)}$
 - 3 0 otherwise.
- Set $k = k + 1$, **until** stopping criterion is satisfied (such as no significant changes in \mathbf{w}^k w.r.t $\mathbf{w}^{(k-1)}$)