COMPSCI 689
Lecture 8: Learning for SVMs

Benjamin M. Marlin

College of Information and Computer Sciences
University of Massachusetts Amherst

Slides by Benjamin M. Marlin (marlin@cs.umass.edu).

Support Vector Regression

$$f_\theta(\mathbf{x}) = \mathbf{x}\theta$$

$$\hat{\theta} = \arg\min_\theta C \sum_{n=1}^{N} L_\epsilon(y_n, f_\theta(\mathbf{x}_n)) + ||\mathbf{w}||_2^2$$

$$L_\epsilon(y, y') = \begin{cases} 0 & \text{... if } |y - y'| < \epsilon \\ |y - y'| - \epsilon & \text{... otherwise} \end{cases}$$

Support Vector Classification

$$f_\theta(\mathbf{x}) = \text{sign}(g_\theta(\mathbf{x}))$$

$$g_\theta(\mathbf{x}) = \mathbf{x}\theta$$

$$\hat{\theta} = \arg\min_\theta C \sum_{n=1}^{N} \max(0, 1 - y_n g_\theta(\mathbf{x}_n)) + ||\mathbf{w}||_2^2$$

Margin

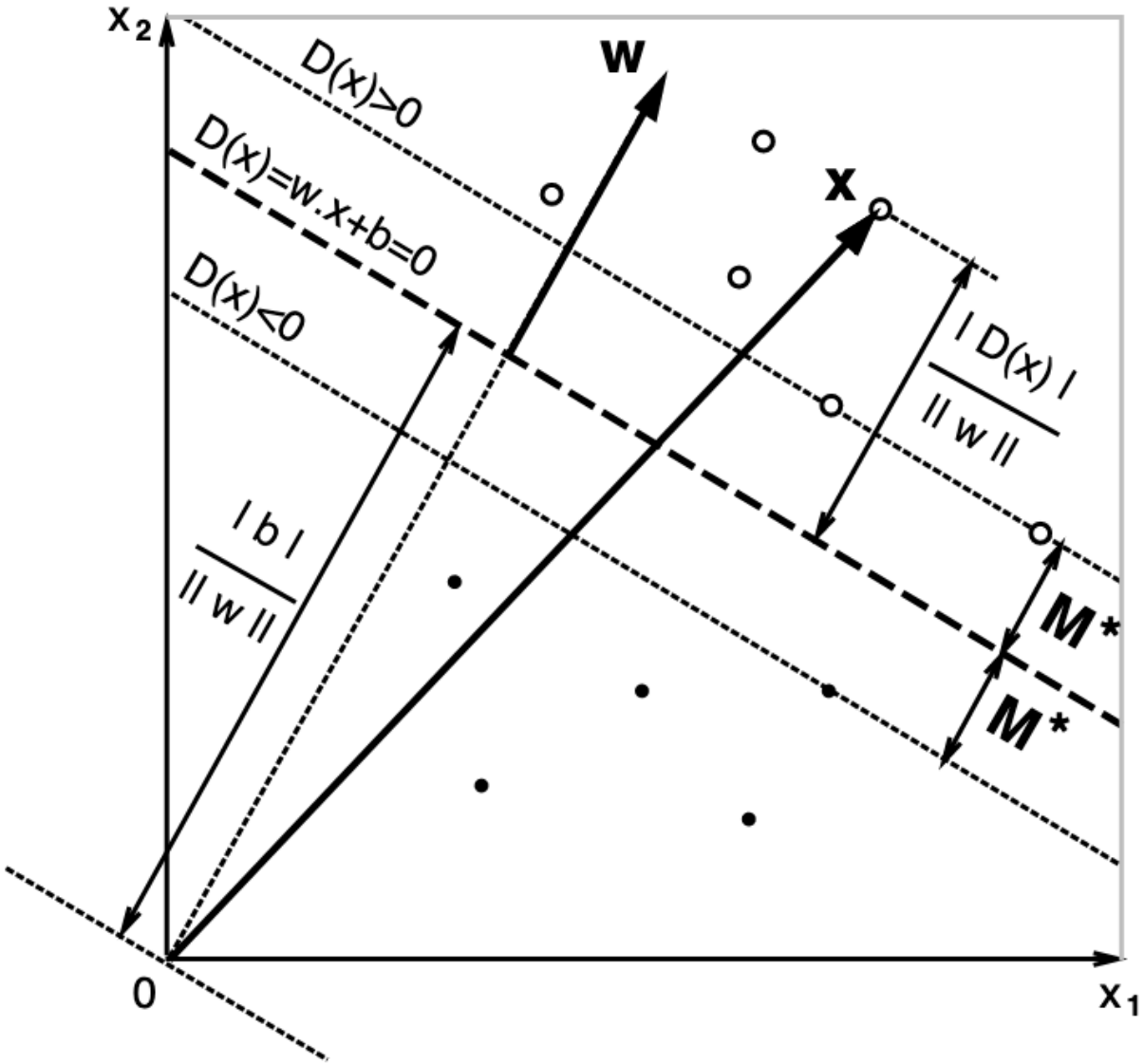
- Suppose we have a data point $\mathbf{x}_n$. The signed distance from the data point to the separating hyperplane $\mathcal{H} = \{\mathbf{x}|\mathbf{x}\theta = 0\}$ is given by:

$$D_n(\theta) = \frac{y_n g_\theta(\mathbf{x}_n)}{\|\mathbf{w}\|_2}$$

- The value of the margin is then the minimum over the data set of the signed distances of the data points to the separating hyper-plane:

$$M(\theta) = \min_n D_n(\theta)$$

Margin Example



Margin Maximization

The original SVC optimization problem was to maximize the value of the margin as a function of the model parameters. This gives us:

$$\hat{\theta} = \arg\max_{\theta} \min_{n} D_n(\theta)$$

$$\hat{\theta} = \arg\max_{\theta} \min_{n} \frac{y_n g_\theta(\mathbf{x}_n)}{\|\mathbf{w}\|_2}$$

Quadratic Program Formulation

The direct margin maximization problem is not easy to solve, however, we can re-formulate it into a quadratic program for which there are existing solvers.

$$\hat{\theta} = \arg\min_{\theta, \epsilon_{1:N}} \|\mathbf{w}\|_2^2 + C \sum_{n=1}^{N} \epsilon_n$$
$$\text{s.t. } \forall n \; y_n g_\theta(\mathbf{x}_n) \geq 1 - \epsilon_n$$
$$\forall n \; \epsilon_n \geq 0$$

This version of the problem can be further manipulated into the hinge loss formulation.

Dual Quadratic Program Formulation

The constrained quadratic program can also be re-written in a form where the parameters are weights applied to the data cases. This is called the SVM dual formulation.

$$\arg\max_{\boldsymbol{\alpha}} \sum_{n=1}^{N} \alpha_n - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} y_n y_m \alpha_n \alpha_m \mathbf{K}_{nm}$$

$$\text{s.t. } \forall n \;\; 0 \leq \alpha_n \leq C$$

$$\sum_{n=1}^{N} \alpha_n y_n = 0$$

The matrix $\mathbf{K}$ that appears in the objective contains the inner products between all pairs of training data vectors: $\mathbf{K}_{nm} = \mathbf{x}_n \mathbf{x}_m^T$. The matrix $\mathbf{K}$ is thus positive semi-definite and the optimization problem is maximizing a concave function.

Dual Prediction

Given the value of α, we make predictions using the dual formulation as follows:

$$f_{svm}(\mathbf{x}) = \text{sign}(\sum_{n=1}^{N} y_n \alpha_n \mathbf{x}_n \mathbf{x}^T + b)$$

Note that this requires having access to the training data at prediction time. Also note that both the learning problem and the prediction problem are both based on computing inner products between data cases.

Basis Expansion and Kernels

We can apply basis expansion to any formulation of the learning problem, but it's particularly interesting in the dual formulation:

$$f_{svm}(\mathbf{x}) = \text{sign}(\sum_{n=1}^{N} y_n \alpha_n \phi(\mathbf{x}_n)\phi(\mathbf{x})^T + b)$$
$$= \text{sign}(\sum_{n=1}^{N} y_n \alpha_n \mathcal{K}(\mathbf{x}_n, \mathbf{x}) + b)$$

The function $\mathcal{K}$ is referred to as a *kernel function* and must satisfy a property known as *Mercer*'s condition.

Example: Kernels

- Linear Kernel: $\mathcal{K}(\mathbf{x}', \mathbf{x}) = \mathbf{x}'\mathbf{x}^T$
- Polynomial Kernel: $\mathcal{K}(\mathbf{x}', \mathbf{x}) = (1 + \mathbf{x}'\mathbf{x}^T)^B$
- Exponential (RBF) Kernel: $\mathcal{K}(\mathbf{x}', \mathbf{x}) = \exp\left(-\beta\|\mathbf{x}' - \mathbf{x}\|_2^2\right)$
- ... and many more kernel functions, including for inputs that are not real-valued.

Back to the Hinge Loss Formulation

$$f_\theta(\mathbf{x}) = \text{sign}(g_\theta(\mathbf{x}))$$

$$g_\theta(\mathbf{x}) = \mathbf{x}\theta = \mathbf{x}\mathbf{w} + b$$

$$\hat{\theta} = \arg\min_\theta C \sum_{n=1}^{N} \max(0, 1 - y_n g_\theta(\mathbf{x}_n)) + ||\mathbf{w}||_2^2$$

Global Optimality of Convex Non-Differentiable Functions

- Our existing optimization theory only holds for the case of differentiable functions.
- However, it turns out that many results generalize to the case of non-differentiable functions that are convex and we can use them to directly minimize the hinge loss.
- To begin, strongly convex non-differentiable functions have a unique global minima, exactly as with convex differentiable functions.
- We will begin with the characterization of the minimizer of a non-differentiable convex function. [1]

[1] This section mostly follows the presentation in Boyd's EE364b - Convex Optimization II course.

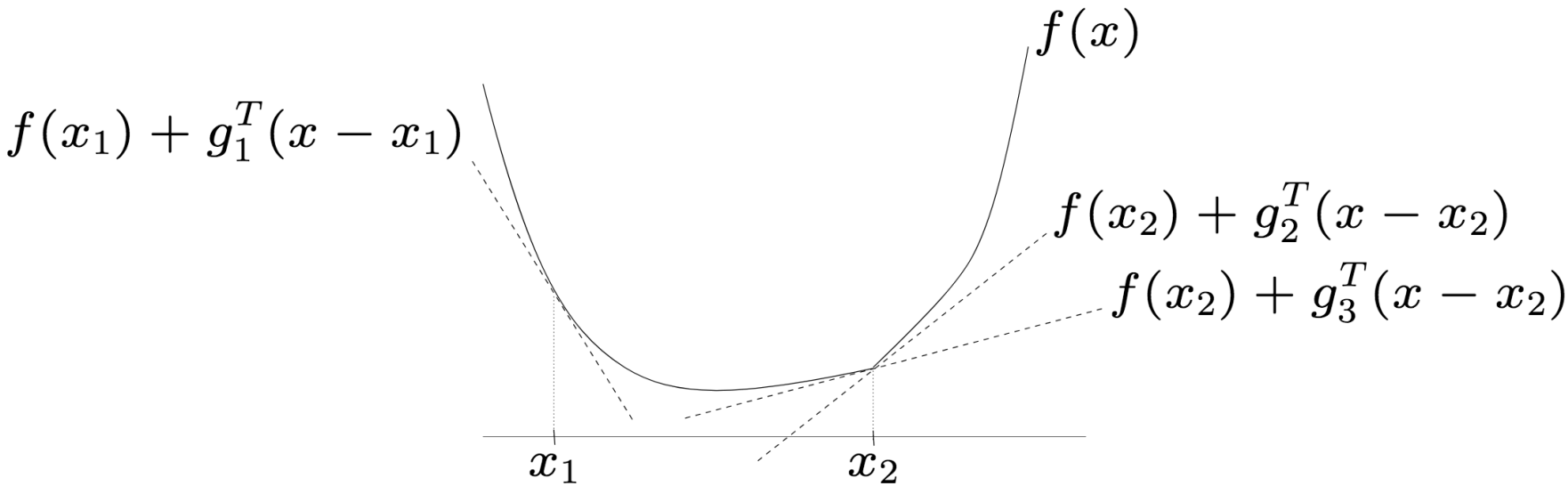
Subgradient

- Let $f : \mathbb{R}^D \to \mathbb{R}$. A vector $\mathbf{g} \in \mathbb{R}^D$ is said to be a sub-gradient of f at a point $\mathbf{x}_o \in \mathbb{R}^D$ if for all $\mathbf{x} \in \mathbb{R}^D$:

$$f(\mathbf{x}) \geq f(\mathbf{x}_o) + \mathbf{g}^T \cdot (\mathbf{x} - \mathbf{x}_o)$$

- That is to say, the hyperplane defined by $h(\mathbf{x}) = f(\mathbf{x}_o) + \mathbf{g}^T \cdot (\mathbf{x} - \mathbf{x}_o)$ lies at or below below f everywhere and touches f at $\mathbf{x}_o$.

Example: Subgradients

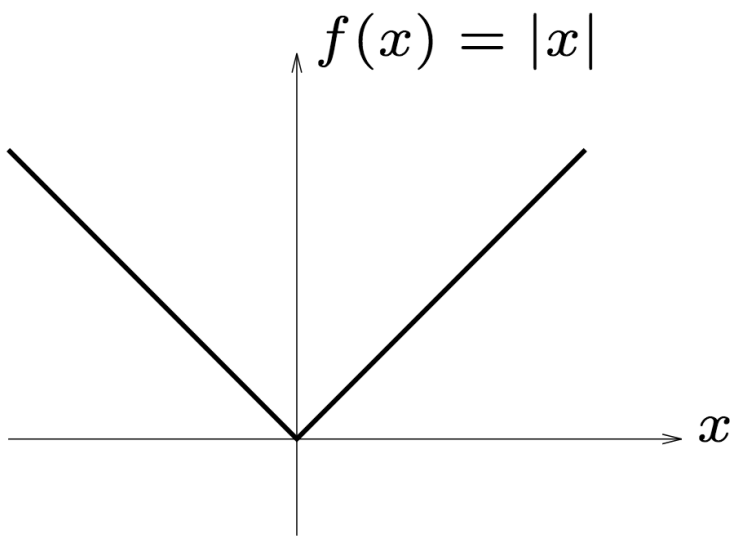


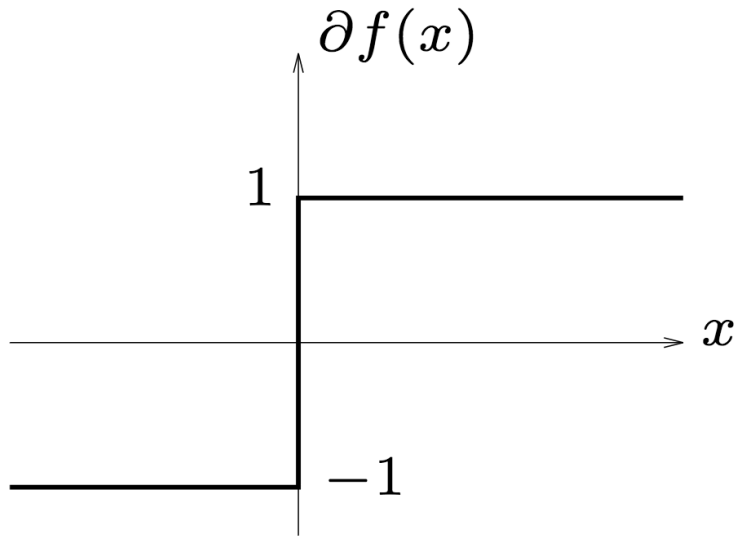
In this example, $\mathbf{g}_1$ is the unique subgradient of f at $\mathbf{x}_1$. Due to f being non-differentiable at $\mathbf{x}_2$, both $\mathbf{g}_2$ and $\mathbf{g}_3$ are subgradients of f at $\mathbf{x}_2$.

Subdifferentials

- If f is convex and is differentiable at $\mathbf{x}_o$, then $\nabla f(\mathbf{x}_o)$ is its unique subgradient at $\mathbf{x}_o$.
- If f is convex and non-differentiable at $\mathbf{x}_o$, it will generally have more that one vector $\mathbf{g}$ satisfying the subgradient property.
- The set of all subgradients of f at $\mathbf{x}_o$ is called the subdifferential of f at $\mathbf{x}_o$ denoted by $\partial f(\mathbf{x}_o)$.
- $\partial f(\mathbf{x}_o)$ is a closed, convex set in $\mathbb{R}^D$. If f is convex, $\partial f(\mathbf{x}_o)$ is always non-empty.

Example: Subdifferentials





The righthand plot shows $\partial f(x)$ for $f(x) = |x|$.
We have $\partial f(\mathbf{x}) = \{\text{sign}(x)\}$ for $x \neq 0$.
When $x = 0$, the line $|0| + g \cdot (x - 0) = g \cdot x$ will lie below f everywhere only if $g \in [-1, 1]$.

Characterizing the Global Minimum

- Let $f : \mathbb{R}^D \to \mathbb{R}$ be a convex function.
- $\mathbf{x}_*$ is the global minimizer of f if and only if $\mathbf{0} \in \partial f(\mathbf{x}_*)$.
- This is a generalization of the idea of a stationary point to include the case of non-differentiable functions.

Finding Subdifferentials

- Suppose that x_0 is a point of non-differentiability for a 1-dimensional convex function $f(x)$.
- Suppose that in the neighborhood $[a, x_0]$, for some $a < x_0$, the value of $f(x)$ is given by a differentiable function $g(x)$ and in the neighborhood $[x_0, b]$ for some $b > x_0$, the value of $f(x)$ is given by a differentiable function $h(x)$.
- Then, the subdifferential of $f(x)$ at x_0 is:

$$\partial f(x_0) = \left[\frac{dg(x)}{dx}|_{x_0}, \frac{dh(x)}{dx}|_{x_0} \right]$$

Partial Linearity and Chain Rule

The subdifferential operator satisfies partial linearity and chain rule properties:

- **Scaling:** If $f(\mathbf{x})$ is a convex function and $\alpha > 0$, then if $\mathbf{g} \in \partial f(\mathbf{x})$, $\alpha\mathbf{g} \in \partial(\alpha f(\mathbf{x}))$
- **Addition:** If $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ are convex functions and $\mathbf{g}_1 \in \partial f_1(\mathbf{x})$ and $\mathbf{g}_2 \in \partial f_2(\mathbf{x})$, then $\mathbf{g}_1 + \mathbf{g}_2 \in \partial(f_1(\mathbf{x}) + f_2(\mathbf{x}))$
- **Chain Rule**: If $f : \mathbb{R} \to \mathbb{R}$ is a convex function and $h : \mathbb{R}^D \to \mathbb{R}$ is a linear function $h(\mathbf{x}) = \mathbf{x}\mathbf{a} + b$, then the sub-differential of $f(h(\mathbf{x}))$ is $\{g\mathbf{a}^T | g \in \partial f(y), y = h(\mathbf{x})\}$.

These properties can be used to determine the subdifferentials of more some complex functions if they reduce to certain combinations or compositions of simpler functions.

Sub-gradient Descent

- If f is differentiable at $\mathbf{x}$ then $\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}$ and $-\mathbf{g} = -\nabla f(\mathbf{x})$ is a descent direction unless $\nabla f(\mathbf{x}) = 0$.
- If f is not differentiable at $\mathbf{x}$, and $\mathbf{0} \notin \partial f(\mathbf{x})$, then there exists at least one $\mathbf{g} \in \partial f(\mathbf{x})$ where $-\mathbf{g}$ is a descent direction.
- If f is not differentiable at $\mathbf{x}$, then there may exist some $\mathbf{g} \in \partial f(\mathbf{x})$ where $-\mathbf{g}$ is a **not** descent direction.

Sub-gradient Descent

Despite the fact that not all sub-gradients yield descent directions, it is still possible to minimize a convex non-differentiable function using a fairly basic sub-gradient descent procedure:

- Initialize $\mathbf{x}_0 \in \mathbb{R}^D, f_{min} = \infty, \mathbf{x}_* = 0$
- For k from 1 to K:
  - Let $\mathbf{g}_k \in \partial f(\mathbf{x}_{k-1})$
  - Set $\mathbf{x}_k = \mathbf{x}_{k-1} - \alpha_k \mathbf{g}_k$
  - If $f(\mathbf{x}_k) < f_{min}$ then set $f_{min} = f(\mathbf{x}_k)$ and $\mathbf{x}_* = \mathbf{x}_k$
- Return $\mathbf{x}_*$

Questions: How to choose α_k? How to choose K?

Subgradient Descent Convergence

- Line search is typically not used for sub-gradient descent procedures. It is more common to use a fixed sequence of step sizes.
- Common step size rules include $\alpha_k = \alpha/(\beta + k)$ or $\alpha_k = \alpha/\sqrt{k}$.
- Under either of these step size rules, we have that the sequence of subgradient descent iterates $\mathbf{x}_k$ satisfies:

$$\lim_{k \to \infty} f(\mathbf{x}_k) = \min_{\mathbf{x}} f(\mathbf{x})$$

Subgradient Descent with Momentum

- Initialize $\mathbf{x}_0 \in \mathbb{R}^D, f_{min} = \infty, \mathbf{x}_* = 0$
- For k from 1 to K:
  - Let $\mathbf{g}_k \in \partial f(\mathbf{x}_{k-1})$
  - Let $\mathbf{d}_k = \gamma \mathbf{d}_{k-1} + \alpha_k \mathbf{g}_k$
  - Set $\mathbf{x}_k = \mathbf{x}_{k-1} - \mathbf{d}_k$
  - If $f(\mathbf{x}_k) < f_{min}$ then set $f_{min} = f(\mathbf{x}_k)$ and $\mathbf{x}_* = \mathbf{x}_k$
- Return $\mathbf{x}_*$

Typically use with $0 < \gamma < 1$. $\gamma = 0.9$ is a common choice.

Nesterov Accelerated Subgradient Descent

- Initialize $\mathbf{x}_0 \in \mathbb{R}^D, f_{min} = \infty, \mathbf{x}_* = 0$
- For k from 1 to K:
  - Let $\mathbf{g}_k \in \partial f(\mathbf{x}_{k-1} - \gamma \mathbf{d}_{k-1})$
  - Let $\mathbf{d}_k = \gamma \mathbf{d}_{k-1} + \alpha_k \mathbf{g}_k$
  - Set $\mathbf{x}_k = \mathbf{x}_{k-1} - \mathbf{d}_k$
  - If $f(\mathbf{x}_k) < f_{min}$ then set $f_{min} = f(\mathbf{x}_k)$ and $\mathbf{x}_* = \mathbf{x}_k$
- Return $\mathbf{x}_*$

Typically use with $0 < \gamma < 1$. $\gamma = 0.9$ is a common choice.

Example: Finding a Subgradient for SVC

- We begin the risk function:

$$R(\theta, \mathcal{D}) = C \sum_{n=1}^{N} \max(0, 1 - y_n g_\theta(\mathbf{x}_n)) + ||\mathbf{w}||_2^2$$
$$= C \sum_{n=1}^{N} L(y_n g_\theta(\mathbf{x}_n)) + ||\mathbf{w}||_2^2$$

- We are looking for a vector $\mathbf{h}$ such that:

$$\mathbf{h} \in \partial R(\theta, \mathcal{D})$$

Example: Finding a Subgradient for SVC

- By the additivity and scaling properties, we have that if $\mathbf{h}_n \in \partial L(y_n g_\theta(\mathbf{x}_n))$ and $\mathbf{h}_w \in \partial ||\mathbf{w}||_2^2$, then:

$$C \sum_{n=1}^{N} \mathbf{h}_n + \mathbf{h}_w \in \partial R(\theta, \mathcal{D})$$

- We will proceed by finding suitable vectors $\mathbf{h}_n$ and $\mathbf{h}_w$ using properties of sub-gradients.

Example: Finding a Subgradient for SVC

- Consider $h_w \in \partial ||\mathbf{w}||_2^2$.
- Since $||\mathbf{w}||_2^2$ is a differentiable function with gradient $2\mathbf{w}$, we have that $h_w = 2[\mathbf{w}; 0]$.

Example: Finding a Subgradient for SVC

- Consider $\mathbf{h}_n \in \partial L(y_n g_\theta(\mathbf{x}_n))$. Recall $L(z) = \max(0, 1 - z)$ and $g_\theta(\mathbf{x}_n) = \mathbf{x}_n\theta$.
- By the chain rule, we have that $\mathbf{h}_n \in \partial L(y_n \mathbf{x}_n \theta)$ if and only if:

$$\mathbf{h}_n \in \{k y_n \mathbf{x}_n^T \mid k \in \partial L(z), z = y_n \mathbf{x}_n \theta\}$$

- We next need to figure out what $\partial L(z)$ is.

Example: Finding a Subgradient for SVC

- Consider $\partial L(z) = \partial \max(0, 1 - z)$. This is a non-differentiable convex function with one point of non-differentiability at $z = 1$.
- For $z > 1$ the functions is constant so $\partial L(z) = \{0\}$ for $z > 1$
- For $z < 1$ the functions is $1 - z$ so $\partial L(z) = \{-1\}$ for $z < 1$.
- At $z = 0$ we have $\partial L(z) = [-1, 0]$ following subdifferential rules for 1-D convex functions.
- We choose the following sub-gradient function for the hinge loss:

$$L'(z) = \begin{cases} 0 & \dots z \geq 1 \\ -1 & \dots z < 1 \end{cases}$$

Example: Finding a Subgradient for SVC

- Consider again, $\mathbf{h}_n \in \partial L(y_n \mathbf{x}_n \theta)$ if and only if:

$$\mathbf{h}_n \in \{k y_n \mathbf{x}_n^T \mid k \in \partial L(z), z = y_n \mathbf{x}_n \theta\}$$

- We have shown that $L'(z) \in \partial L(z)$, so we have that:

$$\mathbf{h}_n = L'(y_n \mathbf{x}_n \theta) y_n \mathbf{x}_n^T \in \partial L(y_n \mathbf{x}_n \theta)$$

- This gives us a final answer for a subgradient of the risk:

$$\mathbf{h} = C \sum_{n=1^N} L'(y_n \mathbf{x}_n \theta) y_n \mathbf{x}_n^T + 2[\mathbf{w}; 0]$$