Machine Learning

4771

Instructors:
Adrian Weller and Ilia Vovsha

Lecture 12: Large Margin & Optimal Hyperplane

- Structural Risk Minimization (SRM)
- Large Margin, Optimal Hyperplane (Burges Tutorial)
- Optimization
- Support Vector Machines (Bishop 7.1, Burges Tutorial)

Constructive Bound

- With probability (1-eta), for the function that minimizes empirical risk, the inequality below holds true

$$R(\alpha_\ell) < R_{emp}(\alpha_\ell) + \frac{\mathrm{E}(\ell)}{2}\left(1+\sqrt{1+\frac{4R_{emp}(\alpha_\ell)}{\mathrm{E}(\ell)}}\right)$$

where

$$\mathrm{E}(\ell) = 4\,\frac{h\left(1+\ln(2\ell/h)\right)-\ln(\eta/4)}{\ell}$$

Large Sample Size

- Suppose we have a *large sample size* (ℓ/h is large)
  - ➢ The value of actual risk is determined by value of empirical risk
  - ➢ The principle of ERM gives good results in practice
- Justification (we drop constants and show what the bound is proportional to):

$$\mathrm{E}(\ell) = 4\frac{h\left(1+\ln\left(2\ell/h\right)\right)-\ln\left(\eta/4\right)}{\ell} \approx \frac{h}{\ell} + \frac{\ln\left(2\ell/h\right)}{\left(\ell/h\right)} \approx \delta$$

$$R_{emp}\left(\alpha_\ell\right) + \frac{\mathrm{E}(\ell)}{2}\left(1+\sqrt{1+\frac{4R_{emp}\left(\alpha_\ell\right)}{\mathrm{E}(\ell)}}\right) \approx R_{emp}\left(\alpha_\ell\right) + \delta\left(1+\sqrt{1+\frac{R_{emp}\left(\alpha_\ell\right)}{\delta}}\right)$$

$$\approx R_{emp}\left(\alpha_\ell\right) + \delta\left(\sqrt{\frac{R_{emp}\left(\alpha_\ell\right)}{\delta}}\right) \approx R_{emp}\left(\alpha_\ell\right) + \sqrt{\delta R_{emp}\left(\alpha_\ell\right)}$$

Large Sample Size

- Suppose we have a large sample size (ℓ/h is large)
  - The value of actual risk is determined by value of empirical risk
  - The principle of ERM gives good results
- Justification (we drop constants and show what the bound is proportional to):

$$R(\alpha_\ell) <\approx \left\{ R_{emp}(\alpha_\ell) + \sqrt{\delta R_{emp}(\alpha_\ell)} \right\}$$

Small Sample Size

- Suppose we have a *small sample size* ($\ell/h < 20$)
  - Small empirical risk doesn't guarantee small actual risk anymore
  - Need to minimize bound over both terms simultaneously
  - To do this, we make the VC dimension (capacity) a *controlling variable*
- This observation motivates a new *induction principle*: Structural Risk Minimization
- What do we mean by a controlling variable?
- How do we justify this new induction principle?

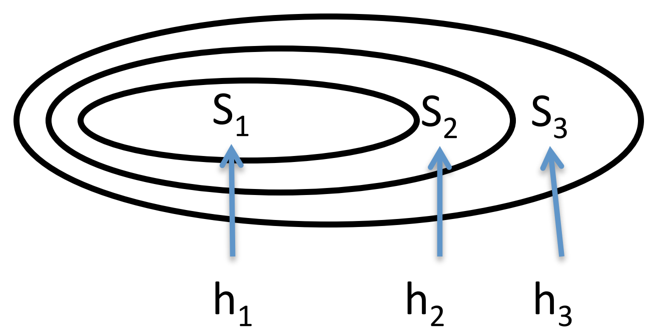
SRM Principle (idea)

- Instead of minimizing empirical risk at any cost, search for the optimal relationship between:
  1. Amount of empirical data
  2. Quality of approximation by the function chosen from a given set of functions
  3. Value that characterizes the capacity of a set of functions
- Lets impose a *structure (S*)* on the set of loss functions
- We assume that any element S_k of the structure S* has a finite VC dimension h_k
- The sequence $\{h_k\}$ for elements $\{S_k\}$ of S* is non-decreasing (as k is increased)

$$S_1 \subset S_2 \subset \cdots \subset S_n \subset \cdots$$

$$S^* = \bigcup_k S_k, \quad S_k = \{L(z,\alpha) : \alpha \in \Lambda_k\}$$

$$h_1 \le h_2 \le \cdots \le h_n \le \cdots$$



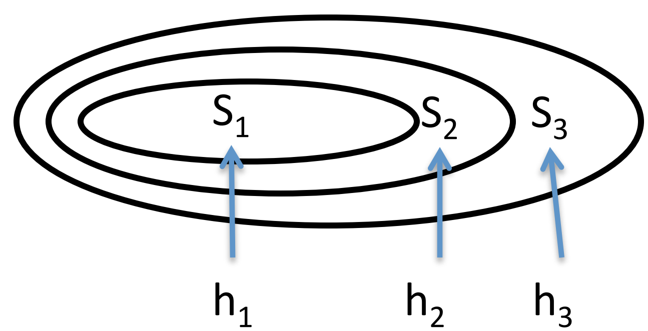
SRM Principle (idea)

- For a given sample, the SRM principle chooses the element S_k of the structure for which the smallest bound on the risk (the smallest guaranteed risk) is achieved
- Within the element S_k, we choose the function that minimizes empirical risk
- General model of capacity control
- We need to provide an *admissible structure* (which satisfies conditions) and then choose the function that yields the best guaranteed risk
- Support Vector Machine (SVM) does just that

$$S_1 \subset S_2 \subset \cdots \subset S_n \subset \cdots$$

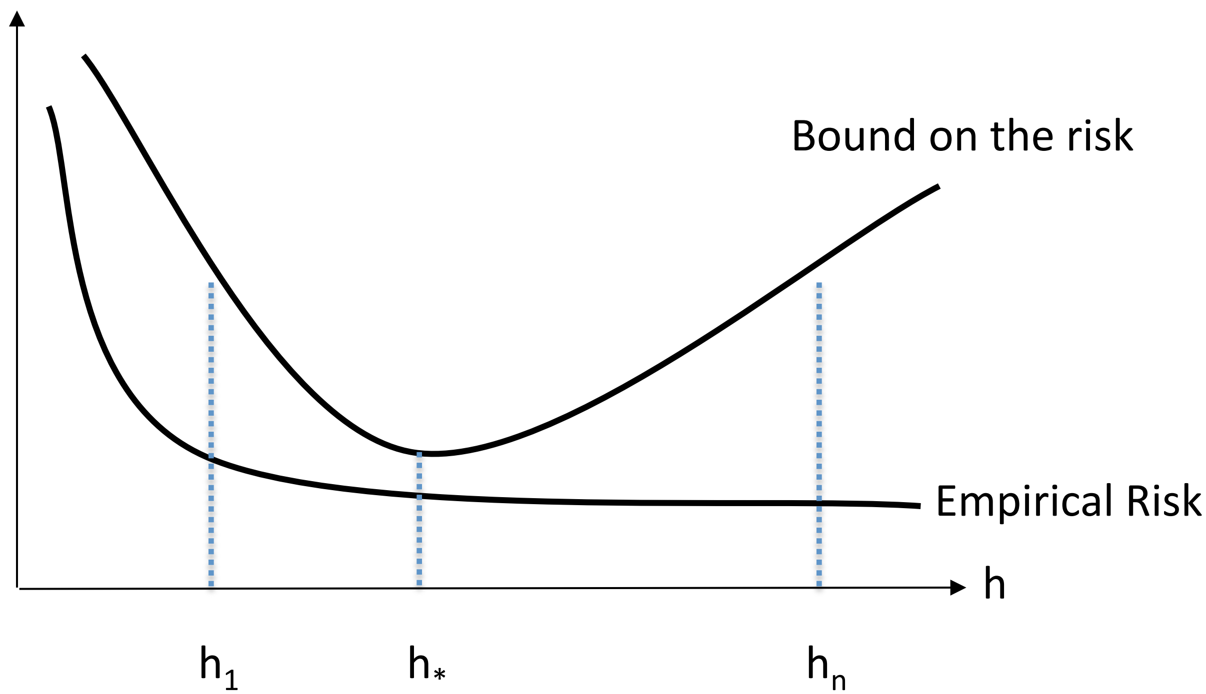
$$S^* = \bigcup_k S_k, \quad S_k = \{L(z,\alpha) : \alpha \in \Lambda_k\}$$

$$h_1 \leq h_2 \leq \cdots \leq h_n \leq \cdots$$



SRM Principle (idea)

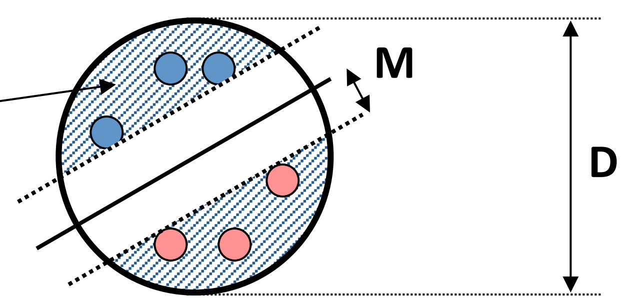
- How do we justify SRM?
- Result: SRM is always consistent and defines a bound on the rate of convergence



Gap Tolerant Classifiers (definition)

- Recall: for N-D linear classifiers, h = N+1
- Not quite satisfactory in practice!
- What if I have lots of redundant features (dimensions)? h should be less than N+1
- But VC estimate does not distinguish between such cases and cases where features are valuable!
- Solution: constrain linear classifiers to data inside a sphere
- *Gap Tolerant Classifier*: linear classifier whose activity is constrained to a sphere & outside a margin

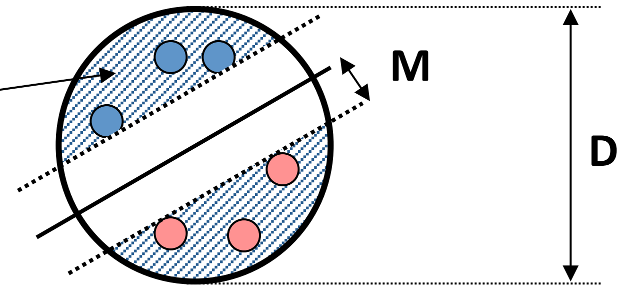
Only count errors in shaded region Elsewhere have L(x,y)=0



M=margin
D=diameter
d=dimensionality

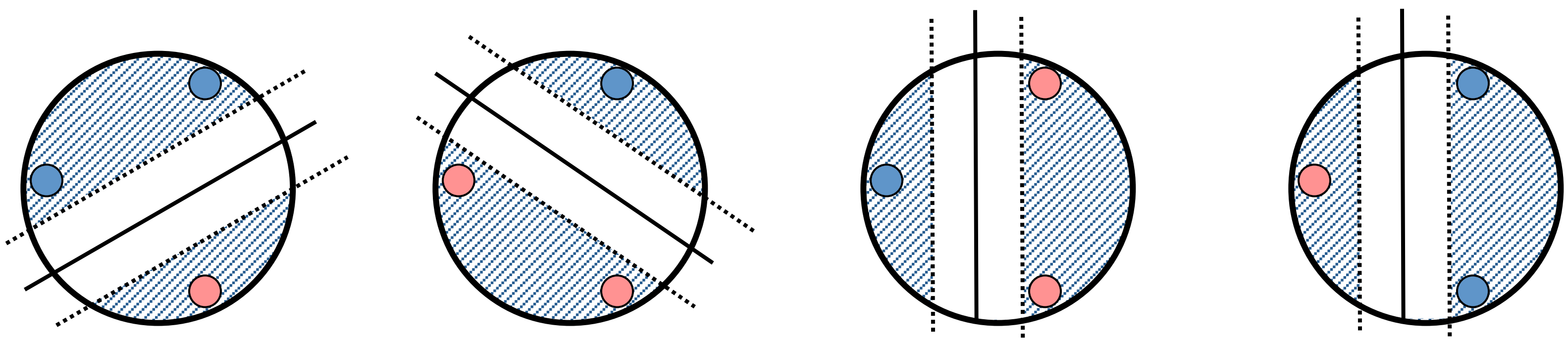
Gap Tolerant Classifiers (idea)

Only count errors in shaded region Elsewhere have L(x,y)=0

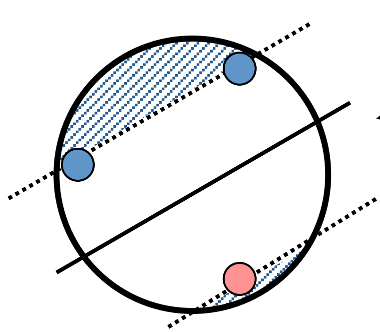


M=margin
D=diameter
d=dimensionality

- If M is small relative to D, can still shatter 3 points:

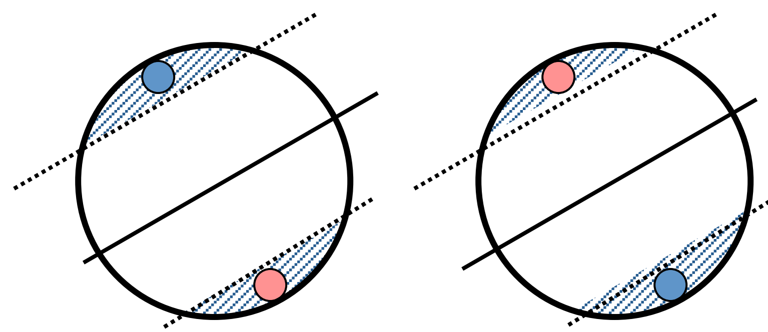


- But as M grows relative to D, can only shatter 2 points!



Can't shatter 3

Can shatter 2



Large Margin

- We have observed that: as the margin grows relative to data sphere, we can shatter fewer points
- In other words, the larger the margin, the smaller the VC dimension
- The general relation between h & M is expressed as:

$$h \leq \min\left\{ \left\lceil \frac{r^2}{M^2} \right\rceil,\ N \right\} + 1, \quad r = \max_i \|x_i\|$$

- Previously we just had h = N+1.
- Now we have a bound on h in terms of M and radius (r) of the data sphere
- This reflects a fairly typical case where the real data is bounded (if its not, then by default h = N+1)
- Note: sometimes bound is expressed in terms of diameter (margin is taken to be the width between the hyperplanes)
- General rule: maximizing margin reduces the VC dimension (inverse relation)

Relation to Perceptron

• ***Theorem***: assuming conditions {1,2} below are satisfied, the sequence of weight vectors determined by the online perceptron algorithm will converge to a solution vector in finite number of steps

1. Assume all data lies inside a sphere of radius r: $r = \max_i \|x_i\|$
2. Assume that the data is linearly separable: $\forall i : y_i((w^*)^T x_i) \geq \gamma > 0$

• The bound on the number of steps (k) is expressed in terms of the margin:

$$k \leq \frac{r^2}{\gamma^2} \|\mathbf{w}^*\|^2$$

Optimal Hyperplane (idea)

- Consider a linearly separable 2-class problem:
  - Data set: $\{(x_1,y_1),\ldots,(x_\ell,y_\ell)\}, \quad x_i \in \Re^n, y_i \in \{-1,1\}$
  - Decision boundary: $f(x;w) = w^T x + b = 0$
  - Symmetry:

$$\frac{w^T x_i + b > 0: \quad assign\ 1}{w^T x_i + b < 0: \quad assign\ -1} \Rightarrow y_i(w^T x_i + b) > 0$$

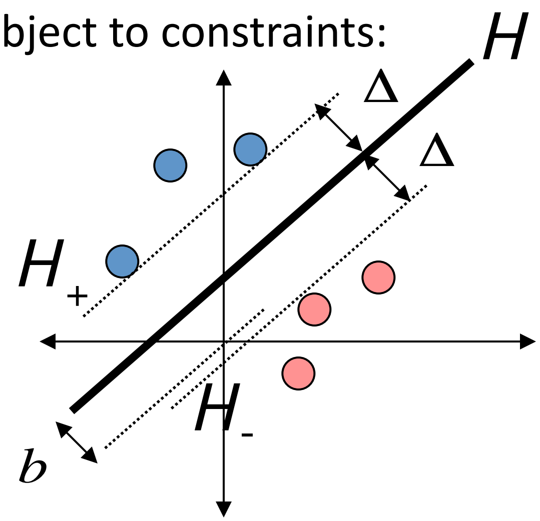
- There are many solutions (solution region). Perceptron chooses some solution vector
- Can we require that the hyperplane with maximum margin is selected?
- Can we guarantee it is unique?

Optimal Hyperplane (definition)

- Define two quantities: $h_1(w) = \min_{i:y_i=1}\left(w^T x_i\right),\ h_2(w) = \max_{i:y_i=-1}\left(w^T x_i\right)$
- Consider the unit vector $\mathbf{w_0}$ which maximizes margin subject to constraints:

$$\max_w \Delta(w) = \frac{h_1(w) - h_2(w)}{2}$$

$$s.t \quad \|w\| = 1, \quad \forall i:\ y_i(w^T x_i + b) > 0$$



- The vector $\mathbf{w}^*$ and the constant b^* determine the *maximal margin hyperplane* or the *optimal hyperplane* H

$$b^* = -\left(h_1(w^*) + h_2(w^*)\right)/2$$

- Note: the optimal hyperplane is unique (not proved here)

Better Formulation

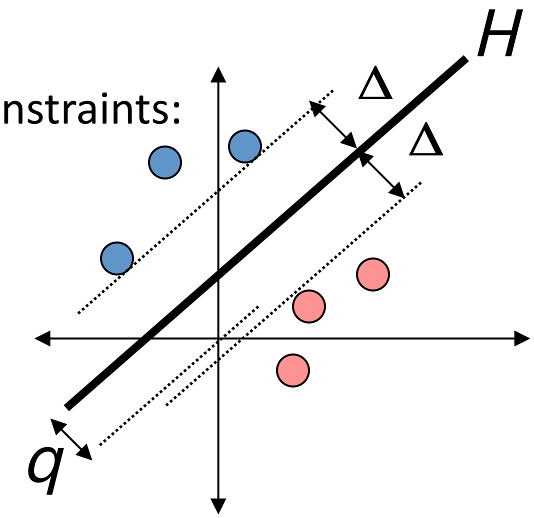
- Goal: find effective methods for constructing the optimal hyperplane
- Consider equivalent problem: instead of restricting the norm of the weight vector (hyperplane), lets scale the value of f(x) for the closest points to the hyperplane

$$\forall i: \; y_i(w^T x_i + b) \geq 1$$

- Now we are trying to minimize the norm subject to these constraints:

$$\min_w \frac{1}{2}\|w\|^2$$

$$s.t \quad \forall i: \; y_i(w^T x_i + b) \geq 1$$



- Not hard to show: if we normalize the vector which minimizes the above we obtain the unit vector solution **w*** on the previous slide
- Note: the distance to the origin is not just the value of b anymore (denoted q above)

Quadratic Program

- Recall geometry of linear surface: discriminant function ***f(x)*** is proportional to the distance from **x** to ***H***

$$dist = \frac{f(x)}{\|w\|} = \frac{(w^T x + b)}{\|w\|}, \quad dist2origin = q = \frac{f(0)}{\|w\|} = \frac{|b|}{\|w\|}$$

$$\text{margin} = \Delta = \frac{|f(x) = \pm 1|}{\|w\|} = \frac{1}{\|w\|}, \quad width = 2\Delta = \frac{2}{\|w\|}$$

- We have a quadratic program (QP), just plug into a solver (matlab: quadprog), done!

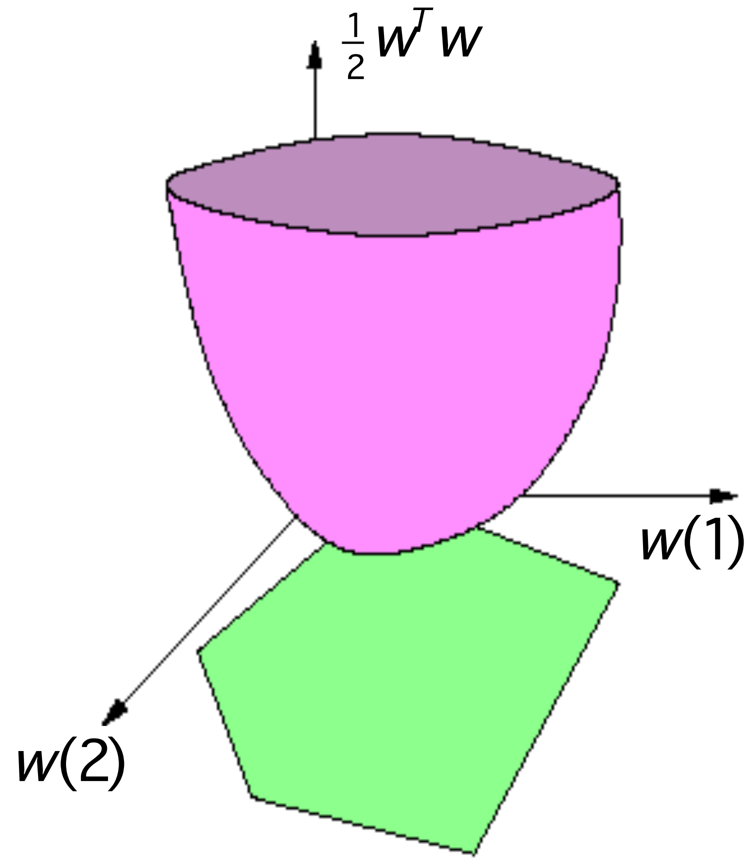
$$\min_w \frac{1}{2}\|w\|^2$$

$$s.t \quad \forall i:\ y_i(w^T x_i + b) \geq 1$$

- We would solve the problem in *primal space*, but can also solve it in dual space

QP Visualization

- Each data point adds a linear inequality to QP
- Each point cuts a half plane of allowable planes and reduces green region
- The optimal hyperplane is the closest point to the origin that is still in the green region
- The perceptron algorithm just puts us randomly in the green region

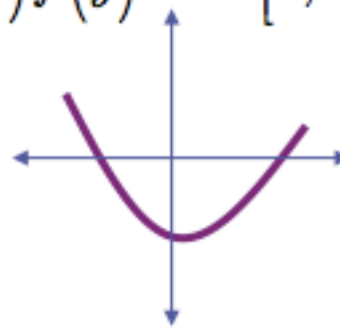


Convexity

- **Convex functions:** $f\left(tx+\left(1-t\right)y\right) \leq tf\left(x\right)+\left(1-t\right)f\left(y\right) \quad t\in\left[0,1\right]$

$$f\left(x\right)=\exp\left(x\right),\ f\left(\vec{x}\right)=\vec{x}^Tb+\tfrac{1}{2}\vec{x}^TH\vec{x},\ f\left(\vec{x}\right)=\vec{x}$$

Have non-negative second derivatives (bowls)

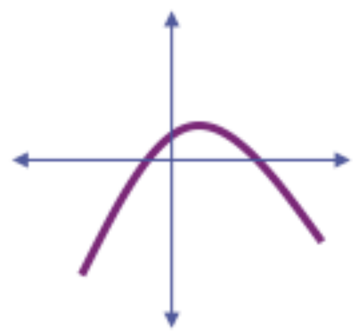


$$\frac{\partial^2 f\left(x\right)}{\partial x^2}=\exp\left(x\right),\ \frac{\partial^2 f\left(\vec{x}\right)}{\partial\vec{x}\partial\vec{x}}=H,\ \frac{\partial^2 f\left(\vec{x}\right)}{\partial\vec{x}\partial\vec{x}}=0$$

- **Concave functions:** $f\left(tx+\left(1-t\right)y\right) \geq tf\left(x\right)+\left(1-t\right)f\left(y\right) \quad t\in\left[0,1\right]$

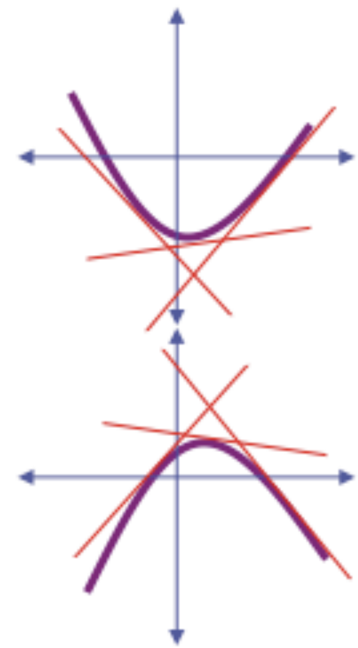
$$f\left(x\right)=\log\left(x\right),\ f\left(\vec{x}\right)=\vec{x}^Tb-\tfrac{1}{2}\vec{x}^TH\vec{x},\ f\left(\vec{x}\right)=\vec{x}$$

Have non-positive second derivatives (caves)



$$\frac{\partial^2 f\left(x\right)}{\partial x^2}=-\frac{1}{x^2},\ \frac{\partial^2 f\left(\vec{x}\right)}{\partial\vec{x}\partial\vec{x}}=-H,\ \frac{\partial^2 f\left(\vec{x}\right)}{\partial\vec{x}\partial\vec{x}}=0$$

Duality



- Every convex function f has a dual f*:
  All tangent lines below it form an epigraph
  The f* gives the intercept for each slope.

$$f(x) = \max_\lambda \left(x^T \lambda - f^*(\lambda) \right)$$

- Every concave function f has a dual f*
  All tangent lines above it form an epigraph
  The f* gives the intercept for each slope.

$$f(x) = \min_\lambda \left(x^T \lambda - f^*(\lambda) \right)$$

- This * is called the Legendre Transform or Fenchel Dual
- The dual of the dual f** is f
- Example: $f(x) = \frac{1}{2} c x^2 \quad \rightarrow \quad f^*(\lambda) = \frac{1}{2c} \lambda^2$
- We can replace a minimization over x like this

$$\min_x f(x) = \min_x \max_\lambda \left(\lambda x - f^*(\lambda) \right)$$

...and can work with a maximization of its dual instead

Optimization: Inequality Constraints

- Problem: given a function of several variables, find its stationary point subject to one inequality constraint
- Formally (general case):

$$\max_{\mathbf{x}} \; f(\mathbf{x})$$

$$s.t. \quad g(\mathbf{x}) \geq 0$$

- Consider the geometry of the problem, there are now two solutions possible:

  1. On the boundary (constraint is *active*, g(x) = 0)
  2. Inside the region (constraint is *inactive*, g(x) > 0)

- For case 2, the constraint has no effect. Case 1 is analogous to equality constraint discussed previously, but the sign of the multiplier is crucial (gradient should be oriented away from the region g(x) > 0)

Optimization: Inequality Constraints

- For case 2 (region), the constraint has no effect.
- Case 1 (boundary) is analogous to equality constraint discussed previously, but the sign of the multiplier is crucial (gradient should be oriented away from the region defined by the constraint g(x) > 0)

1. Boundary: $\nabla f(x) = -\lambda \nabla g(x),\ \lambda > 0$
2. Region: $\nabla f(x) = 0 \quad \equiv \quad \nabla L(x, \lambda = 0)$

- We can combine both cases into one: $\lambda g(x) = 0$

KKT Conditions

A. Define a function: $L(x,\lambda) = f(x) + \lambda g(x)$

B. Find the stationary point of ***L*** with respect to {x, λ} and subject to:

$$g(x) \geq 0, \quad \lambda \geq 0, \quad \lambda g(x) = 0$$

- These are known as the *Karush-Kuhn-Tucker* (KKT) conditions
- If we wish to minimize the function f(x) we need to define the Lagrangian as:

$$L(x,\lambda) = f(x) - \lambda g(x)$$

Multiple Constraints

- Problem: given a function of several variables, find its stationary point subject to one or more equality and inequality constraints
- Formally (general case): $\max_{\mathbf{x}} \; f(\mathbf{x})$

$$s.t. \quad g_j(\mathbf{x}) = 0, \quad j = 1,\ldots,J$$
$$h_k(\mathbf{x}) \geq 0, \quad k = 1,\ldots,K$$

- Define the Lagrangian:

$$L(\mathbf{x},\{\lambda\},\{\mu\}) = f(\mathbf{x}) + \sum_{j=1}^{J} \lambda_j g_j(\mathbf{x}) + \sum_{k=1}^{K} \mu_k h_k(\mathbf{x}),$$

$$s.t: \quad \forall k : \mu_k \geq 0, \; \mu_k h_k(\mathbf{x}) = 0$$

Dual Form Derivation

- Recall optimal hyperplane problem in primal space:

$$\min_w \frac{1}{2}\|w\|^2 \quad s.t \;\; \forall i: \; y_i(w^T x_i + b) \geq 1$$

- This is a convex program, define the Lagrangian and find stationary point:

$$L(w,b,\alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{\ell} \alpha_i \left[y_i(w^T x_i + b) - 1 \right], \; \alpha_i \geq 0$$

- Minimize L over {w,b}, maximize over {alphas}:

$$\frac{\partial L(w,b,\alpha)}{\partial w} = w - \sum_{i=1}^{\ell} \alpha_i y_i x_i = 0 \;\Rightarrow w = \sum_{i=1}^{\ell} y_i \alpha_i x_i$$

$$\frac{\partial L(w,b,\alpha)}{\partial b} = -\sum_{i=1}^{\ell} \alpha_i y_i = 0 \;\Rightarrow \sum_{i=1}^{\ell} y_i \alpha_i = 0$$

Dual Form

This is a convex program, define the Lagrangian and find stationary point:

$$L(w,b,\alpha) = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{\ell} \alpha_i \left[y_i (w^T x_i + b) - 1 \right], \ \alpha_i \geq 0$$

- Minimize L over {w,b}, maximize over {alphas}:

$$\frac{\partial L(w,b,\alpha)}{\partial w} \Rightarrow w = \sum_{i=1}^{\ell} y_i \alpha_i x_i, \quad \frac{\partial L(w,b,\alpha)}{\partial b} \Rightarrow \sum_{i=1}^{\ell} y_i \alpha_i = 0, \quad \alpha_i \geq 0$$

- Plug back into the Lagrangian and get the dual form:

$$\max_{\alpha} D(\alpha) = \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} y_i y_j \alpha_i \alpha_j \left(x_i \cdot x_j \right)$$

$$s.t: \ \sum_{i=1}^{\ell} y_i \alpha_i = 0, \ \alpha_i \geq 0$$

Why Solve in Dual Space?

- QP runs in cubic polynomial time (in terms of # of variables)
- QP in primal space has complexity O(d^3), where d is the dimensionality of the input vectors (weight vector)
- QP in dual space has complexity O(ell^3), where ell is the number of examples
- More importantly: dual space yields “deeper results”

$$\min_{w} \frac{1}{2}\|w\|^2$$

$$s.t\ \ \forall i:\ y_i(w^T x_i + b) \geq 1$$

$$\max_{\alpha} D(\alpha) = \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{\ell} y_i y_j \alpha_i \alpha_j \left(x_i \cdot x_j\right)$$

$$s.t:\ \sum_{i=1}^{\ell} y_i \alpha_i = 0,\ \alpha_i \geq 0$$

$$\max_{\alpha} D(\alpha) \Rightarrow \alpha^* \Rightarrow w^* = \sum_{i=1}^{\ell} y_i \alpha_i^* x_i$$