Language and Statistics II

Lecture 16: Going Discriminative (part two)

Noah Smith

Lecture Overview

- Quick review
- Maximum margin training
  - Nonseparable data
  - Hinge loss
  - Training
    - Dual
    - Sparsity and support vectors
    - Factored structure prediction with SVMs
  - Kernels
  - MIRA
- Discriminative methods in general:
  - Bringing in “global” features
  - Reranking

Note: Much material was adapted from the Klein & Taskar ACL 2005 tutorial. Highly recommended reading!

Quick Review

- Motivation: only model/discriminate what is necessary.
- Perceptron: find *a* linear separator.
- Exp-loss and boosting
- Log-loss
  = conditional estimation of a log-linear model
  = maximum "softmax" margin
- Maximum margin, arbitrary loss function
  A QP with way too many constraints!

(Multiclass) Support Vector Machines

First form:

Note constraint on **w**. This prevents us from cheating by using really big weights. (Can think of it as built-in regularization.)

$$\max_{\mathbf{w}:\frac{1}{2}\mathbf{w}\cdot\mathbf{w}\leq 1} \gamma$$

$$s.t.\ \forall i, \forall y \in \mathrm{GEN}(x_i),$$

$$\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \mathbf{w}\cdot\mathbf{f}(x_i, y) \geq \gamma\ell(y, y_i; x_i)$$

Second form: change of variable.

Note that the objective is quadratic (indeed, psd!), and the constraints are linear.

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w}$$

$$s.t.\ \forall i, \forall y \in \mathrm{GEN}(x_i),$$

$$\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \mathbf{w}\cdot\mathbf{f}(x_i, y) \geq \ell(y, y_i; x_i)$$

(Multiclass) Support Vector Machines

Intuition: find weights that make alternative, incorrect *y* "as far away as they are bad."

badness = loss

far-away-ness = margin

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w}$$

$$s.t.\ \forall i, \forall y \in \mathrm{GEN}(x_i),$$

$$\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \mathbf{w}\cdot\mathbf{f}(x_i, y) \geq \ell(y, y_i; x_i)$$

(Multiclass) Support Vector Machines

Bad news: one constraint for every wrong *y* for every example!

(Think about parsing or sequences … exponentially bad!)

Bad news: what if the data aren't separable?

$$\min_{\mathbf{w}} \frac{1}{2} \mathbf{w} \cdot \mathbf{w}$$

$$s.t.\ \forall i, \forall y \in \mathrm{GEN}(x_i),$$

$$\mathbf{w} \cdot \mathbf{f}(x_i, y_i) - \mathbf{w} \cdot \mathbf{f}(x_i, y) \geq \ell(y, y_i; x_i)$$

Slack Variable for Non-Separability

"Cut the constraints some slack" - loss on *i*th example diminished by ξ_i.

Objective pays proportional to the amount of slack.

C is "capacity." Larger *C* = more smoothing.

$$\min_{\mathbf{w},\xi} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} + C\sum_i \xi_i$$

$$s.t.\ \forall i, \forall y \in \mathrm{GEN}(x_i),$$

$$\mathbf{w}\cdot\mathbf{f}(x_i,y_i) - \mathbf{w}\cdot\mathbf{f}(x_i,y) \geq \ell(y,y_i;x_i) - \xi_i$$

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w}$$

$$s.t.\ \forall i, \forall y \in \mathrm{GEN}(x_i),$$

$$\mathbf{w}\cdot\mathbf{f}(x_i,y_i) - \mathbf{w}\cdot\mathbf{f}(x_i,y) \geq \ell(y,y_i;x_i)$$

Solving for ξ_i

$$\forall i, \forall y \in \mathrm{GEN}(x_i),$$

$$\mathbf{w} \cdot \mathbf{f}(x_i, y_i) - \mathbf{w} \cdot \mathbf{f}(x_i, y) \geq \ell(y, y_i; x_i) - \xi_i$$

$$\xi_i \geq \ell(y, y_i; x_i) - \mathbf{w} \cdot \mathbf{f}(x_i, y_i) + \mathbf{w} \cdot \mathbf{f}(x_i, y)$$

$$\forall i, \quad \xi_i = \max_{y \in \mathrm{GEN}(x_i)} \left[\ell(y, y_i; x_i) + \mathbf{w} \cdot \mathbf{f}(x_i, y) \right] - \mathbf{w} \cdot \mathbf{f}(x_i, y_i)$$

Having solved for the slack variable, we can substitute for it!

$$\min_{\mathbf{w}} \frac{C'}{2} \mathbf{w} \cdot \mathbf{w} - \sum_i \left(\mathbf{w} \cdot \mathbf{f}(x_i, y_i) - \max_{y \in \mathrm{GEN}(x_i)} \left[\mathbf{w} \cdot \mathbf{f}(x_i, y) + \ell(y, y_i; x_i) \right] \right)$$

“Min-max” formulation …

Compare with Log-loss (again)

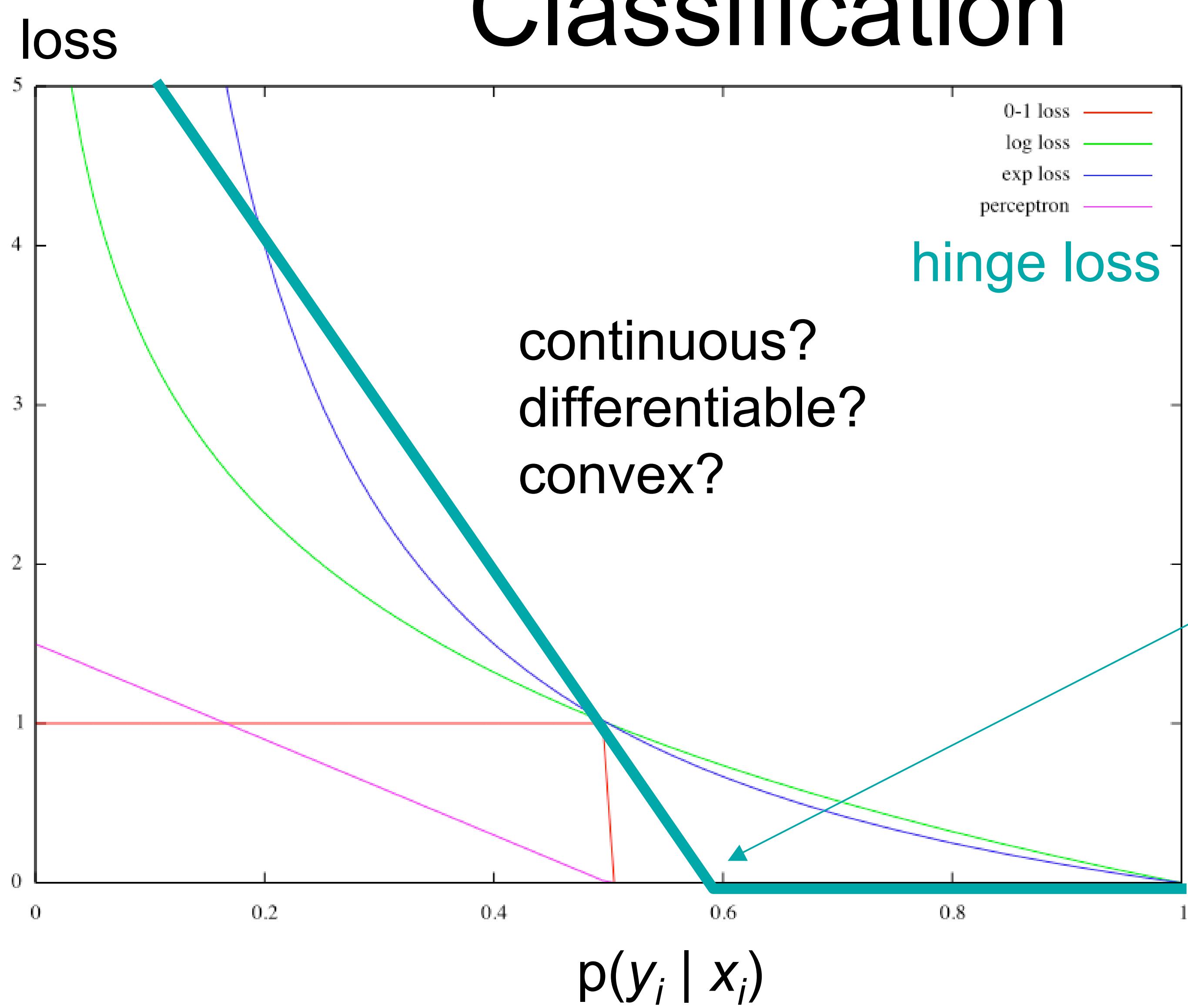
$$\min_{\mathbf{w}} \frac{C'}{2} \mathbf{w} \cdot \mathbf{w} - \sum_i \left(\mathbf{w} \cdot \mathbf{f}(x_i, y_i) - \log \sum_{y \in \mathrm{GEN}(x_i)} \exp\left[\mathbf{w} \cdot \mathbf{f}(x_i, y) \right] \right)$$

Conditional training for log-linear models (with quadratic regularizer/Gaussian prior)

$$\min_{\mathbf{w}} \frac{C'}{2} \mathbf{w} \cdot \mathbf{w} - \sum_i \left(\mathbf{w} \cdot \mathbf{f}(x_i, y_i) - \max_{y \in \mathrm{GEN}(x_i)} \left[\mathbf{w} \cdot \mathbf{f}(x_i, y) + \ell(y, y_i; x_i) \right] \right)$$

“Min-max” formulation of the SVM objective.

Loss Functions for Binary Classification



once y_i wins by "enough," objective stops pushing for greater separation

Making Training Tractable

- Let's use the slack variable formulation for now.
- To get rid of the exponentially many constraints, we must use **Lagrange multipliers**.

$$\min_{\mathbf{w},\xi} \frac{1}{2} \mathbf{w} \cdot \mathbf{w} + C \sum_i \xi_i$$

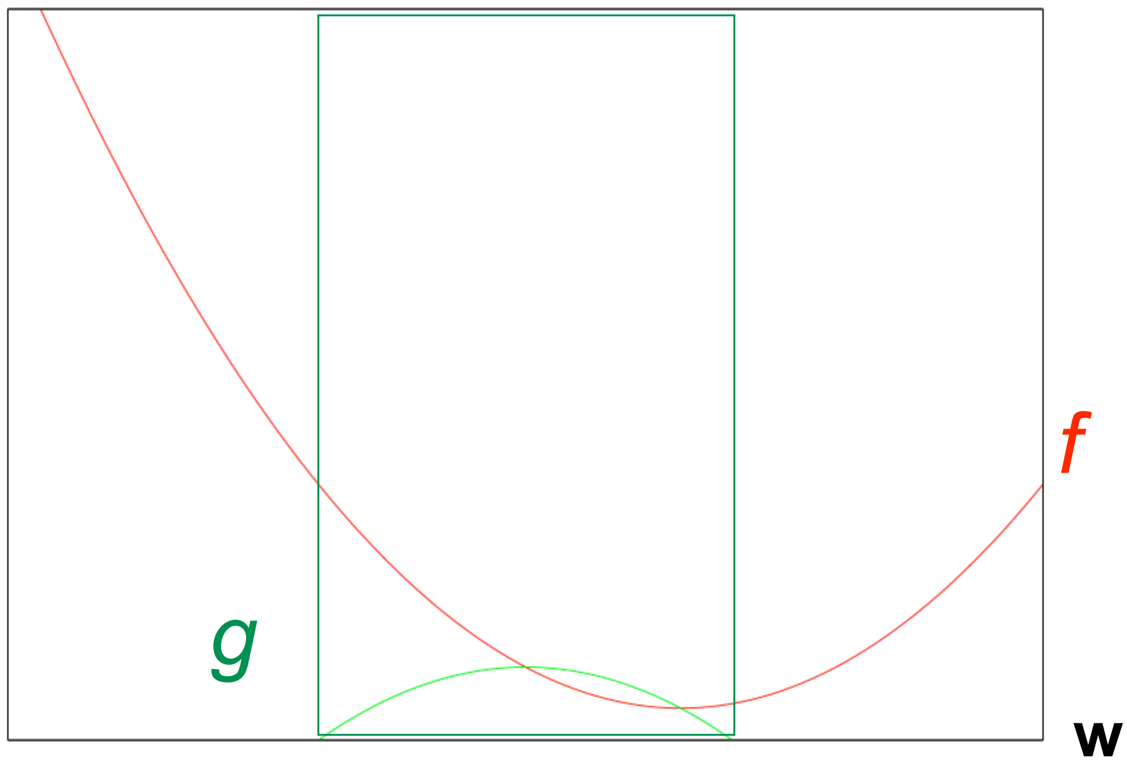
$$s.t.\ \forall i, \forall y \in \mathrm{GEN}(x_i),$$

$$\mathbf{w} \cdot \mathbf{f}(x_i, y_i) - \mathbf{w} \cdot \mathbf{f}(x_i, y) \geq \ell(y, y_i; x_i) - \xi_i$$

Mini-course on Lagrange Multipliers

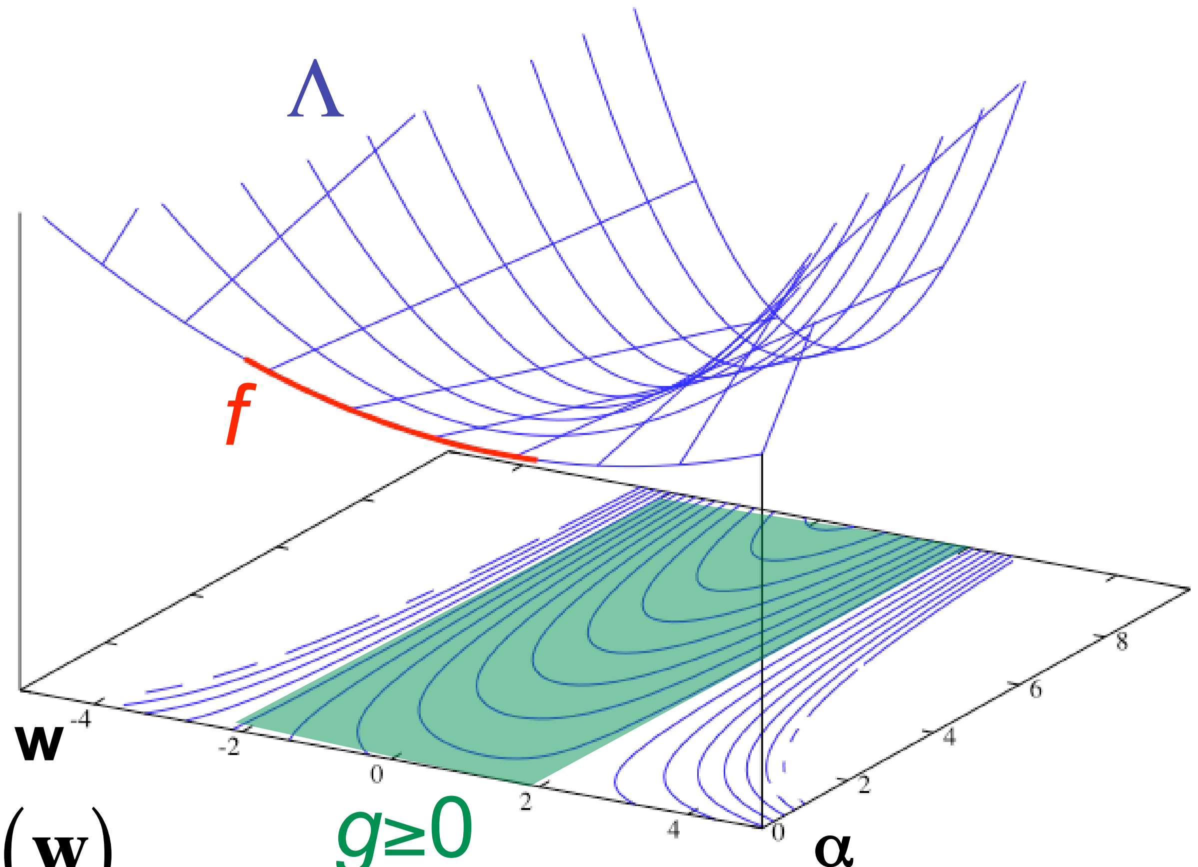
- These shouldn't be too new to you.
- We have used them twice before!
  - To prove that relative frequencies maximize likelihood for multinomials.
  - To derive (unconstrained) maximum likelihood from (constrained) maximum entropy.
- This should not be scary!

Lagrange Duality



$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}:g(\mathbf{w})\geq 0} f\left(\mathbf{w}\right)$$

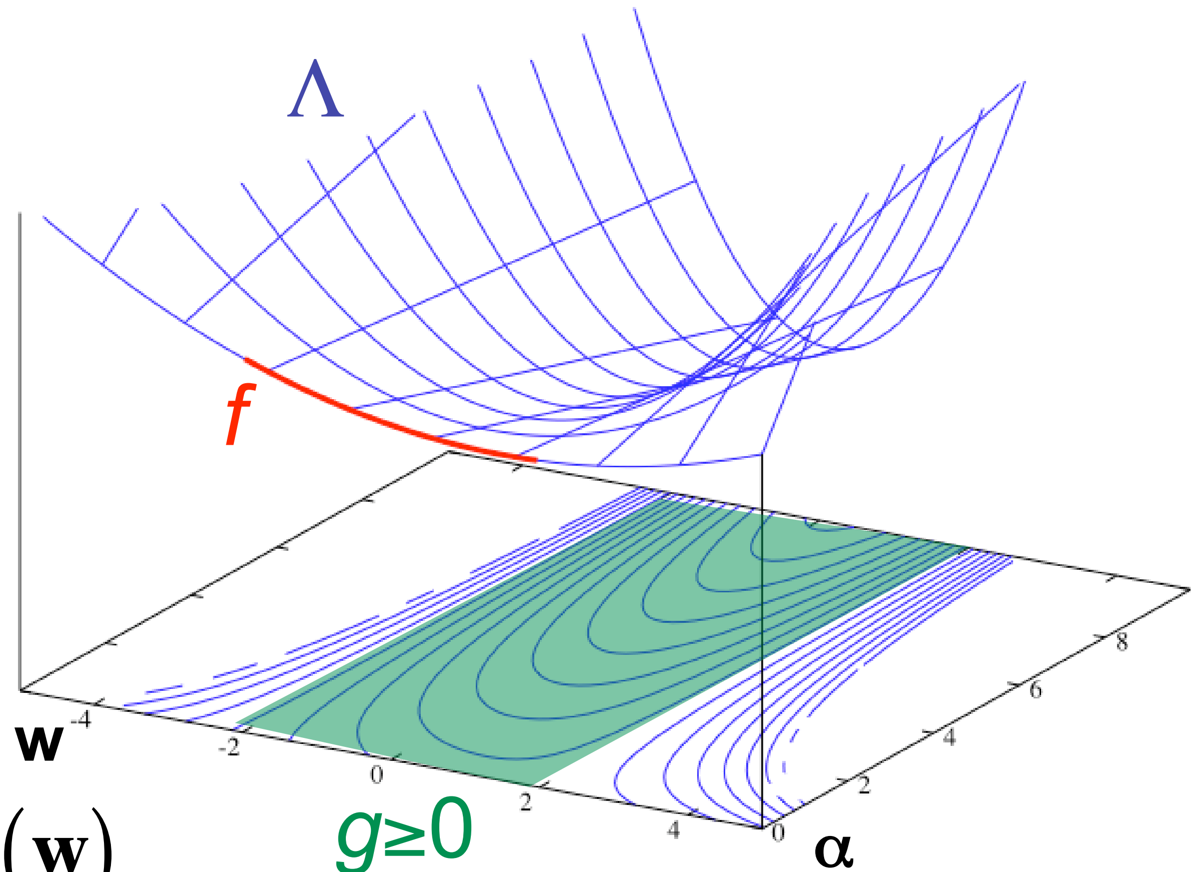
Lagrange Duality



$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}:g(\mathbf{w})\geq 0} f\left(\mathbf{w}\right)$$

$$\Lambda\left(\mathbf{w},\alpha\right) = f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right)$$

$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}} \max_{\alpha:\alpha\geq 0} \Lambda\left(\mathbf{w},\alpha\right) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda\left(\mathbf{w},\alpha\right)$$



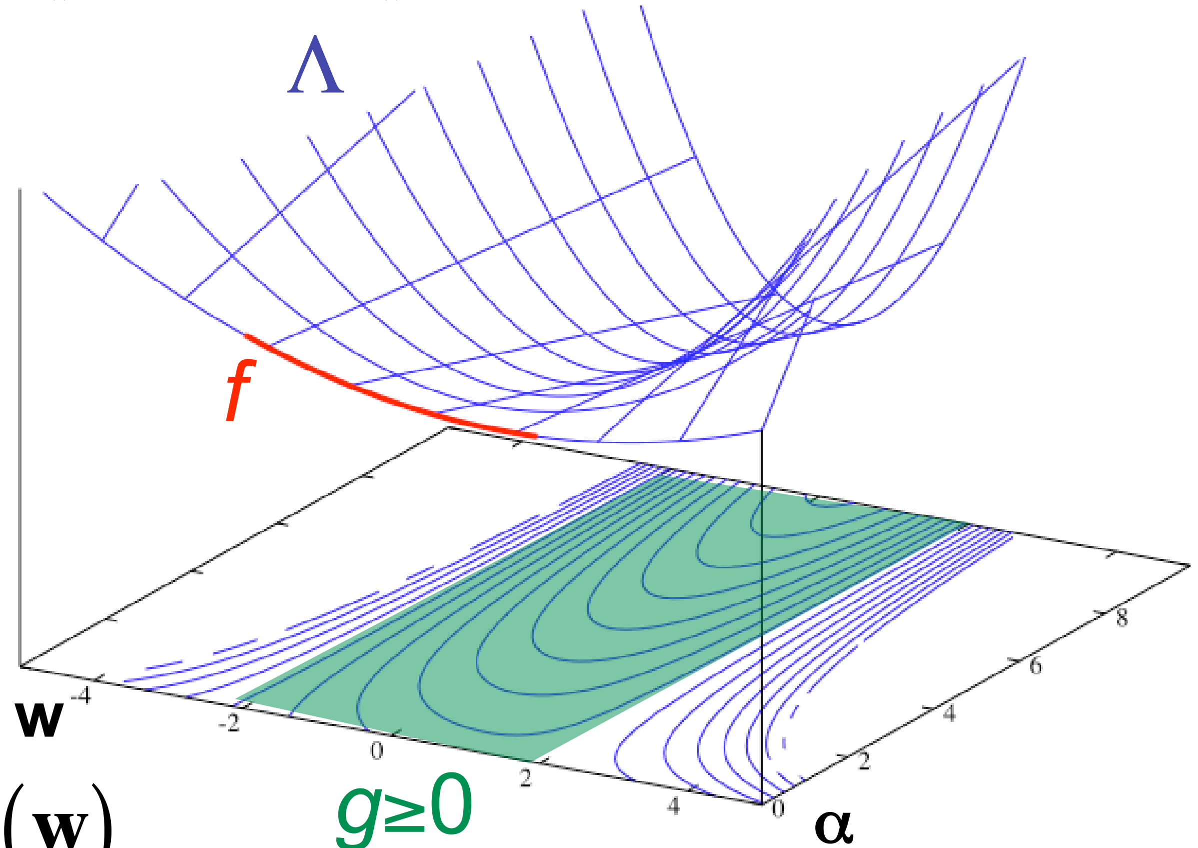
$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}:g(\mathbf{w})\geq 0} f\left(\mathbf{w}\right)$$

$$\Lambda\left(\mathbf{w},\alpha\right) = f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right)$$

$$\boxed{f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}} \max_{\alpha:\alpha\ge 0} \Lambda\left(\mathbf{w},\alpha\right) = \max_{\alpha:\alpha\ge 0} \min_{\mathbf{w}} \Lambda\left(\mathbf{w},\alpha\right)}$$

primal

$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}} \Lambda\left(\mathbf{w}\right) = \min_{\mathbf{w}} \max_{\alpha:\alpha\ge 0} \Lambda\left(\mathbf{w},\alpha\right)$$



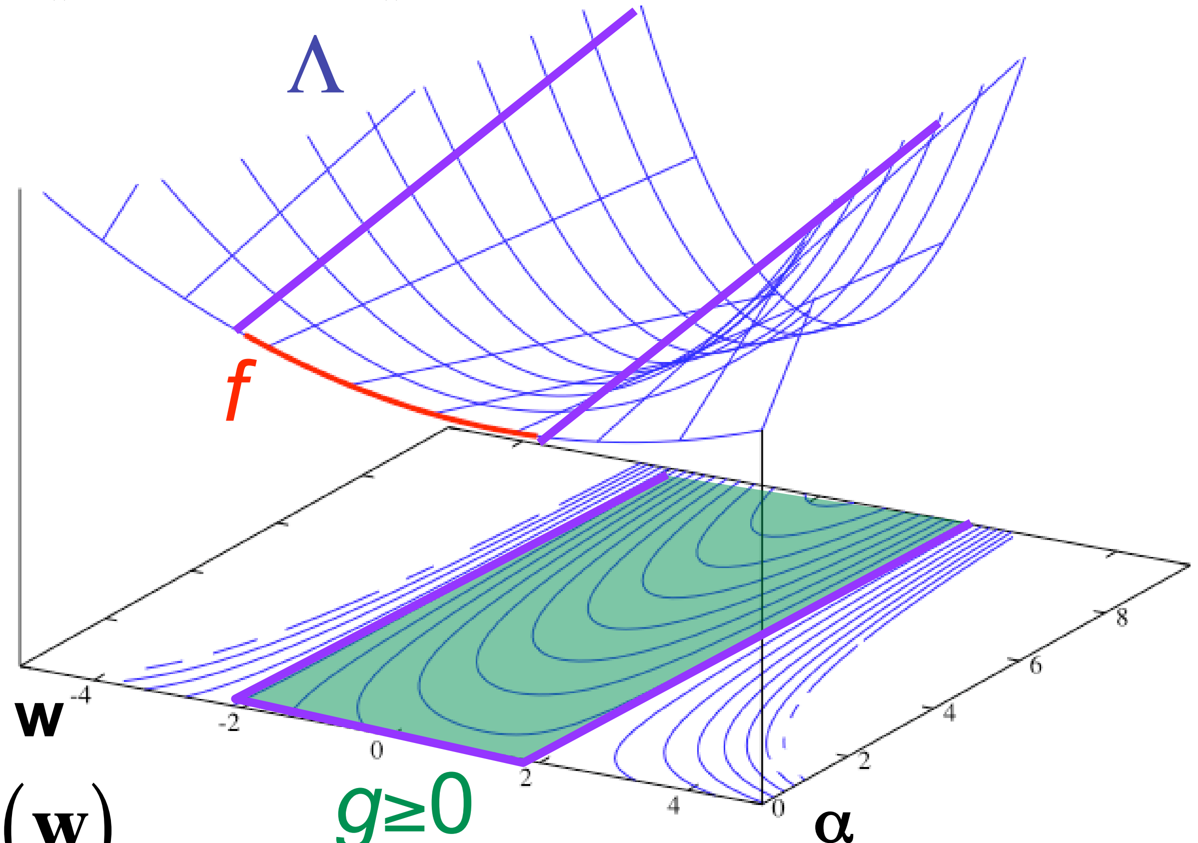
$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}:g(\mathbf{w})\ge 0} f\left(\mathbf{w}\right)$$

$$\Lambda\left(\mathbf{w},\alpha\right) = f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right) \qquad \Lambda\left(\mathbf{w}\right) = \max_{\alpha:\alpha\ge 0}\left[f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right)\right]$$

$$\boxed{f(\mathbf{w}^*) = \min_{\mathbf{w}} \max_{\alpha:\alpha\geq 0} \Lambda(\mathbf{w},\alpha) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda(\mathbf{w},\alpha)}$$

primal

$$f(\mathbf{w}^*) = \min_{\mathbf{w}} \Lambda(\mathbf{w}) = \min_{\mathbf{w}} \max_{\alpha:\alpha\geq 0} \Lambda(\mathbf{w},\alpha)$$



$$f(\mathbf{w}^*) = \min_{\mathbf{w}:g(\mathbf{w})\geq 0} f(\mathbf{w})$$

$$\Lambda(\mathbf{w},\alpha) = f(\mathbf{w}) - \alpha \cdot g(\mathbf{w}) \qquad \Lambda(\mathbf{w}) = \max_{\alpha:\alpha\geq 0}\left[f(\mathbf{w}) - \alpha \cdot g(\mathbf{w})\right]$$

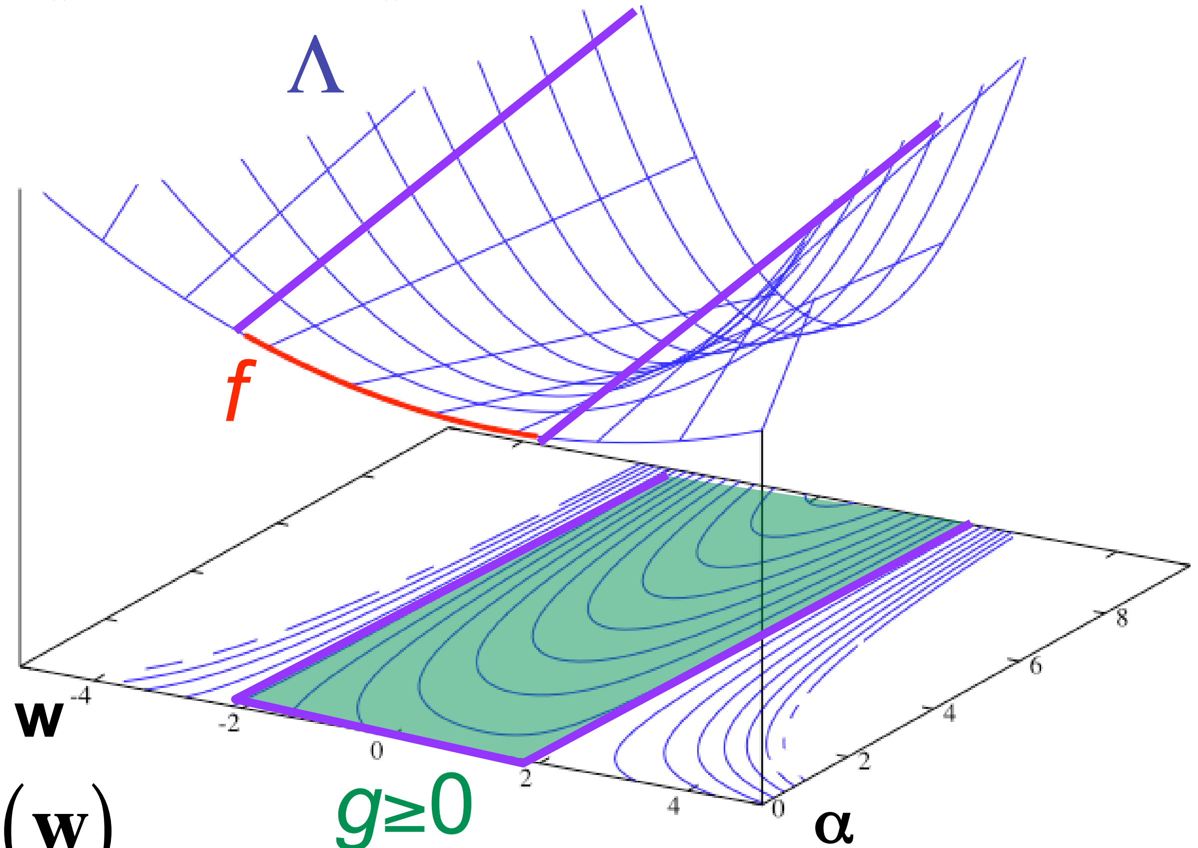
$$f(\mathbf{w}^*) = \min_{\mathbf{w}} \max_{\alpha:\alpha\geq 0} \Lambda(\mathbf{w},\alpha) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda(\mathbf{w},\alpha)$$

primal

$$f(\mathbf{w}^*) = \min_{\mathbf{w}} \Lambda(\mathbf{w}) = \min_{\mathbf{w}} \max_{\alpha:\alpha\geq 0} \Lambda(\mathbf{w},\alpha)$$

Inside the feasible region, the maximizing α is 0. $\Lambda(\mathbf{w})$ tracks $f(\mathbf{w})$.

Outside the feasible region, the maximizing α goes to ∞. So does $\Lambda(\mathbf{w})$!



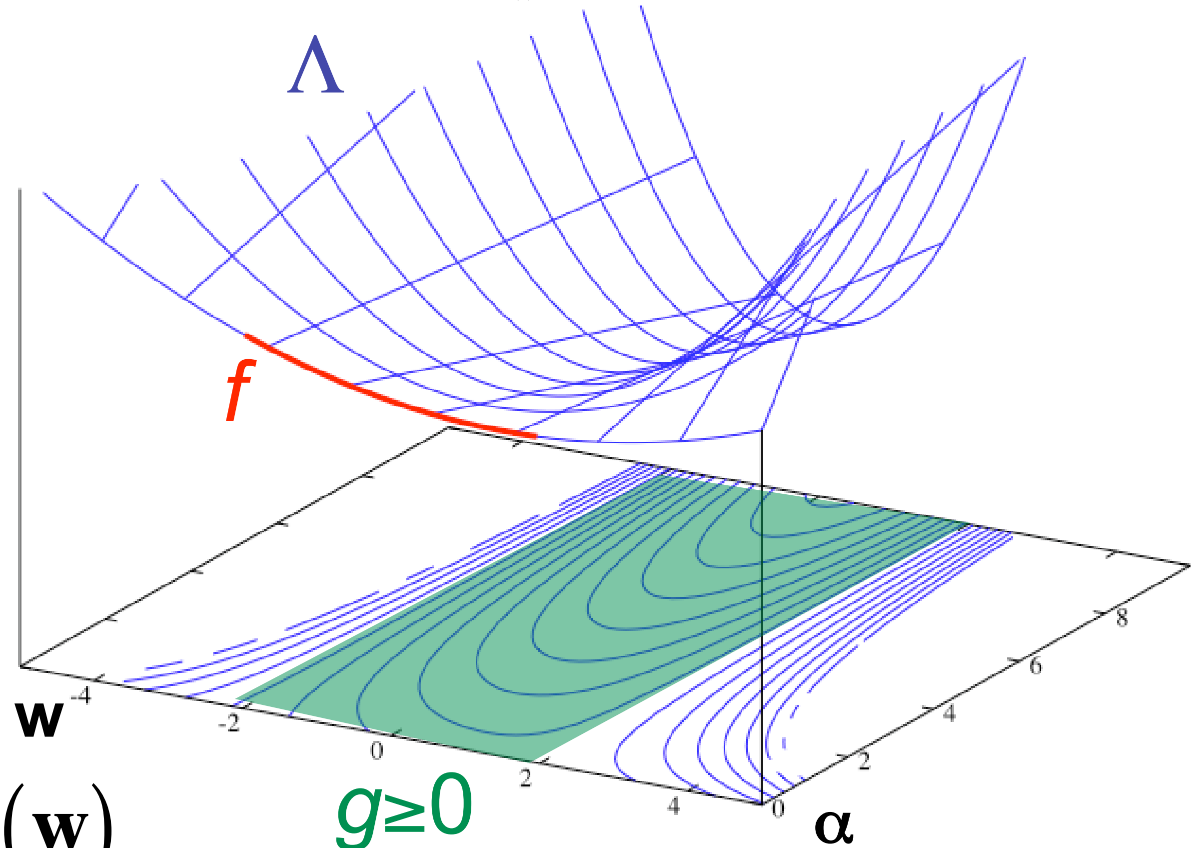
$$f(\mathbf{w}^*) = \min_{\mathbf{w}:g(\mathbf{w})\geq 0} f(\mathbf{w})$$

$$\Lambda(\mathbf{w},\alpha) = f(\mathbf{w}) - \alpha \cdot g(\mathbf{w}) \qquad \Lambda(\mathbf{w}) = \max_{\alpha:\alpha\geq 0}\left[f(\mathbf{w}) - \alpha \cdot g(\mathbf{w})\right]$$

$$f(\mathbf{w}^*) = \min_{\mathbf{w}} \max_{\alpha:\alpha\geq 0} \Lambda(\mathbf{w},\alpha) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda(\mathbf{w},\alpha)$$

dual

$$f(\mathbf{w}^*) = \max_{\alpha:\alpha\geq 0} \Lambda(\alpha) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda(\mathbf{w},\alpha)$$



$$f(\mathbf{w}^*) = \min_{\mathbf{w}:g(\mathbf{w})\geq 0} f(\mathbf{w})$$

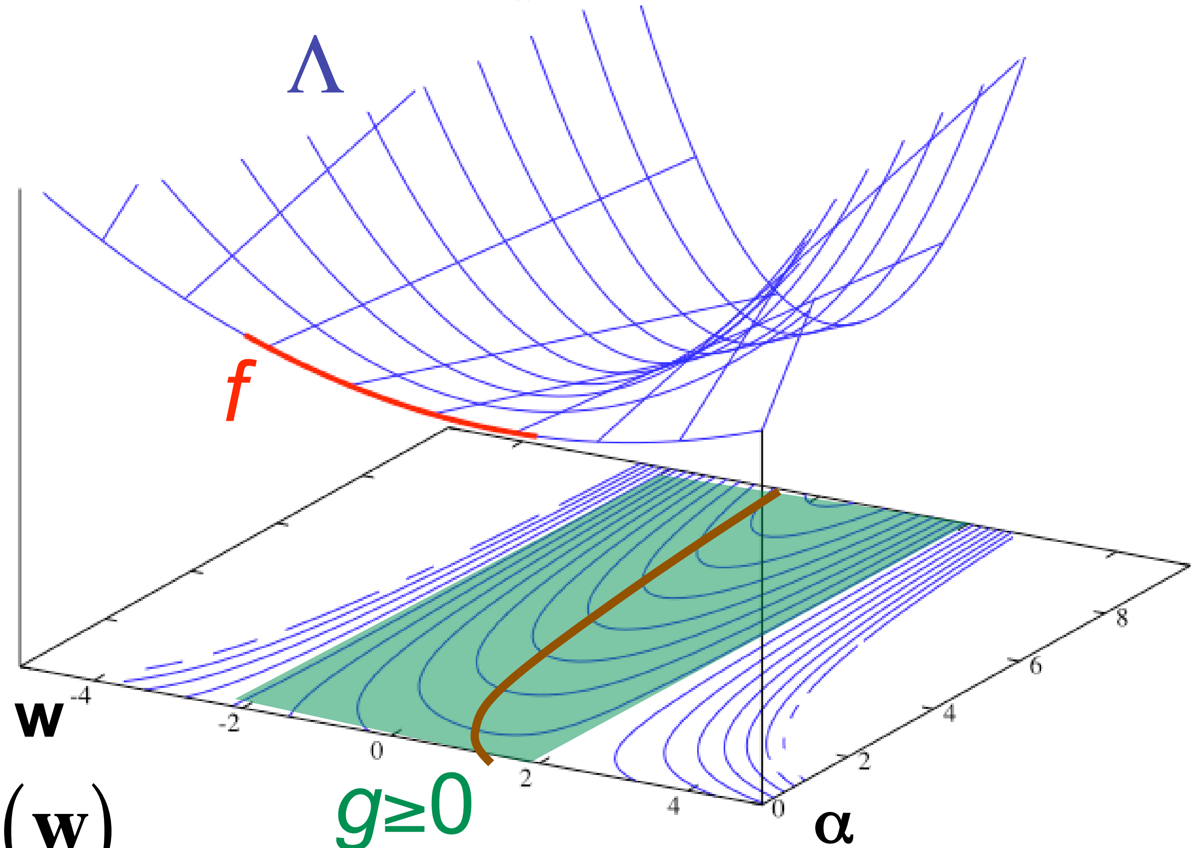
$$\Lambda(\mathbf{w},\alpha) = f(\mathbf{w}) - \alpha \cdot g(\mathbf{w}) \qquad \Lambda(\alpha) = \min_{\mathbf{w}} \left[f(\mathbf{w}) - \alpha \cdot g(\mathbf{w}) \right]$$

$$\boxed{f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}} \max_{\alpha:\alpha\geq 0} \Lambda\left(\mathbf{w},\alpha\right) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda\left(\mathbf{w},\alpha\right)}$$

dual

$$f\left(\mathbf{w}^*\right) = \max_{\alpha:\alpha\geq 0} \Lambda\left(\alpha\right) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda\left(\mathbf{w},\alpha\right)$$

What do we know about the α that maximizes $\Lambda(\alpha)$?



$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}:g(\mathbf{w})\geq 0} f\left(\mathbf{w}\right)$$

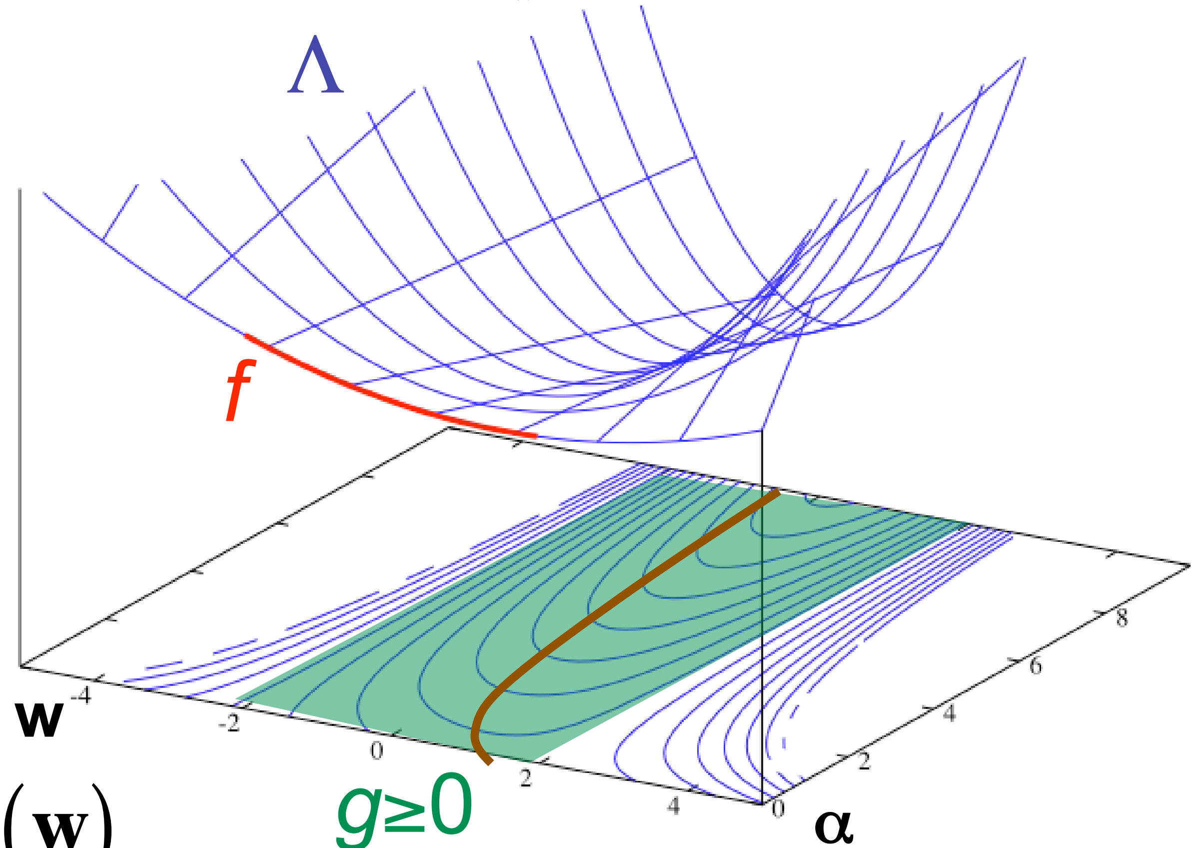
$$\Lambda\left(\mathbf{w},\alpha\right) = f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right) \qquad \Lambda\left(\alpha\right) = \min_{\mathbf{w}}\left[f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right)\right]$$

$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}} \max_{\alpha:\alpha\geq 0} \Lambda\left(\mathbf{w},\alpha\right) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda\left(\mathbf{w},\alpha\right)$$

dual

$$f\left(\mathbf{w}^*\right) = \max_{\alpha:\alpha\geq 0} \Lambda\left(\alpha\right) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda\left(\mathbf{w},\alpha\right)$$

If the constraint is **inactive** ($g > 0$) at the minimum, then the solution is $\alpha = 0$.



$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}:g(\mathbf{w})\geq 0} f\left(\mathbf{w}\right)$$

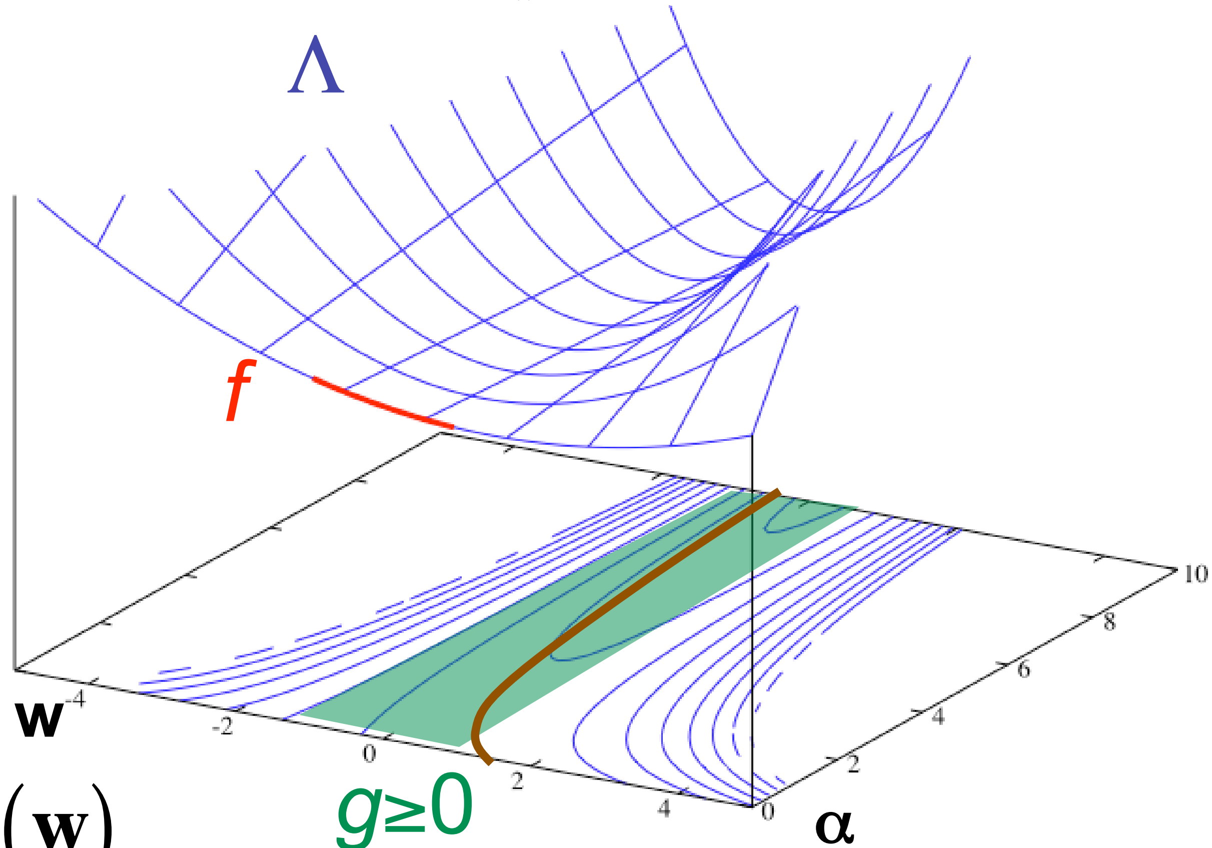
$$\Lambda\left(\mathbf{w},\alpha\right) = f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right) \qquad \Lambda\left(\alpha\right) = \min_{\mathbf{w}}\left[f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right)\right]$$

$$\boxed{f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}} \max_{\alpha:\alpha\geq 0} \Lambda\left(\mathbf{w},\alpha\right) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda\left(\mathbf{w},\alpha\right)}$$

dual

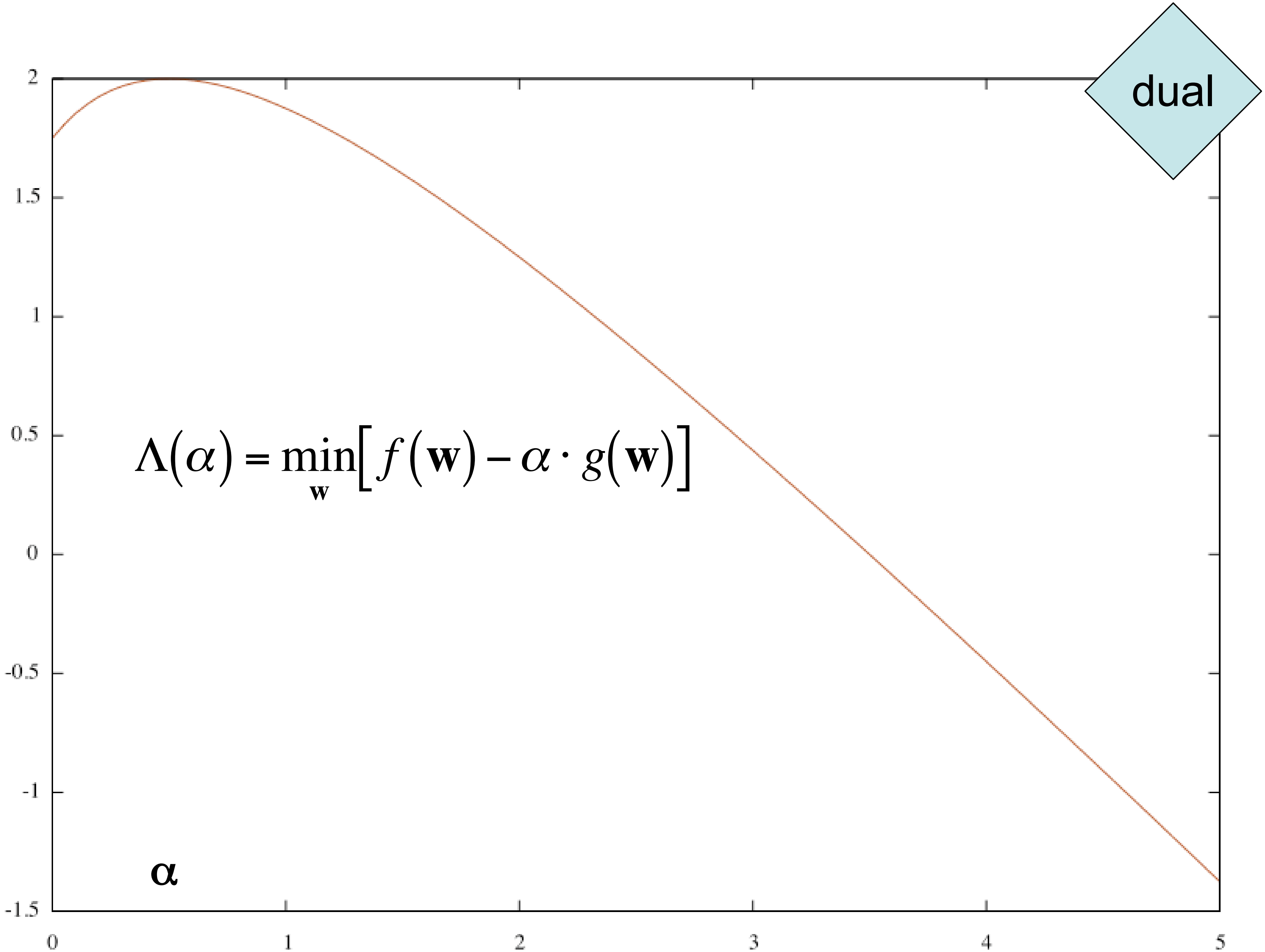
$$f\left(\mathbf{w}^*\right) = \max_{\alpha:\alpha\geq 0} \Lambda\left(\alpha\right) = \max_{\alpha:\alpha\geq 0} \min_{\mathbf{w}} \Lambda\left(\mathbf{w},\alpha\right)$$

If the constraint is **active** ($g = 0$) at the minimum, then …



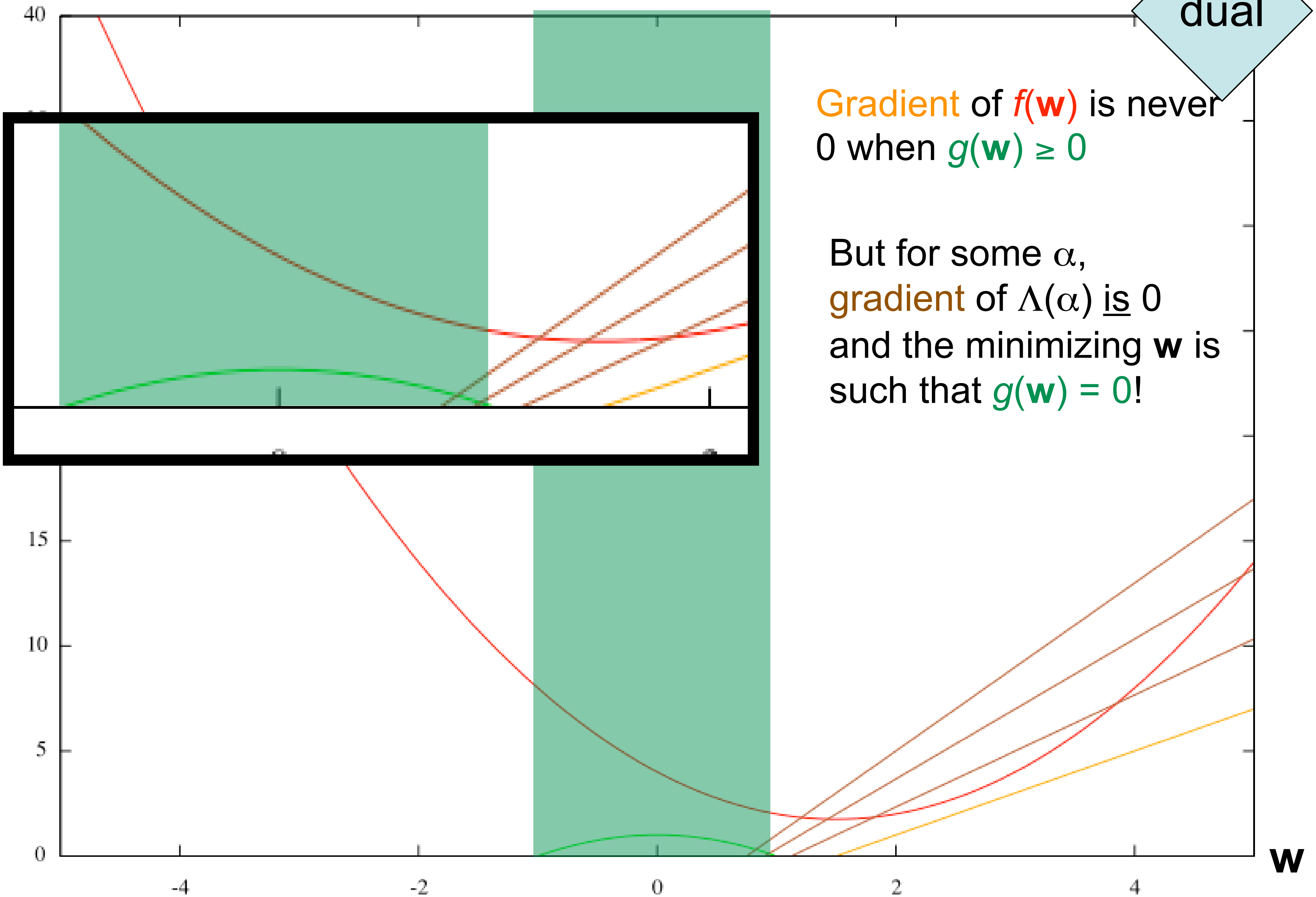
$$f\left(\mathbf{w}^*\right) = \min_{\mathbf{w}:g(\mathbf{w})\geq 0} f\left(\mathbf{w}\right)$$

$$\Lambda\left(\mathbf{w},\alpha\right) = f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right) \qquad \Lambda\left(\alpha\right) = \min_{\mathbf{w}}\left[f\left(\mathbf{w}\right) - \alpha \cdot g\left(\mathbf{w}\right)\right]$$



dual
$\Lambda(\alpha) = \min_{\mathbf{w}} \left[f(\mathbf{w}) - \alpha \cdot g(\mathbf{w}) \right]$
α
2
1.5
1
0.5
0
-0.5
-1
-1.5
0
1
2
3
4
5

dual



Gradient of $f(\mathbf{w})$ is never 0 when $g(\mathbf{w}) \geq 0$

But for some α, gradient of $\Lambda(\alpha)$ <u>is</u> 0 and the minimizing **w** is such that $g(\mathbf{w}) = 0$!

Primal and Dual

Primal:

- Infinite penalty for not meeting the constraints.
- Optimizing α^* will always be zero in feasible region.

Dual:

- Solve analytically for **w** in terms of α.
- Gradient of constraint "makes up for" nonzero gradient of *f*, if necessary … pushing **w** to feasible boundary.
- Maximizing w.r.t. α gives a feasible, optimal solution.
- Then go back and solve for **w**.

Back to SVMs

- Just like in the example, the max margin objective has primal and dual forms.
- Slack variable version:

$$\min_{\mathbf{w},\xi} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} + C\sum_i \xi_i$$

$$s.t.\ \forall i, \forall y \in \mathrm{GEN}(x_i),\ \mathbf{w}\cdot\mathbf{f}(x_i,y_i) - \mathbf{w}\cdot\mathbf{f}(x_i,y) \geq \ell(y,y_i;x_i) - \xi_i$$

- Primal:

$$\min_{\mathbf{w},\xi}\max_{\alpha:\alpha\geq 0} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} + C\sum_i \xi_i - \sum_i \sum_{y\in\mathrm{GEN}(x_i)} \alpha_{i,y}\left[\mathbf{w}\cdot\mathbf{f}(x_i,y_i) - \mathbf{w}\cdot\mathbf{f}(x_i,y) - \ell(y,y_i;x_i) + \xi_i\right]$$

- Dual:

$$\max_{\alpha:\alpha\geq 0}\min_{\mathbf{w},\xi} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} + C\sum_i \xi_i - \sum_i \sum_{y\in\mathrm{GEN}(x_i)} \alpha_{i,y}\left[\mathbf{w}\cdot\mathbf{f}(x_i,y_i) - \mathbf{w}\cdot\mathbf{f}(x_i,y) - \ell(y,y_i;x_i) + \xi_i\right]$$

The Key Trick

- Think of the Lagrange multipliers ($\alpha_{i,y}$) as **constants**.
- Solve for **w** and ξ analytically in terms of the $\alpha_{i,y}$. (How?)
- Then optimize over values of $\alpha_{i,y}$ only.
- You should be able to then show that:

$$\sum_i \sum_{y \in \mathrm{GEN}(x_i)} \alpha_{i,y} = C$$

$$\mathbf{w} = \sum_i \sum_{y \in \mathrm{GEN}(x_i)} \alpha_{i,y} \left(\mathbf{f}(x_i, y_i) - \mathbf{f}(x_i, y) \right)$$

$$\Lambda(\alpha) = \min_{\mathbf{w},\xi} \Lambda(\mathbf{w}, \xi, \alpha) = -\frac{1}{2} \left\| \sum_i \sum_{y \in \mathrm{GEN}(x_i)} \alpha_{i,y} \left(\mathbf{f}(x_i, y_i) - \mathbf{f}(x_i, y) \right) \right\|^2 + \sum_i \sum_{y \in \mathrm{GEN}(x_i)} \alpha_{i,y} \ell(y, y_i; x_i)$$

The Dual Problem

- So solve for the αs and then compute **w**.
- Each $\alpha_{i,y}$ corresponds to a constraint
  - $\alpha_{i,y}$ is only positive if the (i, y) constraint is active; then y is a **support vector**.
- Now only have nonnegativity constraints on $\alpha_{i,y}$.
- But for exponential-sized GEN, still too many variables!

$$\mathbf{w} = \sum_i \sum_{y \in \mathrm{GEN}(x_i)} \alpha_{i,y} \left(\mathbf{f}(x_i, y_i) - \mathbf{f}(x_i, y) \right)$$

$$\Lambda(\alpha) = \min_{\mathbf{w},\xi} \Lambda(\mathbf{w}, \xi, \alpha) = -\frac{1}{2} \left\| \sum_i \sum_{y \in \mathrm{GEN}(x_i)} \alpha_{i,y} \left(\mathbf{f}(x_i, y_i) - \mathbf{f}(x_i, y) \right) \right\|^2 + \sum_i \sum_{y \in \mathrm{GEN}(x_i)} \alpha_{i,y} \ell(y, y_i; x_i)$$

Factored Models

- Recall that features become more expensive as they become less local.
  - Bigram vs. trigram HMM
  - Vanilla PCFG vs. parent-annotated PCFG
- Very common assumptions:

factored features

$$\mathbf{f}(x,y) = \sum_p \mathbf{f}_p(x_p, y_p)$$

$$\mathbf{w} \cdot \mathbf{f}(x,y) = \sum_p \mathbf{w} \cdot \mathbf{f}_p(x_p, y_p)$$

factored loss

$$\ell(y', y; x) = \sum_p [[y'_p \neq y_p]]$$

Factored Models

- Are we giving anything up?

(The question returns in assignment 4!)

$$\mathbf{f}(x,y) = \sum_p \mathbf{f}_p(x_p, y_p)$$

$$\mathbf{w} \cdot \mathbf{f}(x,y) = \sum_p \mathbf{w} \cdot \mathbf{f}_p(x_p, y_p)$$

$$\ell(y', y; x) = \sum_p [[y'_p \neq y_p]]$$

Back to Min-Max

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i \left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \max_{y\in\mathrm{GEN}(x_i)} \left[\mathbf{w}\cdot\mathbf{f}(x_i, y) + \ell(y, y_i; x_i) \right] \right)$$

---

assumptions

$$\mathbf{f}(x,y) = \sum_p \mathbf{f}_p(x_p, y_p)$$

$$\mathbf{w}\cdot\mathbf{f}(x,y) = \sum_p \mathbf{w}\cdot\mathbf{f}_p(x_p, y_p)$$

$$\ell(y', y; x) = \sum_p [[y'_p \neq y_p]]$$

Back to Min-Max

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i \left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \max_{y\in\mathrm{GEN}(x_i)} \left[\mathbf{w}\cdot\mathbf{f}(x_i, y) + \ell(y, y_i; x_i) \right] \right)$$

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i \left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \max_{y\in\mathrm{GEN}(x_i)} \left[\sum_p \mathbf{w}\cdot\mathbf{f}_p(x_{ip}, y_p) + [[y_p \neq y_{ip}]] \right] \right)$$

---

assumptions

$$\mathbf{f}(x, y) = \sum_p \mathbf{f}_p(x_p, y_p)$$

$$\ell(y', y; x) = \sum_p [[y'_p \neq y_p]]$$

$$\mathbf{w}\cdot\mathbf{f}(x, y) = \sum_p \mathbf{w}\cdot\mathbf{f}_p(x_p, y_p)$$

Convert Inner "Max" to a Linear Program

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i\left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \max_{y\in\mathrm{GEN}(x_i)}\left[\mathbf{w}\cdot\mathbf{f}(x_i, y) + \ell(y, y_i; x_i)\right]\right)$$

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i\left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \max_{y\in\mathrm{GEN}(x_i)}\left[\sum_p \mathbf{w}\cdot\mathbf{f}_p(x_{ip}, y_p) + [[y_p \neq y_{ip}]]\right]\right)$$

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i\left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \max_{\mathbf{z}:y(\mathbf{z})\in\mathrm{GEN}(x_i)}\left[\left(\mathbf{F}_i^T\mathbf{w} + \vec{\ell}_i\right)\cdot\mathbf{z}\right]\right)$$

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i\left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \max_{\substack{\mathbf{A}_i\mathbf{z}\leq\mathbf{b}_i,\\ \mathbf{z}\geq\mathbf{0}}}\left[\left(\mathbf{F}_i^T\mathbf{w} + \vec{\ell}_i\right)\cdot\mathbf{z}\right]\right)$$

Notation

$$\mathbf{F}_i = \left[\mathbf{f}_{p_1}\left(x_i, y(\mathbf{z})\right) \quad \mathbf{f}_{p_2}\left(x_i, y(\mathbf{z})\right) \quad \cdots \quad \mathbf{f}_{p_m}\left(x_i, y(\mathbf{z})\right) \right]$$

$$\vec{\ell}_i = \begin{bmatrix} [[y_{ip_1} \neq y(\mathbf{z})_{p_1}]] \\ \vdots \\ [[y_{ip_m} \neq y(\mathbf{z})_{p_m}]] \end{bmatrix}$$

$\mathbf{A}_i, \mathbf{b}_i, \mathbf{z}$ are defined problem-specifically

Duality Returns!

- Primal LP

$$\max_{\mathbf{z}} \quad \mathbf{c}\cdot\mathbf{z}$$
$$\text{s.t.} \quad \mathbf{Az} \le \mathbf{b}$$
$$\mathbf{z} \ge \mathbf{0}$$

- Dual LP

$$\min_{\lambda} \quad \mathbf{b}\cdot\vec{\lambda}$$
$$\text{s.t.} \quad \mathbf{A}^T\vec{\lambda} \ge \mathbf{c}$$
$$\vec{\lambda} \ge \mathbf{0}$$

at optimum: $\mathbf{c}\cdot\mathbf{z} = \mathbf{b}\cdot\vec{\lambda}$

Convert Inner "Max" to a Tractable Linear Program

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i \left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \max_{y\in \mathrm{GEN}(x_i)} \left[\mathbf{w}\cdot\mathbf{f}(x_i, y) + \ell(y, y_i; x_i) \right] \right)$$

…

$$\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i \left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \max_{\substack{\mathbf{A}_i\mathbf{z}\le\mathbf{b}_i,\\ \mathbf{z}\ge\mathbf{0}}} \left[\left(\mathbf{F}_i^T\mathbf{w} + \vec{\ell}_i \right)\cdot\mathbf{z} \right] \right)$$

$$\min_{\mathbf{w},\vec{\lambda}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i \left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \mathbf{b}_i\cdot\vec{\lambda}_i \right)$$

$$s.t.\quad \forall i,\ \mathbf{A}_i^T\vec{\lambda}_i \ge \mathbf{F}_i^T\mathbf{w} + \vec{\ell}_i$$
$$\vec{\lambda}_i \ge 0$$

Taskar et al. (2004): polynomial # of constraints

Take the Dual®

$$\min_{\mathbf{w},\vec{\lambda}} \frac{1}{2}\mathbf{w}\cdot\mathbf{w} - C\sum_i \left(\mathbf{w}\cdot\mathbf{f}(x_i, y_i) - \mathbf{b}_i \cdot \vec{\lambda}_i\right)$$

$$\begin{aligned} s.t. \quad & \forall i, \mathbf{A}_i^T \vec{\lambda}_i \geq \mathbf{F}_i^T \mathbf{w} + \vec{\ell}_i \\ & \vec{\lambda}_i \geq 0 \end{aligned}$$

$$\max_{\vec{\mu}} \vec{\ell}_i \cdot \vec{\mu}_i - \frac{1}{2}\left\| \sum_i C\mathbf{f}(x_i, y_i) - \mathbf{F}_i \vec{\mu}_i \right\|^2$$

$$\begin{aligned} s.t. \quad & \forall i, \mathbf{A}_i^T \vec{\mu}_i \leq C\mathbf{b}_i \\ & \vec{\mu}_i \geq 0 \end{aligned}$$

How many variables?

What I've Skipped

- Training technique: Sequential minimal optimization (SMO; Platt 1998)
  - Breaks big optimization problem into a bunch of smaller ones.
- Exactly how to express labeling, parsing, and other NLP problems as LPs.
  - Homework problem!

A Word About Kernels

- So far, everything has been **linear**.
  - Dot-products of various things with weight and feature vectors.
- You can think of the dot-product **a**·**b** as a similarity measure between **a** and **b**.
  - The greater a dot-product is, the more similar.
- Kernels **generalize** this into more dimensions.
  - Still a dot product, but now between ϕ(**a**) and ϕ(**b**)
  - In higher-dimensional spaces, may be possible to find a separating hyperplane.
- Kernel trick: efficient computation of the new dot product permits non-linear classification.

Some Kernels

polynomial:

$$k(\mathbf{a},\mathbf{b}) = (\mathbf{a}\cdot\mathbf{b}+1)^d = \left(1+\sum_i a_i b_i\right)^d = \mathbf{a}\cdot\mathbf{b} + a_1 b_1(\mathbf{a}\cdot\mathbf{b}) + \cdots + a_n b_n(\mathbf{a}\cdot\mathbf{b}) + \cdots$$

radial basis function:

$$k(\mathbf{a},\mathbf{b}) = \exp\left(-\gamma\|\mathbf{a}-\mathbf{b}\|^2\right)$$

sigmoid:

$$k(\mathbf{a},\mathbf{b}) = \tanh(\kappa\mathbf{a}\cdot\mathbf{b}+c)$$

Kernels

- Not widely used in NLP, but a few specialized kernels have been developed for trees, sequences, etc.

- Central ideas:
  – Maximizing the margin
  – Neat math tricks to make it tractable when ported to NLP problems

Next Time

- MIRA, a useful online training algorithm
- When the features get big, the tough get to **reranking**!