CSC 411 Lecture 17: Support Vector Machine

Ethan Fetaya, James Lucas and Emad Andrews

University of Toronto

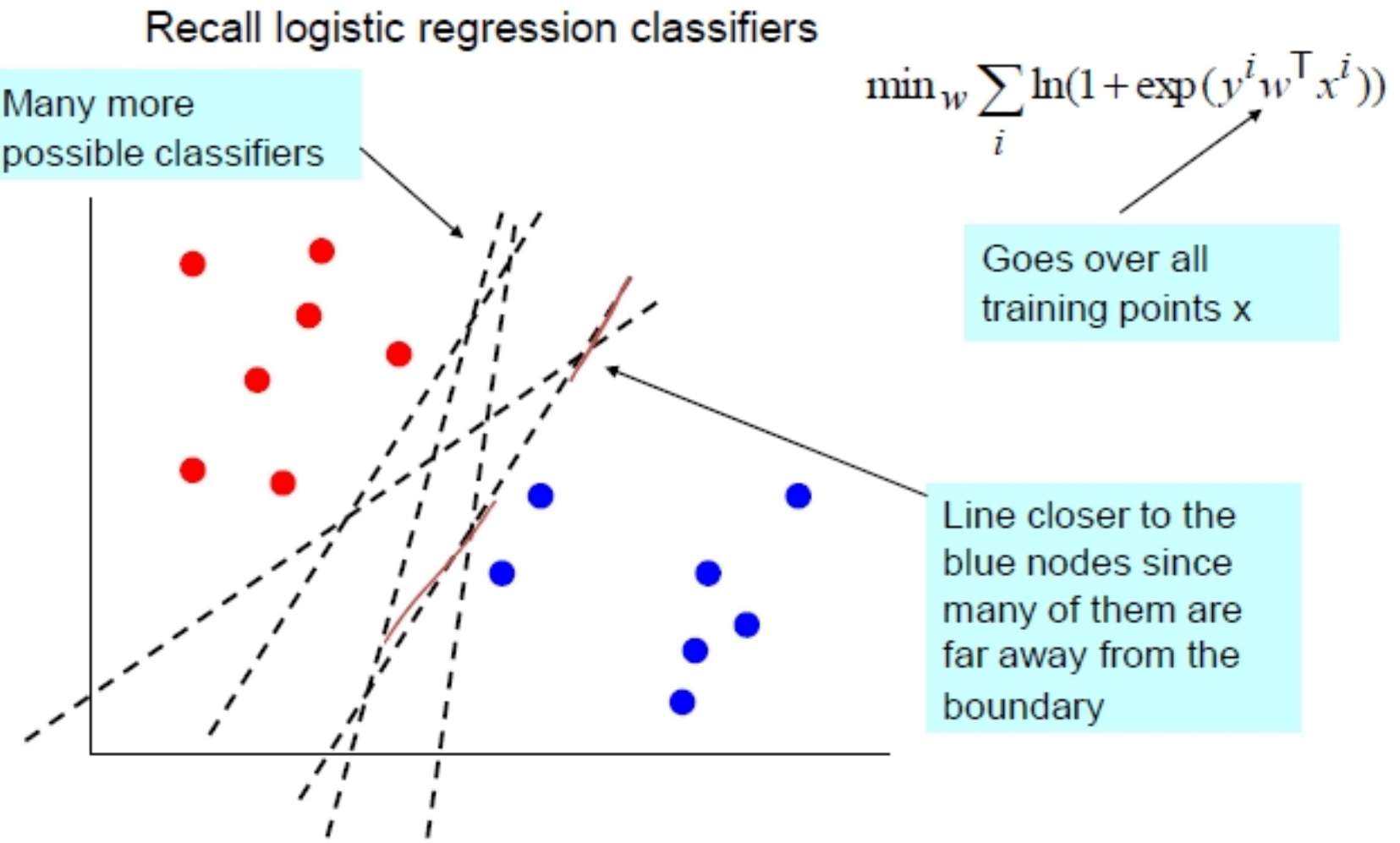
Today

- Max-margin classification
- SVM
  - Hard SVM
  - Duality
  - Soft SVM

Today

- We are back to **supervised** learning
- We are given training data $\{(\mathbf{x}^{(i)}, t^{(i)})\}_{i=1}^{N}$
- We will look at **classification**, so $t^{(i)}$ will represent the class label
- We will focus on **binary** classification (two classes)
- We will consider a **linear** classifier first (next class non-linear decision boundaries)
- Tiny change from before: instead of using $t = 1$ and $t = 0$ for positive and negative class, we will use $t = 1$ for the positive and $t = -1$ for the negative class

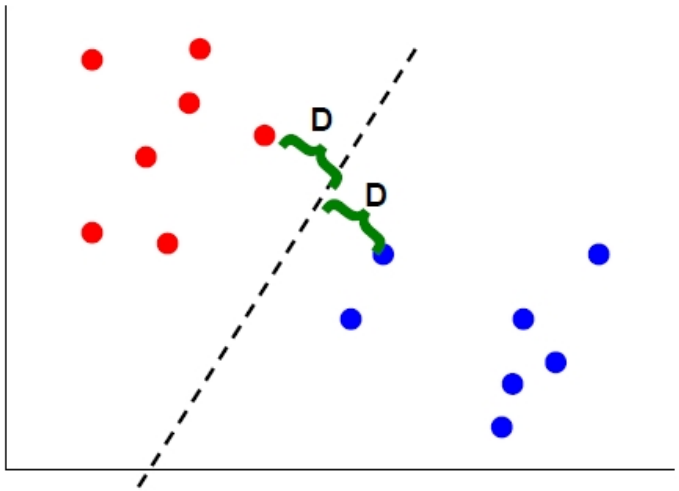
Logistic Regression



$$y = \begin{cases} 1 & \text{if } (\mathbf{w}^T\mathbf{x} + b) \geq 0 \\ -1 & \text{if } (\mathbf{w}^T\mathbf{x} + b) < 0 \end{cases}$$

Max Margin Classification

- If the data is linearly separable, which separating hyperplane do you pick?
- Aim: learn a boundary that leads to the largest margin (buffer) from points on both sides



- Why: intuition; theoretical support; and works well in practice
- Subset of vectors that support (determine boundary) are called the support vectors

Hard SVM

- Assume (for now) the data is linearly separable.
  - There exists $\mathbf{w}$ and b such that $\forall i: \; sign(\mathbf{w}^T\mathbf{x}^{(i)} + b) = t^{(i)}$
  - Equivalently: $\forall i: \; t^{(i)} \cdot (\mathbf{w}^T\mathbf{x}^{(i)} + b) > 0$
- We want to maximize the margin, how do we formulate it mathematically?
- What is the distance of $\mathbf{x}^{(i)}$ from the hyperplane $\mathbf{w}^T\mathbf{x} + b = 0$? $\frac{|\mathbf{w}^T\mathbf{x}^{(i)}+b|}{||\mathbf{w}||}$ (show!). Equivalently (assuming $\mathbf{w}$ separates) $\frac{t^{(i)}\cdot(\mathbf{w}^T\mathbf{x}^{(i)}+b)}{||\mathbf{w}||}$
- The margin for $\mathbf{w}$ and b: $\min_i \frac{t^{(i)}\cdot(\mathbf{w}^T\mathbf{x}^{(i)}+b)}{||\mathbf{w}||}$ - distance to closest data point.

Hard SVM Objective

Hard SVM objective V1:

$$\mathbf{w}, b = \arg\max_{\mathbf{w},b} \left[\min_i \frac{t^{(i)} \cdot (\mathbf{w}^T \mathbf{x}^{(i)} + b)}{||\mathbf{w}||} \right]$$
$$s.t. \quad \forall i : t^{(i)} \cdot (\mathbf{w}^T \mathbf{x}^{(i)} + b) > 0$$

- The straightforward way to write the SVM objective.
- Note: If the data is linearly separable you don't really need the $t^{(i)} \cdot (\mathbf{w}^T \mathbf{x}^{(i)} + b) > 0$ constraints (why?).
- Writing it differently will lead to easier optimization.
- The objective is scale invariant - we can normalize the "margin" to one.

Hard SVM objective V2:

$$\mathbf{w}, b = \arg\max_{\mathbf{w},b} \left[\frac{1}{||\mathbf{w}||} \right] = \arg\min_{\mathbf{w},b} \frac{1}{2} ||\mathbf{w}||^2$$
$$s.t. \quad \min_i t^{(i)} \cdot (\mathbf{w}^T \mathbf{x}^{(i)} + b) = 1$$

Hard SVM Objective

- Further simplification - "margin" is at least one.

Hard SVM (primal) objective:

$$\mathbf{w}, b = \arg\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2$$
$$s.t. \quad \forall i: \; t^{(i)} \cdot (\mathbf{w}^T \mathbf{x}^{(i)} + b) \geq 1$$

- Why is this equivalent?
  - If the "margin" isn't exactly one we can scale $\mathbf{w}$ down and get a smaller norm.
- Convex quadratic programming problem.

SVM Dual Form

- Convert the constrained minimization to an unconstrained optimization problem: represent constraints as penalty terms:

$$\min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2 + \text{penalty\_term}$$

- For data $\{(\mathbf{x}^{(i)}, t^{(i)})\}_{i=1}^N$, use the following penalty

$$\max_{\alpha_i \geq 0} \quad \alpha_i[1 - (\mathbf{w}^T\mathbf{x}^{(i)} + b)t^{(i)}] = \begin{cases} 0 & \text{if} \quad (\mathbf{w}^T\mathbf{x}^{(i)} + b)t^{(i)} \geq 1 \\ \infty & \text{otherwise} \end{cases}$$

- Rewrite the minimization problem

$$\min_{\mathbf{w},b}\{\frac{1}{2}||\mathbf{w}||^2 + \sum_{i=1}^{N} \max_{\alpha_i \geq 0} \alpha_i[1 - (\mathbf{w}^T\mathbf{x}^{(i)} + b)t^{(i)}]\}$$

where α_i are the Lagrange multipliers

$$= \min_{\mathbf{w},b} \max_{\alpha_i \geq 0}\{\frac{1}{2}||\mathbf{w}||^2 + \sum_{i=1}^{N} \alpha_i[1 - (\mathbf{w}^T\mathbf{x}^{(i)} + b)t^{(i)}]\}$$

SVM Dual Form

- Let:

$$J(\mathbf{w}, b; \alpha) = \frac{1}{2}||\mathbf{w}||^2 + \sum_{i=1}^{N} \alpha_i [1 - (\mathbf{w}^T \mathbf{x}^{(i)} + b) t^{(i)}]$$

- Swap the "max" and "min": This is a lower bound

$$\max_{\alpha_i \geq 0} \min_{\mathbf{w}, b} J(\mathbf{w}, b; \alpha) \leq \min_{\mathbf{w}, b} \max_{\alpha_i \geq 0} J(\mathbf{w}, b; \alpha)$$

- Equality holds in certain conditions
  - Called "strong duality"

KKT conditions

- Solving:

$$\max_{\alpha_i \geq 0} \min_{\mathbf{w},b} J(\mathbf{w}, b; \alpha) = \max_{\alpha_i \geq 0} \min_{\mathbf{w},b} \frac{1}{2}||\mathbf{w}||^2 + \sum_{i=1}^{N} \alpha_i [1 - (\mathbf{w}^T \mathbf{x}^{(i)} + b) t^{(i)}]$$

- Convex analysis theory: The solution satisfies all constraints and the following KKT conditions:

$$\begin{aligned}
\frac{\partial J(\mathbf{w}, b; \alpha)}{\partial \mathbf{w}} &= \mathbf{w} - \sum_{i=1}^{N} \alpha_i \mathbf{x}^{(i)} t^{(i)} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{N} \alpha_i t^{(i)} \mathbf{x}^{(i)} \\
\frac{\partial J(\mathbf{w}, b; \alpha)}{\partial b} &= -\sum_{i=1}^{N} \alpha_i t^{(i)} = 0 \\
\alpha_i [1 - (\mathbf{w}^T \mathbf{x}^{(i)} + b) t^{(i)}] &= 0 \quad (\textit{Complementary slackness})
\end{aligned}$$

- Then substitute back to get final dual objective:

$$L = \max_{\alpha_i \geq 0} \{ \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} t^{(i)} t^{(j)} \alpha_i \alpha_j (\mathbf{x}^{(i)^T} \cdot \mathbf{x}^{(j)}) \}$$

- Set $b^* = -\frac{1}{2} \left(\max_{i:t^{(i)}=-1} \mathbf{w}^{*T} \mathbf{x}^{(i)} + \max_{i:t^{(i)}=1} \mathbf{w}^{*T} \mathbf{x}^{(i)} \right)$

Support Vectors

- From KKT conditions we have

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i t^{(i)} \mathbf{x}^{(i)}, \qquad \forall i : \alpha_i [1 - (\mathbf{w}^T \mathbf{x}^{(i)} + b) t^{(i)}] = 0$$

- If point $\mathbf{x}^{(i)}$ isn't on the boundary, then $\alpha_i = 0$!
- The optimal solution is a sum of only the support vectors
- Can show that if there are relatively few support vectors then SVM generalizes well.

Summary of Linear SVM

- Binary and linear separable classification
- Linear classifier with maximal margin
- Showed two objectives, primal and dual
  - Both are convex quadratic optimization problems.
  - Primal optimizes d variables, dual optimizes N variables.
- The weights are

$$\mathbf{w} = \sum_{i=1}^{N} \alpha_i t^{(i)} \mathbf{x}^{(i)}$$

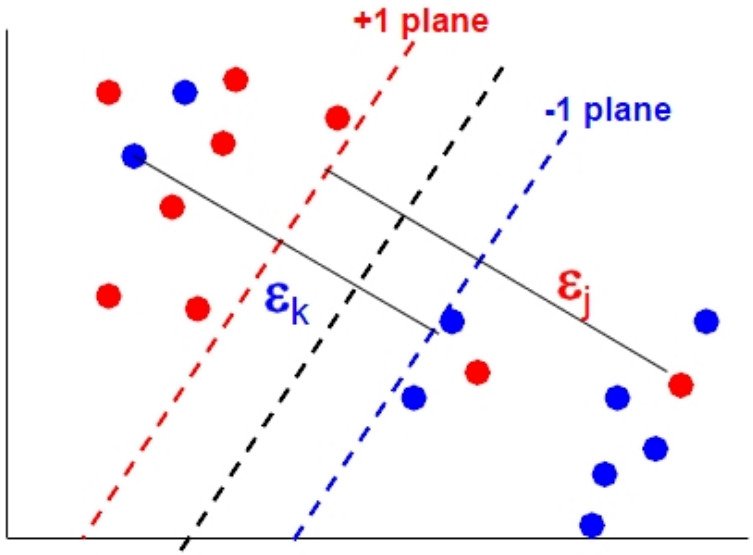
- Only a small subset of α_i's will be nonzero, and the corresponding $\mathbf{x}^{(i)}$'s are the support vectors $\mathbf{S}$
- Prediction on a new example:

$$y = \text{sign}[b + \mathbf{x}^T \mathbf{w}] = \text{sign}[b + \mathbf{x}^T \cdot (\sum_{i \in \mathbf{S}} \alpha_i t^{(i)} \mathbf{x}^{(i)})]$$

Non-separable data

- So far we assume the data is separable, what do we do if its not?
- the constraints $t^{(i)} \cdot (\mathbf{w}^T \mathbf{x}^{(i)} + b) \geq 1$ cannot be satisfied for all data points.
- Solution: Allow the constraints to be violated, but penalize for it.
- Introduce slack variables $\xi_i \geq 0$ and make the new constraints $t^{(i)} \cdot (\mathbf{w}^T \mathbf{x}^{(i)} + b) \geq 1 - \xi_i$
- What is the minimal slack needed?
  - $\xi_i = \max\{1 - t^{(i)} \cdot (\mathbf{w}^T \mathbf{x}^{(i)} + b), 0\}$
- Penalty is $C \cdot \sum_i \xi_i$ for some hyperparameter C.
- Some variations use $C \cdot \sum_i \xi_i^2$ penalty

Soft-SVM



- The new objective:

$$\min_{\mathbf{w},b,\xi} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N}\xi_i$$

$$\text{s.t} \quad \xi_i \geq 0; \quad \forall i \quad t^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1 - \xi_i$$

- Example lies on wrong side of hyperplane $\xi_i > 1$
- Therefore $\sum_i \xi_i$ upper bounds the number of training errors
- C trades off training error vs model complexity
- This is known as the soft-margin extension
- Can show a dual form is the same except we have an additional $\alpha_i \leq C$ constraint (and b^* computation).

Hinge Loss

- Our objective:

$$\min_{\mathbf{w},b,\xi} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N}\xi_i$$
$$\text{s.t} \quad \xi_i \geq 0; \quad \forall i \quad t^{(i)}(\mathbf{w}^T\mathbf{x}^{(i)} + b) \geq 1 - \xi_i$$

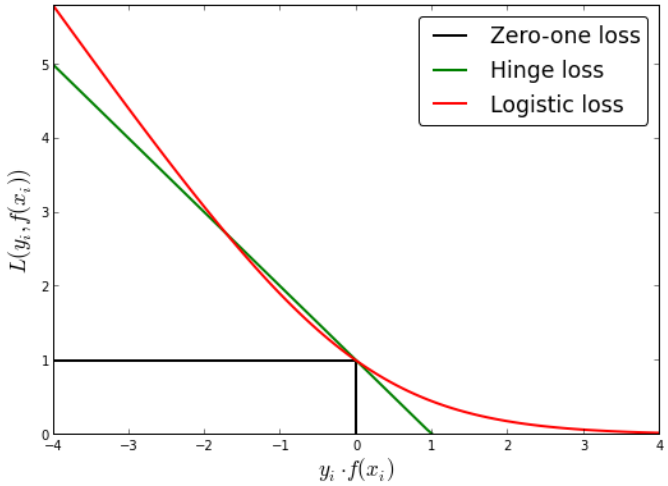
- We can plug in the optimal ξ_i and get the equivalent objective

$$\min_{\mathbf{w},b,} \frac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{N}\ell_{hinge}(\mathbf{w}^T\mathbf{x}^{(i)} + b, t^{(i)})$$

- We define the hinge loss as

$$\ell_{hinge}(\mathbf{w}^T\mathbf{x}^{(i)} + b, t^{(i)}) = \max\{1 - t^{(i)} \cdot (\mathbf{w}^T\mathbf{x}^{(i)} + b), 0\}$$

Hinge Loss



- $\ell_{hinge}(z, t) = \max\{1 - z \cdot t, 0\}$
- $\ell_{logistic}(z, t) = \ln(1 + \exp(-z \cdot t))$

- Another surrogate loss for 0-1 loss.
- Convex.
- Not smooth at the kink but ok (can use subgradients, not covered in this course)
- Hinge loss can be used with other learning algorithms like neural networks.

Example: Pedestrian Detection

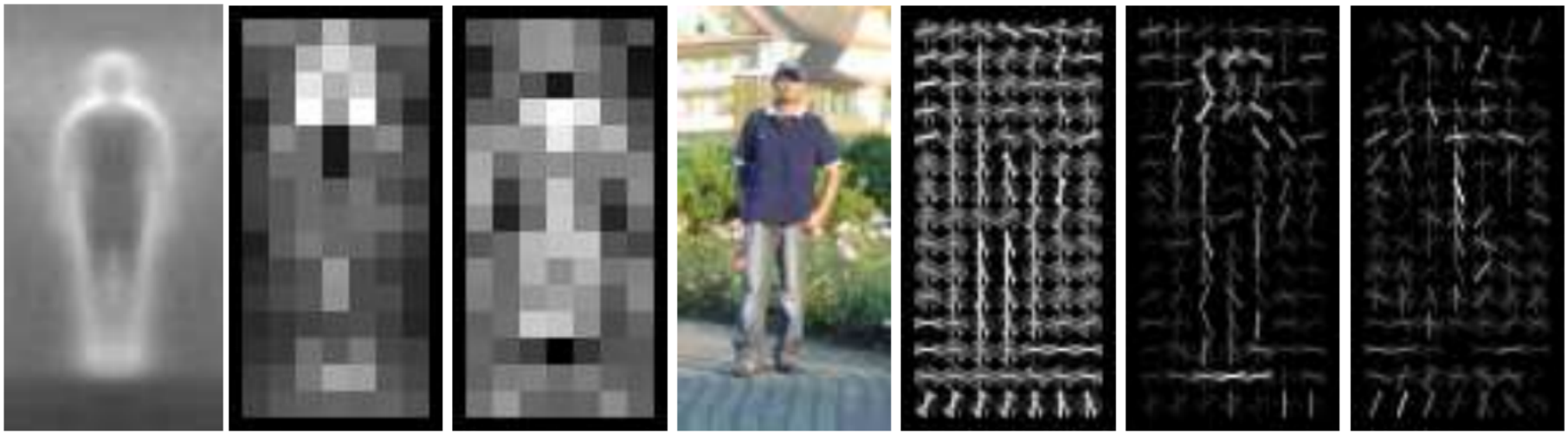


Figure 6. Our HOG detectors cue mainly on silhouette contours (especially the head, shoulders and feet). The most active blocks are centred on the image background just *outside* the contour. (a) The average gradient image over the training examples. (b) Each "pixel" shows the maximum positive SVM weight in the block centred on the pixel. (c) Likewise for the negative SVM weights. (d) A test image. (e) It's computed R-HOG descriptor. (f,g) The R-HOG descriptor weighted by respectively the positive and the negative SVM weights.

- Big breakthrough in computer vision (2006)
- Linear SVM + hand crafted features (HOG)

[Image credit: "Histograms of Oriented Gradients for Human Detection"]

SVM Theoretical Guarantees *

- Assume the data is separated by a margin γ and that $||\mathbf{x}|| \leq 1$
- Can show that with probability at least $1 - \delta$ the 0-1 loss of (hard) SVM will be bounded by

$$\mathcal{O}\left(\sqrt{\frac{1/\gamma^2}{N}} + \sqrt{\frac{\log(1/\delta)}{N}}\right)$$

- Main observation: This does not depend on the dimension!
- Can show similar results for "soft" SVM.
- Very important for kernels (soon)

Recap

- Maximum margin classifiers.
- Convex quadratic optimization.
- Primal and dual objective, which one to use depends on dimension and data size.
  - Open source packages like sklearn support both.
- If data isn't separable - use slack variables to penalize constraint violations.
- Another perspective - hinge loss with l_2 regularization.
- Important note: If you use a off-the-shelf quadratic solver it will be very slow, special SVM solvers like SMO are much better.
- Can use SGD, see Pegasos