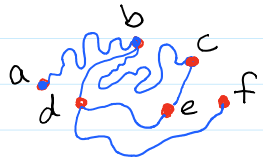
Tutorial

February 9, 2016 12:06 PM

Alireza Shafaei

- Duals
- Fenchel Dual
- Convex Conjugates
- Generative Classifiers

Duals

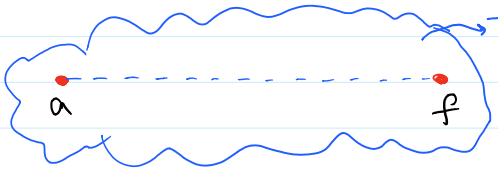


A tangible example!

Consider a graph, in which, instead of putting weights on the edges, we draw the edges with proportional lengths to their weights.

(P) what's the length of the shortest path between a and f ?

let's look at another problem (D) How far can I move a and f away before breaking the graph?



The maximum distance between a and f is also the shortest distance between a and f !!

Sometimes we can define problems that look entirely different to our original problem, but they have intimate connections that we can take advantage of!

If You've taken CPSC 420, the min-cut and the max-flow problems are another pair of duals!

⊕ Are the dual problems unique? (I don't know the answer)

In optimization we also have duals. One name that you'd often hear is the Lagrange Dual

One special case of the Lagrange Duals is referred to as the Fenchel Dual which we will be studying today.

$$\arg\min_{w \in \mathbb{R}^d} p(w) = f(Xw) + g(w) \quad \Rightarrow \text{The primal problem}$$

$$X \in \mathbb{R}^{n \times d}$$

$$\arg\max_{z \in \mathbb{R}^n} D(z) = -f^*(-z) - g^*(X^T z) \quad \Rightarrow \text{The dual problem}$$

f^* & g^* functions are the convex conjugate functions of f and g, we will talk about them later!

⊕ if $n < d$, the dual problem requires fewer parameters to optimize

⊕ Strong Duality: $P(w^*) = D(z^*)$

↳ if you optimize p & D separately, the optimal value of the functions is going to be equal!

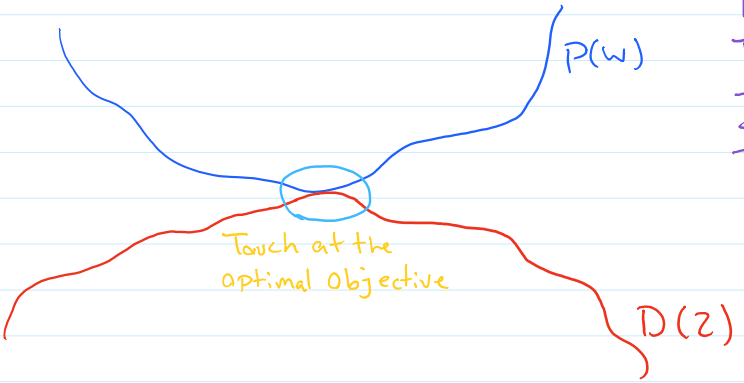
⊕ Weak Duality: $\forall w, z \quad D(z) \leq P(w)$

↳ The dual objective is always below the primal

You can imagine the relationship to be something like this

THIS IS NOT THE REAL PICTURE

You can imagine the relationship to be something like this

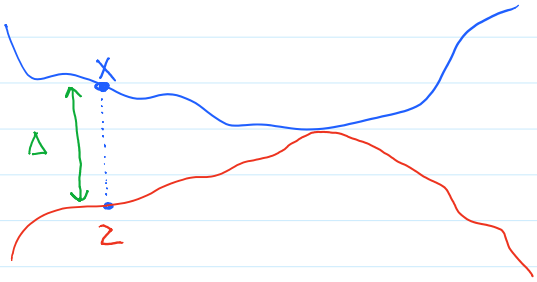


THIS IS NOT THE REAL PICTURE Just AN EXAMPLE To Emphasize The relationship.

When do you terminate your optimization? Ideally you want to know how close to the optimal value you are.

- ☆ When the objective value doesn't change much between the iterations?
  - ↳ what if the almost flat surface lasts way too long?
- ☆ The gradient norm is below some threshold?
  - ↳ It doesn't tell you how far from the optimal point x you are

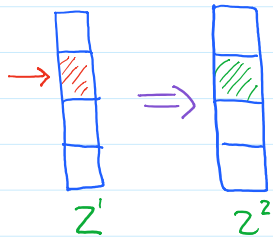
When you have a nice dual, every now and then you can evaluate $D(z)$ with the corresponding z to your current x, and the difference gives you an idea of how far from the optimal objective you are!



and you can terminate when Δ is below some threshold!

☆ If the primal is strongly convex, the dual is smooth! and maybe the dual can be solved faster than the primal!

☆ It often leads to problems for which coordinate methods are effective!



Select a coordinate — update with the optimal value keeping everything else fixed.

if $D(z') < D(z^2)$ (in maximization) is guaranteed, you can happily apply coordinate ascent methods this is the case with your dual SVM assignment.

☆ for more properties check the course lecture slides!

Convex Conjugates

if we have $f(x)$, the convex conjugate is defined as

$$f^*(y) = \sup_{x \in D} \{y^T x - f(x)\}$$

→ where the result is some finite value (will see an example)

Intuition? I've seen stories about what this formulation means, but none of them has motivated me to define such a weird looking function. So I suggest we should just move on!

f doesn't have to be convex, but f^* always is convex!

Let's look at two problems

$f(x) = \frac{1}{2}\|x-b\|^2$

$$f^*(y) = \sup_{x \in D} \left\{ y^T x - \frac{1}{2}(x-b)^T(x-b) \right\}$$

Quadratic function of x → We can solve it exactly!

$\nabla: y - x + b = 0 \Rightarrow x = y + b$

$$\Rightarrow f^*(y) = y^T(y+b) - \frac{1}{2} y^T y = \frac{1}{2} y^T y + y^T b \quad (1)$$

$f(x) = \lambda \|x\|_1$

$$f^*(y) = \sup_{x \in D} \left\{ y^T x - \lambda \sum |x_i| \right\} = \sup_{x \in D} \left\{ \sum y_i x_i - \lambda |x_i| \right\} = \sum \sup \left\{ y_i x_i - \lambda |x_i| \right\}$$

must be finite

for one x_i, we have (± 1 or -1)

$$\sup \{ y_i x_i - \lambda \, \text{sign}(x_i) x_i \} = \sup \{ (y_i - \lambda \, \text{sign}(x_i)) x_i \}$$

Case 1: $x_i > 0 \rightarrow (y_i - \lambda) x_i$

if $(y_i - \lambda)$ is positive, we can take $x_i \to \infty$ to maximize
Since the value must be finite, when $y_i - \lambda$ is positive
the Convex Conjugate is not happy!
if $(y_i - \lambda) < 0$, then the best we can do is to set $x_i = 0$

$y_i \leq \lambda \rightarrow 0$
o.w $\rightarrow \infty$

Case 2: $x_i < 0 \rightarrow (y_i + \lambda) x_i$

Like before, if $(y_i + \lambda)$ is negative, we can take $x_i \to -\infty$
therefore we can define it when $y_i + \lambda > 0$ at which the best we can do is 0

$y_i > -\lambda \rightarrow 0$

↳ $y_i > -\lambda \rightarrow 0$
o.w $\rightarrow \infty$

$$\Rightarrow f^*(y) = \begin{cases} 0 & \forall y_i \; -\lambda \le y_i \le \lambda \\ \infty & \text{o.w} \end{cases}$$

↳ must be true for all y_i, because if one of them leads to ∞, the total sum (see above) will also be ∞.

alternatively you can say

$$f^*(y) = \begin{cases} 0 & \|y\|_\infty \le \lambda \\ \infty & \text{o.w} \end{cases} \quad (2)$$

now let's derive a fenchel Dual for this problem

$$\arg\min_w \; P(w) = \underbrace{\frac{1}{2}\|Xw - y\|^2}_{f} + \lambda \underbrace{\|w\|_1}_{g} \qquad \text{(Lasso)}$$

$f(A) : \frac{1}{2}\|A - y\|^2$ (1)

$g(A) : \lambda \|A\|_1$ (2)

$$\Rightarrow \arg\max_z \; D(z) = -\underbrace{\left(\tfrac{1}{2}(-z)^T(-z) + y^T(-z)\right)}_{f^*(-z)} + g^*(X^T z)$$

$$\rightarrow -\frac{1}{2}z^T z + y^T z + g^*(X^T z)$$

g^* is here to ensure $\|X^T z\|_\infty \le \lambda$ we might as well say

$$\arg\max_z \; y^T z - \frac{1}{2} z^T z$$

$$\text{s.t} \quad \|X^T z\|_\infty \le \lambda$$

Generative Classifiers:

let's define the following procedure for generating data for a classification problem!

1- Select a class c from $\{1, \ldots, K\}$ $\Rightarrow P(C)$ (1)

2- Select a random sample X from the class c. $\Rightarrow P(X \mid C = c)$ (2)

3- Print the pair (X, c)

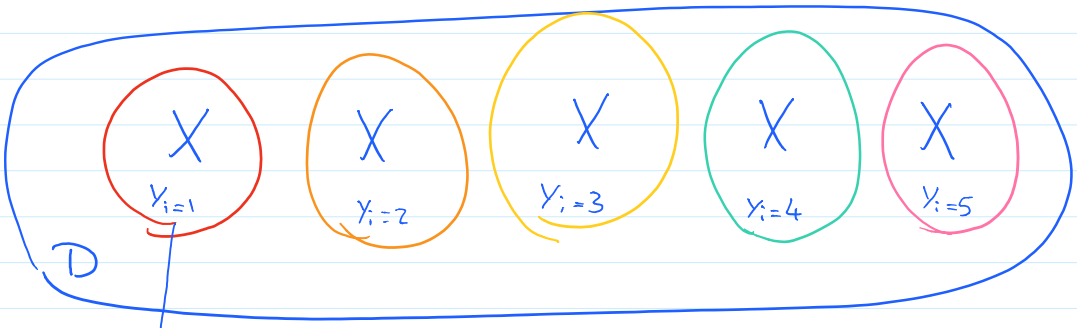
I do this n times and generate a dataset $D = \{(X_i, Y_i)\}_{i=1}^{n}$ where Y_i is the class label.

I give you a dataset D, and ask you to find the underlying distributions of (1) and (2)

(1) is simply the probability of picking class c, let's call it θ_c Obviously $\sum \theta_i = 1$

if we just look at the empirical distribution of the labels in D, the MLE estimate of $\theta_i = \frac{n_i}{N}$.

(2) for modelling the X's we can only look at the subset of D with a particular class label!



↳ It can be a parametric distribution (like a Gaussian) or a non-parametric one!

If we assume each one is a Gaussian, the MLE estimate is simply the empirical μ and Σ.

Okay! now that you have an estimate of the parameters of my process, you can generate a dataset of yourself by simply following the above procedure. (You have a generative model)
Of course the quality of your samples depends on how accurate your assumptions and estimates are!

Now, if I give you a new X, you can classify it as follows

$$P(Y=c \mid X, \theta) = \frac{P(X \mid Y=c, \theta) P(Y=c \mid \theta)}{P(X \mid \theta)}$$

→ Bayes Rule

θ is your parameter!

Part (2) of the above process the probability of picking a particular X from class c!

Just a normalizer does not depend on class. What's the probability of encountering X.

Part (1) of the above process the probability of picking class c

You can evaluate $P(X \mid Y=c, \theta) P(Y=c \mid \theta)$ for all possible classes and predict the class as the one with the highest value!

This is just a simple example! You can have more complicated procedures and deeper hierarchies!