Soft Margin Support Vector Machine

之前讲的这些方法都是Hard-Margin SVM，即必须将所有的样本都分类正确才行。
这往往需要更多更复杂的特征转换，甚至造成过拟合。

这次的Soft-Margin SVM，目的是让分类错误的点越少越好，而不是必须将所有点分类正确，也就是允许有noise存在。
这种做法很大程度上不会使模型过于复杂，不会造成过拟合，而且分类效果是令人满意的。

1.Motivation and Primal Problem

SVM同样可能会造成overfit。
原因有两个
一个是由于我们的SVM模型（即kernel）过于复杂，转换的维度太多，过于powerful了
另外一个是由于我们坚持要将所有的样本都分类正确，即不允许错误存在，造成模型过于复杂。

可以借鉴pocket（pocket的思想不是将所有点完全分开，而是找到一条分类线能让分类错误的点最少）

pocket

$$\min_{b,\mathbf{w}} \sum_{n=1}^{N} [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

hard-margin SVM

$$\min_{b,\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w}$$
$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 \text{ for all } n$$

为了引入允许犯错误的点，我们将Hard-Margin SVM的目标和条件做一些结合和修正，转换为如下形式：

$$\text{combination:} \quad \min_{b,\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$
$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 \text{ for correct } n$$
$$y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq -\infty \text{ for incorrect } n$$

对于分类正确的点，仍需满足$y_n(w^Tz_n+b)\geq 1$
对于noise点，满足$y_n(w^Tz_n+b)\geq -\infty$
修正后的目标除了$\frac{1}{2}w^Tw$项，还添加了$y_n \neq sign(w^Tz_n+b)$，即noise点的个数
参数C的引入是为了权衡目标第一项和第二项的关系，即权衡large margin和noise tolerance的关系。

两个条件合并，得到：

$$\min_{b,\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$
$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \infty \cdot [\![y_n \neq \text{sign}(\mathbf{w}^T\mathbf{z}_n + b)]\!]$$

这个式子存在两个不足的地方。
首先，最小化目标中第二项是非线性的，不满足QP的条件，所以无法使用dual或者kernel SVM来计算。
其次，对于犯错误的点，有的离边界很近，即error小，而有的离边界很远，error很大，上式的条件和目标没有区分small error和large error。这种分类效果是不完美的。

为了改进不足，作如下修正：

- record '**margin violation**' by ξ_n—**linear constraints**
- penalize with **margin violation** instead of **error count** —**quadratic objective**

$$\text{soft-margin SVM:} \quad \min_{b,\mathbf{w},\xi} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C \cdot \sum_{n=1}^{N} \xi_n$$
$$\text{s.t.} \quad y_n(\mathbf{w}^T\mathbf{z}_n + b) \geq 1 - \xi_n \text{ and } \xi_n \geq 0 \text{ for all } n$$

修正后的表达式中，我们引入了新的参数ξ_n来表示每个点犯错误的程度值，$\xi_n \geq 0$。
通过使用error值的大小代替是否有error，让问题变得易于求解，满足QP形式要求。

这种方法类似于我们在机器学习基石笔记中介绍的0/1 error和squared error。
这种soft-margin SVM引入新的参数ξ。

现在，最终的Soft-Margin SVM的目标为：

$$min(b,w,\xi)\ \frac{1}{2}w^Tw+C\cdot\sum_{n=1}^{N}\xi_n$$

条件是：

$$y_n(w^Tz_n+b)\geq 1-\xi_n$$

$$\xi_n\geq 0$$

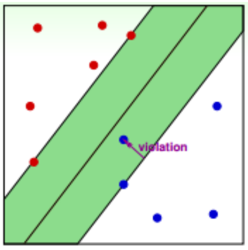
其中，ξ_n表示每个点犯错误的程度，$\xi_n=0$，表示没有错误，ξ_n越大，表示错误越大，即点距离边界（负的）越大。
参数C表示尽可能选择宽边界和尽可能不要犯错两者之间的权衡，因为边界宽了，往往犯错误的点会增加。
large C表示希望得到更少的分类错误，即不惜选择窄边界也要尽可能把更多点正确分类
small C表示希望得到更宽的边界，即不惜增加错误点个数也要选择更宽的分类边界。

与之对应的QP问题中，由于新的参数ξ_n的引入，总共参数个数为$\hat{d}+1+N$，限制条件添加了$\xi_n\geq 0$，则总条件个数为2N。

- record 'margin violation' by ξ_n
- penalize with margin violation

$$\min_{b,\mathbf{w},\xi}\quad \frac{1}{2}\mathbf{w}^T\mathbf{w}+C\cdot\sum_{n=1}^{N}\xi_n$$

$$\text{s.t.}\quad y_n(\mathbf{w}^T\mathbf{z}_n+b)\geq 1-\xi_n \text{ and } \xi_n\geq 0 \text{ for all } n$$



- parameter C: trade-off of large margin & margin violation
  - large C: want less margin violation
  - small C: want large margin
- QP of $\tilde{d}+1+N$ variables, $2N$ constraints

2.Dual Problem

Soft-Margin SVM的原始形式：

$$\text{primal:}\quad \min_{b,\mathbf{w},\xi}\quad \frac{1}{2}\mathbf{w}^T\mathbf{w}+C\cdot\sum_{n=1}^{N}\xi_n$$

$$\text{s.t.}\quad y_n(\mathbf{w}^T\mathbf{z}_n+b)\geq 1-\xi_n \text{ and } \xi_n\geq 0 \text{ for all } n$$

构造一个拉格朗日函数。

Lagrange function with Lagrange multipliers α_n and β_n

$$\mathcal{L}(b,\mathbf{w},\xi,\alpha,\beta) = \frac{1}{2}\mathbf{w}^T\mathbf{w}+C\cdot\sum_{n=1}^{N}\xi_n + \sum_{n=1}^{N}\alpha_n\cdot\left(1-\xi_n-y_n(\mathbf{w}^T\mathbf{z}_n+b)\right)+\sum_{n=1}^{N}\beta_n\cdot(-\xi_n)$$

利用Lagrange dual problem，将Soft-Margin SVM问题转换为如下形式：

$$\max_{\alpha_n\geq 0,\ \beta_n\geq 0}\left(\min_{b,\mathbf{w},\xi}\quad \frac{1}{2}\mathbf{w}^T\mathbf{w}+C\cdot\sum_{n=1}^{N}\xi_n + \sum_{n=1}^{N}\alpha_n\cdot\left(1-\xi_n-y_n(\mathbf{w}^T\mathbf{z}_n+b)\right)+\sum_{n=1}^{N}\beta_n\cdot(-\xi_n)\right)$$

根据之前介绍的KKT条件，我们对上式进行简化。上式括号里面的是对拉格朗日函数$L(b,w,\xi,\alpha,\beta)$计算最小值。那么根据梯度下降算法思想：最小值位置满足梯度为零。

我们先对ξ_n做偏微分：

$$\frac{\partial L}{\partial \xi_n}=0=C-\alpha_n-\beta_n$$

根据上式，得到$\beta_n=C-\alpha_n$，因为有$\beta_n\geq 0$，所以限制$0\leq\alpha_n\leq C$
将$\beta_n=C-\alpha_n$代入到dual形式中并化简，我们发现β_n和ξ_n都被消去了：

$$\max_{0\leq\alpha_n\leq C,\ \beta_n=C-\alpha_n}\left(\min_{b,\mathbf{w}}\quad \frac{1}{2}\mathbf{w}^T\mathbf{w}+\sum_{n=1}^{N}\alpha_n(1-y_n(\mathbf{w}^T\mathbf{z}_n+b))\right)$$

分别令拉格朗日函数L对b和w的偏导数为零，分别得到：

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

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

经过化简和推导，最终标准的Soft-Margin SVM的Dual形式如下图所示：

$$\begin{array}{ll}
\min\limits_{\alpha} & \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{z}_n^T\mathbf{z}_m-\sum_{n=1}^{N}\alpha_n \\
\text{subject to} & \sum_{n=1}^{N}y_n\alpha_n=0; \\
 & 0\leq\alpha_n\leq C, \text{for } n=1,2,\ldots,N; \\
\text{implicitly} & \mathbf{w}=\sum_{n=1}^{N}\alpha_n y_n\mathbf{z}_n; \\
 & \beta_n=C-\alpha_n, \text{for } n=1,2,\ldots,N
\end{array}$$

—only difference to hard-margin: upper bound on α_n

Soft-Margin SVM Dual与Hard-Margin SVM Dual基本一致，只有一些条件不同。
Hard-Margin SVM Dual中$\alpha_n\geq 0$，而Soft-Margin SVM Dual中$0\leq\alpha_n\leq C$，且新的拉格朗日因子$\beta_n=C-\alpha_n$。
在QP问题中，Soft-Margin SVM Dual的参数α_n同样是N个，但是，条件由Hard-Margin SVM Dual中的N+1个变成2N+1个，这是因为多了N个α_n的上界条件。

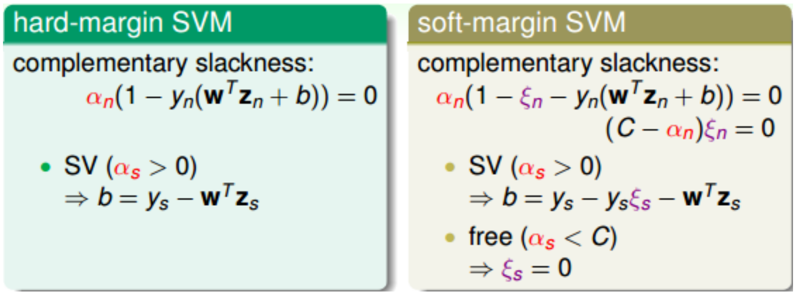
3.Messages behind Soft-Margin SVM

Soft-Margin SVM Dual计算α_n的方法过程：

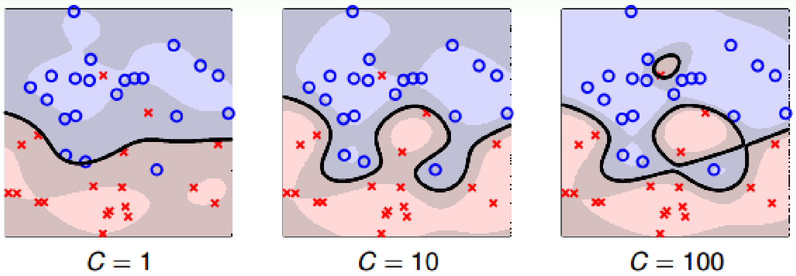
Kernel Soft-Margin SVM Algorithm

1. $q_{n,m}=y_n y_m K(\mathbf{x}_n,\mathbf{x}_m)$; $\mathbf{p}=-\mathbf{1}_N$; $(\mathrm{A},\mathbf{c})$ for equ./lower-bound/**upper-bound** constraints
2. $\boldsymbol{\alpha}\leftarrow \mathrm{QP}(\mathrm{Q_D},\mathbf{p},\mathrm{A},\mathbf{c})$
3. $b\leftarrow ?$
4. return SVs and their α_n as well as b such that for new $\mathbf{x}$,
$$g_{\text{SVM}}(\mathbf{x})=\text{sign}\left(\sum_{\text{SV indices } n}\alpha_n y_n K(\mathbf{x}_n,\mathbf{x})+b\right)$$

在Hard-Margin SVM Dual中，有complementary slackness条件：$\alpha_n(1-y_n(w^Tz_n+b))=0$，找到SV，即$\alpha_s>0$的点，计算得到$b=y_s-w^Tz_s$。

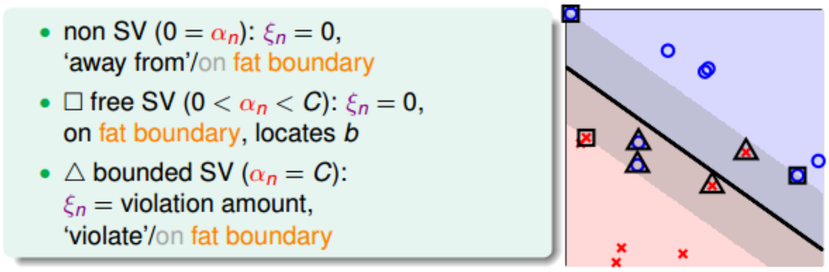


对于Soft-Margin Gaussian SVM，C分别取1，10，100时，相应的margin如下图所示：



C越小，越倾向于得到粗的margin，增加分类错误的点；C越大，越倾向于得到高的分类正确率。
我们发现，当C值很大的时候，虽然分类正确率提高，但很可能把noise也进行了处理，从而可能造成过拟合。
也就是说Soft-Margin Gaussian SVM同样可能会出现过拟合现象，所以参数(γ, C)的选择非常重要。

在Soft-Margin SVM Dual中，根据α_n的取值，就可以推断数据点在空间的分布情况：



$$\alpha_n(1 - \xi_n - y_n(w^T z_n + b)) = 0$$

$$\beta_n \xi_n = (C - \alpha_n)\xi = 0$$

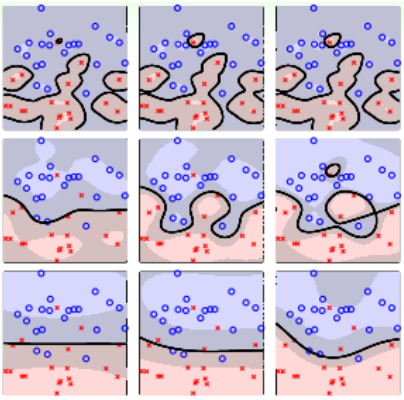
若$\alpha_n = 0$，得$\xi_n = 0$。$\xi_n = 0$表示该点没有犯错，$\alpha_n = 0$表示该点不是SV。所以对应的点在margin之外（或者在margin上），且均分类正确。

若$0 < \alpha_n < C$，得$\xi_n = 0$，且$y_n(w^T z_n + b) = 1$。$\xi_n = 0$表示该点没有犯错，$y_n(w^T z_n + b) = 1$表示该点在margin上。这些点即free SV，确定了b的值。

若$\alpha_n = C$，不能确定ξ_n是否为零，且得到$1 - y_n(w^T z_n + b) = \xi_n$，这个式表示该点偏离margin的程度，$\xi_n$越大，偏离margin的程度越大。只有当$\xi_n = 0$时，该点落在margin上。所以这种情况对应的点在margin之内负方向（或者在margin上），有分类正确也有分类错误的。这些点称为bounded SV。

4.Model Selection

对于Gaussian SVM，不同的参数(C, γ)，会得到不同的margin：



其中横坐标是C逐渐增大的情况，纵坐标是γ逐渐增大的情况。不同的(C, γ)组合，margin的差别很大。用validation选择最好的(C, γ)等参数。

由不同(C, γ)等参数得到的模型在验证集上进行cross validation，选取E_{cv}最小的对应的模型就可以了
例如上图中各种(C, γ)组合得到的E_{cv}如下图所示：

V-Fold cross validation的一种极限就是Leave-One-Out CV，也就是验证集只有一个样本。对于SVM问题，它的验证集Error满足：

$$E_{loocv} \leq \frac{SV}{N}$$

那么，对于non-SV的点，它的$g^- = g$，即对第N个点，它的Error必然为零：

$$e_{non-SV} = err(g^-, non - SV) = err(g, non - SV) = 0$$

另一方面，假设第N个点$\alpha_N \neq 0$，即对于SV的点，它的Error可能是0，也可能是1，必然有：

$$e_{SV} \leq 1$$

综上所述，即证明了$E_{loocv} \leq \frac{SV}{N}$。这符合我们之前得到的结论，即只有SV影响margin，non-SV对margin没有任何影响，可以舍弃。

一般来说，SV越多，表示模型可能越复杂，越有可能会造成过拟合。所以，通常选择SV数量较少的模型，然后在剩下的模型中使用cross-validation，比较选择最佳模型。