Support vector machines*

In a p-dimensional space, a hyperplane is a flat affine subspace of dimension p-1.

$\beta_0 + \beta_1 x_1 + \dots \beta_p x_p = 0$ defines a p-dimensional hyperplane, in the sense that if a point $X = (x_1, \dots x_p)^T$ in p-dimensional space satisfies, then X lies of the hyperplane. if >0 above the plane, <0 below it.

- we compute the perpendicular distance from each point to the plane, the smallest is the margin, and we want to maximize it. we can classify test on which side of the maximal margin hyperplane it lies. The hyperplane thus only depends on the support vectors, but not the others.
- to get the maximal margin classifier $\underset{\beta_0, \beta_1, \dots \beta_p, M}{\text{maximize } M}$ subject to $\sum_{j=1}^{p} \beta_j^2 = 1$

$$y_i(\beta_0 + \beta_1 x_{i1} + \dots \beta_p x_{ip}) \geq M \quad \forall i = 1, \dots, n$$

- The support vector classifier (soft margin classifier), rather than seeking the largest possible margin, we allow some observations to be on the incorrect side of the hyperplane

$$\underset{\beta_0, \beta_1, \dots \beta_p, \epsilon_1, \dots \epsilon_n, M}{\text{maximize } M} \quad \text{subject to } \sum_{j=1}^{p} \beta_j^2 = 1$$

$$y_i(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}) \geq M(1 - \epsilon_i)$$

$$\epsilon_i \geq 0, \quad \sum_{i=1}^{n} \epsilon_i \leq C \quad (C \text{ chosen by CV})$$

$\epsilon_1, \dots \epsilon_n$ are slack variables that and allow observations to be on the wrong side of the margin or hyperplane.

if $\epsilon_i = 0$ then ith observation is on the right side, $\epsilon_i > 0$ wrong side of the margin, $\epsilon_i > 1$ wrong side of hyperplane

→ for $C > 0$ no more than C observations can be on the wrong side of the hyperplane. when C is small, we seek narrow margins that are rarely violated (low bias but high variance). when C is larger, the margin is wider and we allow more violations (more biased, less variance).

- The support vectors with soft margin are the ones that lie directly on the margin, or on the wrong side of the margin, hence when C is large → margin is larger, → many observations are support vector → reduce variance.
- The support vector machines (for non linear data)

we can change the feature space $(x_1, x_2, \dots x_p \to x_1, x_1^2, x_2, x_2^2, \dots)$. The solution is linear in the new space but not in the former one where it is non-linear.

- To enlarge the space in a computational friendly way we can use kernel.
- The inner product of two observations $x_i, x_{i'}$ is given by $\langle x_i, x_{i'} \rangle = \sum_{j=1}^{p} x_{ij} x_{i'j}$
- we can represent the linear vector classifier as $f(x) = \beta_0 + \sum_{i=1}^{n} \alpha_i \langle x, x_i \rangle$ where there are n parameters α_i, $i = 1, \dots, n$ one per training observation.

to estimate $\alpha_1, \dots \alpha_n$ and β_0, we need the $\binom{n}{2}$ inner product between all pairs of training. But α_i is non zero only for the support vector.

- Now suppose that every time the inner product appears in the representation, or in the calculation, we replace with a generalization of the inner product → a kernel

linear kernel $K(x_i, x_{i'}) = \sum_{j=1}^{p} x_{ij} x_{i'j}$ (classify quantify the similarities of a pair of observations using Pearson (standard correlation)

- Polynomial kernel $K(x_i, x_{i'}) = (1 + \sum_{j=1}^{p} x_{ij} x_{i'j})^d$. when the support vector classifier is combined with a non linear kernel, the result is a support vector machine, of the form $f(x) = \beta_0 + \sum_{i \in S} \alpha_i K(x, x_i)$
- radial kernel $K(x_i, x_{i'}) = \exp(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2)$. It has local behavior, only nearby training observations have an effect on the class label.

SVM with more than two classes

- **one versus one**: if there are $K > 2$ classes, we construct $\binom{K}{2}$ SVM, each of which compares a pair of classes. the final class is the most assigned
- **one versus all**, we fit K SVM, each time comparing one of the k classes to the remaining $k-1$ classes

Logistic regression

$f(x) = \beta_0 + \beta_1 x_1 + \dots \beta_p x_p$ as $\min_{\beta_0 \dots \beta_p} \left\{ \sum_{i=1}^{n} \max[0, 1 - y_i f(x_i)] + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$ when λ is large, $\beta_0 \dots \beta_p$ are small -> more violation -> low variance high bias

=> generalization $\min_{\beta_0, \beta_1, \dots \beta_p} \{ L(X, Y, \beta) + \lambda P(\beta) \}$

Support vector regression

Fits coefficients that minimize a different type of loss, where only residuals larger in absolute value than some positive constant contribute to the loss function. This is an extension of the margin used in support vector classifier to the regression setting.