Model Complexity and Regularization

This is a note about model complexity and regularization.

Firstly, I would talk about why regularization could lead to a maximum Margin in SVM (Support Vector Machine).
Secondly, I would share my understanding of model complexity, which is the sum of model structure, number of coefficients and volume of the coefficients space.
Then, I would discuss why regularization can help control model complexity, from a geometric perspective in the feature space.
Lastly, I would do the algebra derivation from a Bayesian perspective.

Outline

1. Regularization and Maximum Margin of Support Vector Machine
2. Model and Model Complexity
3. Regularization and Model Complexity
4. Bayesian Probability and Regularization
5. Summary

1. Regularization and Maximum Margin of Support Vector Machine

Hypothesis of SVM:

$$h_\Theta(x) = \begin{cases} 1 & if\ \Theta^T x \geq 0 \\ 0 & if\ \Theta^T x < 0 \end{cases}$$

The optimization objective of SVM:

$$argmin_\Theta C\ \Sigma_{i=1}^m [y_i cost_1(\Theta^T x_i) + (1-y_i) cost_0(\Theta^T x_i)] + \Sigma_{j=1}^n \Theta_j^2$$

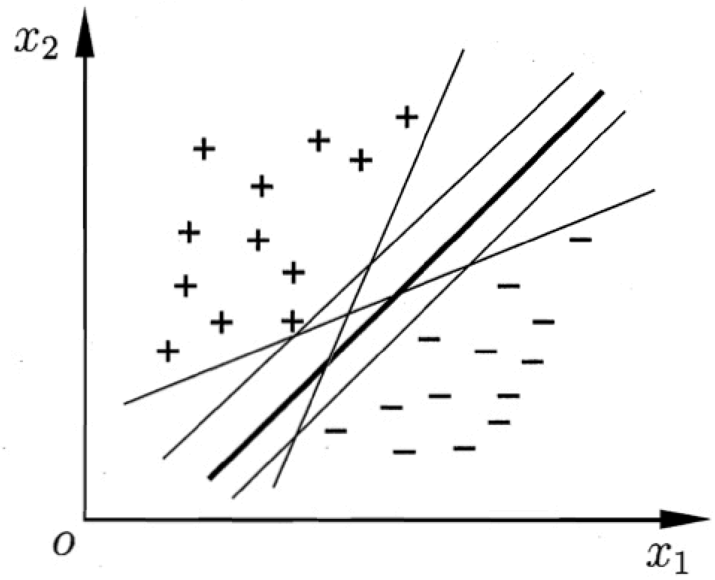
where:

$$cost(z) = \begin{cases} cost_1(z) & \text{, if y} = 1 = \begin{cases} 0 & if\ z \geq 1 \\ a(1-z) & if\ z < 1 \end{cases} \\ cost_0(z) & \text{, if y} = 0 = \begin{cases} 0 & if\ z \leq -1 \\ a(z-(-1)) & if\ z < 1 \end{cases} \end{cases}$$

If we substitute the first part in the SVM objective, we can get with the Logistic Regression loss function: $y_i log(h_\Theta(x_i)) + (1-y_i) log(1-h_\Theta(x_i))$, the objective would become almost same as that of Logistic Regression with regularization term:

$$argmin_\Theta - \frac{1}{m} \Sigma_{i=1}^m [y_i log(h_\Theta(x_i)) + (1-y_i) log(1-h_\Theta(x_i))] + \frac{\lambda}{2m} \Sigma_{j=1}^n \Theta_j^2$$

In summary, despite the cost function, SVM and Logistic Regression are basically the same model. If we take a look from the geometric perspective, we would get the same result.
Considering a linear separable example, the two models are both trying to find a linear decision boundary to separate positive and negative cases:



Suppose the decision boundary is:

$$w^T x + b = 0$$

Then the distance between a training x_0 and the boundary (w, b) can be described as:

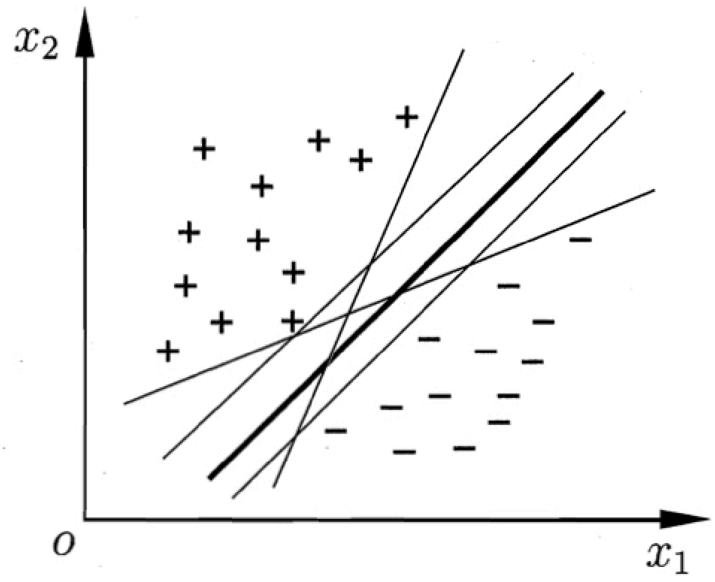
$$r = \frac{|\Theta^T x + b|}{||\Theta||}$$

If the decision boundary can classify all the cases right, which means:

$$\begin{cases} \Theta^t x_i + b \geq +1, y_i = +1 \\ \Theta^t x_i + b \leq -1, y_i = -1 \end{cases}$$

As the following figure shows, the points that are closest to the boundary hold the equal sign. They are called ‘support vectors’. The sum between two support from different classes (positive and negative) to the decision boundary respectively is the ***margin***:

$$\gamma = \frac{2}{||\Theta||}$$



We would the margin to reach its maximum, which is the same as

$$argmin_{\Theta} ||\Theta||^2 = argmin_{\Theta} \Sigma_{j=1}^{n} \Theta_j^2$$

Taking this ***margin*** into Logistic Regression enables us to understand the L2-regularization from a new perspective: the regularization term is helping to maximize the **gap** between positive and negative cases.

2. Model and Model Complexity

In Machine Learning, we always hear about many models: Linear Regression, Logistic Regression, Support Vector Machine, Decision Tree, Random Forest, XGBoost, Neural Networks, etc. When we talk about these models, mostly we refer to an untrained model structure, with multiple coefficients and hyper-parameters to be determined. Once they have been determined, the models can make predictions.

The term model complexity is usually combined with those untrained models. Despite in common sense how complex or how abstract a model is, model complexity defines how 'free' a model can vary to fit potential data. Consider the following three comparisons:

1. **Decision Tree and Random Forest**
   Though, they are both tree based models, it is kind of obvious that Random Forest is a more complex model than Decision Tree, since Random Forest is an ensemble of Decision Trees.
2. **Linear Regression and Polynomial Regression**
   Linear Regression: $h_{\theta 1}(x) = \theta_0 + \theta_1 x$
   Polynomial Regression: $h_{\theta 2}(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + ... + \theta_{10} x^{10}$
   Polynomial is more complex than Linear Regression since it has more coefficients, and that it can fits far more curves.
   For example, $h_{\theta}(x) = 3 + 2x + 4x^2$ can be described by Polynomial Regression but not by Linear Regression.
3. **Two Linear Regressions**
   Model 1: $h_{\theta 1}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$, with $\theta_i \in [-10, 10] \bigcap Z$
   Model 2: $h_{\theta 2}(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2$, with $\theta_i \in [-10000, 10000] \bigcap Z$
   Model 2 had more complexity, since it has more space to 'travel'.
   For example, $h_{\theta}(x) = 11 + 12x_1 + x_2$ can be described by Model 2 but not by Model 1.

In summary, we have:

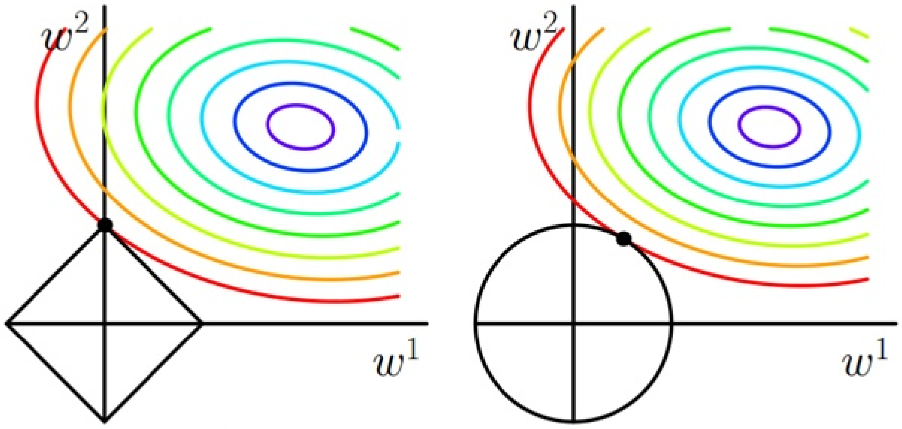
$$Complexity = Model\ Complexity + No.\ of\ Coefficients + Measure\ of\ Feature\ Space$$

3. Regularization and Model Complexity

For each machine learning problem, once we have sticked to one certain model (Linear Regression, SVM, etc.), the optimization objective $J(\theta)$ of the coefficients estimating process often consist of the following two parts:

$$argmin_\theta\ J(\theta) = J_0(\theta) + \lambda Reg(\theta)$$

where $J_0(\theta)$ is often refered to as the cost function, $Reg(\theta)$ is often refered to as the regularization term. λ is the regularization parameter.



(a) ℓ_1-ball meets quadratic function. ℓ_1-ball has corners. It's very likely that the meet-point is at one of the corners.

(b) ℓ_2-ball meets quadratic function. ℓ_2-ball has no corner. It is very unlikely that the meet-point is on any of axes.

In the above figures, considering a two dimensional coefficients' space, the contours stand for the cost function part, where on each contour line, the cost function has the same value; whereas the square and the circle stand for the regularization part of the optimization objective.

Further away from the center of the contour, the cost function $J_0(\theta)$ gets larger; further away from the origin, the regularization term is larger. In order to find the coefficients that minimizes the objective $J(\theta)$, we shall find a balanced point between the origin and the center of the contours.

During this process, the regularization parameter λ control the balance. When it is large, the balanced point would be close to the origin, since we are putting more weight on the regularization term $Reg(\theta)$, vice versa.

4. Bayesian Probability and Regularization

Part 3 and part 4 are all from intuitive perspective. What is the statistical mechanics of regularizations?
Lets discuss from a Bayesian perspective.
The Bayesian Probability is defined as follow:

$$P(A|B) = \frac{P(A|B)P(A)}{P(B)}$$

where $P(A|B)$ is called posterior, $P(B|A)$ is called likelihood, $P(A)$ is called prior, and $P(B)$ is called evidence.

When we are solving the for the parameters, we are trying to maximize the posterior probability of the coefficients $P(\theta|D)$ given the training set D: (symboling the problem as w)

$$w = argmax_{\theta} P(\theta|D) = argmax_{\theta} \frac{P(\theta|D)P(\theta)}{P(D)} = argmax_{\theta} P(\theta|D)P(\theta)$$

Taking log:

$$w = argmax_{\theta}(ln(P(D|\theta)) + lnP(\theta))$$

If the prior comes from a Standard Normal Distribution, then:

$$lnP(\theta) = ln\frac{1}{\sqrt{2\pi}} - \frac{1}{2}||\theta||_2^2$$

Then w would become:

$$\begin{gathered} w = argmax_{\theta}(ln(P(D|\theta)) + lnP(\theta)) = argmax_{\theta}(ln(P(D|\theta)) + ln\frac{1}{\sqrt{2\pi}} - \frac{1}{2}||w||_2^2) \\ = argmax_{\theta}(ln(P(D|\theta)) - \frac{1}{2}||\theta||_2^2) = argmin_{\theta}(-ln(P(D|\theta)) + \frac{1}{2}||\theta||_2^2) \end{gathered}$$

where the first part of the sum corresponds to the Maximum Likelihood Estimation (MLE) and the second part of the sum corresponds to the L2-Regularization.

b. If the prior comes from a Standard Laplace Distribution, then:

$$lnP(\theta) = ln\frac{1}{2} - ||\theta||_1$$

Then w would become:

$$w = argmin_{\theta}(-ln(P(D|\theta)) + ||w||_1)$$

where the first part of the sum corresponds to the Maximum Likelihood Estimation (MLE) and the second part of the sum corresponds to the L1-Regularization.

5. Summary

This note dives into the regularization in the regularization of linear Machine Learning models:

$$argmin_{\theta}\ J(\theta) = J_0(\theta) + \lambda Reg(\theta)$$

From common senses, we have this relationship:

$$Complexity = Model\ Complexity + No.\ of\ Coefficients + Measure\ of\ Feature\ Space$$

The widely acknowledged purpose of regularization is to control the model complexity, so that we can prevent overfitting. Because regularizations can limit the coefficients space, and that they cannot have too much variation.

Despite this, this note also walk through the geometric background of regularization: it enables maximum margin between different classes of the outcome variable.

Also, the note verifies the regularization term from a Bayesian perspective, where regularization corresponds to the maximum posterior probability of the coefficients.