Data Mining Assignment-1
k-Times Markov Sampling for SVMC

Bindu (IIT2018105)
Vikas(IIT2018110)
Ayushi Gupta(IIT2018118)
Sanjana(IIT2018120)
TS. Chinmay(IIT2018138)

I. Abstract -

In this paper for numerical analysis svmc with k-Times Markov sampling is used.
In contrast to the classical SVMC and the previously established SVMC algorithm based on Markov sampling, the SVMC algorithm with k-times Markov sampling not only has lower misclassification rates, takes less time to sample and practice, but also generates a more sparse classifier.

II. Introduction -

We equate the k-times Markov sampling SVMC to the classical SVMC and the Markov sampling SVMC implemented in [1]. These comparisons reveal that, as contrasted to the classical SVMC and the SVMC with Markov sampling in [1], the SVMC with k-times (k = 1, 2), Markov sampling has three advantages at the same time

The rates of misclassification are lower.
1) the misclassification rates are lower
2) the average time spent sampling and training is shorter; 3) the classifiers obtained are lacking in accuracy.

III. Algorithm -

Algorithm 1:
SVMC Algorithm Based on k Times Markov Sampling for Balanced Training Samples
Input: ST , N, k, q, n2
Output: sign(fk)
Step 1: N samples are taken at random from ST, Siid := zj{ j=1to N}. Siid should be trained using SVMC to achieve a provisional learning model f0. Let's say I equals 0.
Step 2: Assume N+ = 0, N = 0, and t = 1.
Step 3: Draw a reference zt from ST at random and label it the new sample. If the label of zt is +1, let N+ = N+ + 1, and if the label of zt is 1, let N = N + 1.

Step 4: Calculate the ratio, = e^((fi,z)/e(fi,zt)) by drawing a random sample z from ST and calling it the candidate sample.
Step 5 : If $\alpha \geq 1$, yt y* = 1 accept z* with probability $\alpha 1$ = e−y* fi /e−yt fi . If $\alpha = 1$ and yt y* = −1 or $\alpha < 1$, accept z* with probability α. If there are n2 candidate samples can not be accepted continually, then set $\alpha 2 = q\alpha$ and accept z* with probability $\alpha 2$. If z* is not accepted, go to Step 4, else let zt+1 = z*, N+ = N+ + 1 if the label of zt+1 is +1 and N+ < N/2, or let zt+1 = z*, N− = N−+1 if the label of zt+1 is −1 and N− < N/2 (if the value α (or $\alpha 1$, $\alpha 2$) is bigger than 1, accept the candidate sample z* with probability 1).
Step 6: Return to Step 4 if N+ +N N, otherwise we get N Markov chain samples SMar. Let I = I + 1 be the case.
SVMC can be used to train SMar and acquire a learning model.
Step 7: If i < k, go to Step 2, else output sign(fk).

Algorithm 2;
SVMC Algorithm Based on k Times Markov Sampling for Unbalanced Training Samples
Input: ST , N, k, q, n2
Output: sign(fk)
Step 1: N samples are taken at random. From ST, Siid := zj{j=1 to N}. Siid should be trained using SVMC to achieve a provisional learning model f0. Let's say I equals 0.
Step 2: Let Ni = 0, t = 1.
Step 3: Draw a reference zt from ST at random and label it the new sample. Let Ni = Ni + 1 be the case.
Step 4: Draw a new sample z from ST at random, which we'll refer to as the candidate sample. Calculate the ratio α, α = e^(−(fi,z*)/e−(fi,zt)).
*Step 5:*If $\alpha = 1$, yt y* = 1 accept z* with probability $\alpha 1$ = e−y* fi /e−yt fi . If $\alpha = 1$ and yt y* = −1 or $\alpha < 1$, accept z* with probability α. If there are n2 candidate samples can not be accepted continually, then set $\alpha 2 = q\alpha$ and accept z* with
probability $\alpha 2$. If z* is not accepted, go to Step 4, else let zt+1 = z*, Ni = Ni + 1 (if α (or $\alpha 1$, $\alpha 2$) is greater than 1, accept z* with probability 1).
Step 6: Return to Step 4 if Ni > N; otherwise, we get N Markov chain samples SMar. Let's say I = i+1. SVMC is used to train SMar and gain a learning model fi.
Step 7: If i < k, go to Step 2, else output sign(fk).
.

IV. Conclusions -

To improve the performance two algorithms were discussed, one of them is for balanced training samples while the other deals with unbalanced samples as in real life two class classification dataset is available.
The experimental results revealed that the SVMC with k-times (k = 1, 2) Markov sampling outperforms the classical SVMC and the SVMC with Markov sampling implemented in [1] in terms of learning efficiency (misclassification speeds, total time of sampling and preparation, and support vector numbers).