Assignment 2

Devansh Jain, 190100044

26 Sept 2021

Contents

1 Perceptron

1.1 CS 337: Theory

1.

Both 1-vs-1 and 1-vs-rest approach use binary classification algorithm.
1-vs-rest has K classifiers - one for each class.
1-vs-1 has K(K-1)/2 classifiers - one for each pair of classes.
In 1-vs-rest, each classifier predicts score (probability) and the class with highest score is chosen.
In 1-vs-1, each classifier predicts one class and the class which has been predicted the most is chosen.

Advantages and Disadvantages
Single classifier in 1-vs-1 uses subset of data, so each classifier is faster for 1-vs-1.
1-vs-rest trains less number of classifiers and hence is faster overall.
1-vs-1 is less prone to imbalance in dataset due to dominance in specific classes.

2.

$$\begin{aligned}
score(f, y) &= \sum_i f_i w_{y_i} \\
&= f^T w_y
\end{aligned}$$

$$\begin{aligned}
score(f, y)_{new} &= f^T w_{y_{new}} \\
&= f^T w_{y_{old}} + f^T f \\
&= f^T w_{y_{old}} + ||f||_2^2 \\
&\geq f^T w_{y_{old}} \\
&\geq score(f, y)_{old}
\end{aligned}$$

$$\begin{aligned}
score(f, y')_{new} &= f^T w_{y'_{new}} \\
&= f^T w_{y'_{old}} - f^T f \\
&= f^T w_{y'_{old}} - ||f||_2^2 \\
&\leq f^T w_{y'_{old}} \\
&\leq score(f, y')_{old}
\end{aligned}$$

3.

$$\mathcal{L}(f, y) = max(0, -yf^T w)$$

$$\nabla_w \mathcal{L}(f, y) = \begin{cases} 0 & \text{if } yf^T w \geq 0 \\ -yf & \text{if } yf^T w < 0 \text{, i.e. point is misclassified} \end{cases}$$

Gradient descent update:

$$w^{(k+1)} = w^{(k)} - \eta \nabla_w \mathcal{L}(f, y)$$

$$w^{(k+1)} = \begin{cases} w^{(k)} & \text{if } yf^T w \geq 0 \\ w^{(k)} + \eta yf & \text{if } yf^T w < 0 \text{, i.e. point is misclassified} \end{cases}$$

We can see that the gradient descent update rule is similar to perceptron update rule.
Using the fact that the perceptron update rule converges for linearly separable dataset, we can conclude that gradient descent algorithm also converges for linearly separable dataset.

P.S. The loss function takes into account only a single data point. To be precise, we are doing stochastic gradient descent here.

4.

In proof of the convergence theorem, if we use $\eta = 0.5$.
We get $\sqrt{k}r \geq ||w^{(k)}||_2 \geq u^T w^{(k)} \geq k\eta\gamma$.
Thus, $k \leq \dfrac{r^2}{\eta^2\gamma^2}$.
Upper bound on number of iterations under the modified algorithm is 4M.

<u>Alt</u>:
If we do double update at every step, i.e. take the same point again for every misclassification. It would be equivalent to original perceptron, so number of iterations would be 2M.
This is possible only if we chose the points in this fashion.

5.

$k \leq \dfrac{r^2}{\gamma^2}$, where $r \geq ||f||_2 \ \forall f \in \mathcal{D}$ and $\exists u, ||u||_2 = 1, \gamma \leq |u^T f| \ \forall f \in \mathcal{D}$.
Here, $f = [f_1 \ f_2 \ 1]$.
So, $r = \sqrt{3}$ and if we take $u = \dfrac{[1 \ 1 \ -1.5]}{\sqrt{4.25}}$, then $\gamma = \dfrac{1}{\sqrt{17}}$.
Thus, $k \leq 51$.