IE613: Online Machine Learning **Jan-Apr 2016**

Lecture 13: Perceptron and Minnow Algorithm

Lecturer: M. K. Hanawal *Scribes: Shivangi Saklani*

13.1 Recapitulation

We have considered the following algorithms till now under adversarial and bandit setting with assumptions varying from realizability to full and incomplete information settings:

- **Realizability Assumption:** All labels are generated by some hypothesis, $h^* \in \mathcal{H}$ i.e. y_t in $S = \{(x_t, y_t) : t = 1 \ \ to \ \ T\}$ is not randomly chosen but are such, that:
  $\exists \quad h^* \in \mathcal{H}$ such that $y_t = h_t^*(x_t)$
  - **Consistent Algorithm:** In this algorithm, we choose a hypothesis h from our hypothesis class $\mathcal{H}$ and generate a label y_t^*, the adversary then tells us about true label y_t and according to the loss occurrence, the hypothesis class is updated at every time point T.
    The mistake bound on this algorithm is $M_{\mathcal{H}}(CA) \leq |\mathcal{H}| - 1$
  - **Halving Algorithm:** In this algorithm, the label y_t^* is generated using majority voting, i.e. if for an instance x_t, majority of $h_t \ \in \ \mathcal{H}$ gives 0, then $y_t^* = 0$ and vice versa. The advantage with this algorithm is that we know the loss corresponding to each hypothesis that we could have chosen and hence at each iteration, we chuck out atleast half of the hypothesis from the Hypothesis class.
    The mistake bound on this algorithm is $M_{\mathcal{H}}(HA) \leq log_2|\mathcal{H}|$
  - **Standard Optimal Algorithm:** This algorithm is in close sync with Halving Algorithm and differs only in the fact that this algorithm considers Ldim($\mathcal{H}$) instead of $|\mathcal{H}|$

  In the unrealisable setting, we have the following cases:

- **Full Information Setting:** In such a setting even though we predict y_t^* according to some hypothesis h, irrespective of the loss incurred, we would come to know about the loss we would have incurred from every other hypothesis.
  - **Weighted Majority Algorithm:** In this algorithm, since the adversary is not choosing its label from any of the hypothesis in our hypothesis class, we do not eliminate our hypothesis at each time point in this method, instead we try to predict as close to the adversary's choices as possible by varying the probabilities of choosing a hypothesis depending on its past record of incurring a loss.
    In unrealisable cases, we have the following concepts:
    * **Regret:** $R_{(H)}(A,T) = \underset{(x_1,y_1)\ldots(x_T,y_T)}{sup}\left[\sum_{t=1}^{T}|y_t^* - y_t| - \underset{h\in\mathcal{H}}{inf}\sum_{t=1}^{T}|h(x_t) - y_t|\right]$
    * **Expected Regret:** $\mathbb{E}\big[R_{(H)}(A,T) = \underset{(x_1,y_1)\ldots(x_T,y_T)}{sup}\left[\sum_{t=1}^{T}|P_t^* - y_t| - \underset{h\in\mathcal{H}}{inf}\sum_{t=1}^{T}|h(x_t) - y_t|\right]$
      where $P_t^* = \mathbb{P}\{y_t^* = 1\}$

* **Pseudo Regret:** $\mathbb{E}\Big[\sum_{t=1}^{T} l_{I_t,t}\Big] - \min_{k\in[K]} \mathbb{E}\Big[\sum_{t=1}^{T} l_{k,t}\Big]$

The Expected Regret, $\mathbb{E}\Big[R_{\mathcal{H}}(WM,T)\Big] \leq \sqrt{2 * log|\mathcal{H}|T}$

- **Incomplete Information Setting:** In this setting, we only get to know the loss corresponding to the arm that we have chosen, in general this setting is also popularly known as the bandit setting. We would now be referring to the hypothesis as arms and once an arm is chosen we would only get an information about the loss incurred on choosing that arm and absolutely no information about the loss corresponding to other arms in the hypothesis class.

  - **Exp3:** In this algorithm, since we do not have full information setting, we use the past track of error made by each arm to update their respective probabilities and since we do not have information about the loss values corresponding to other arms, hence we use an unbiased estimator of loss function to predict the same.

  - **Exp3p:** This algorithm is a revised version of exp3 wherein a tradeoff is made between the unbiasedness of the estimator and the exploration of arms which once made an error. Exp3 reduces the probability of arms that make an error almost close to zero and we sort of lose an opportunity to try out those arms in further rounds where they might have performed better. This algo differs from exp3 in only the estimation of loss value and updation of probabilities wherein exp3p ensures that all arms are sufficiently explored.

  - **Exp3Ix:** In EXP3 algorithm we only focused on exploitation whereas in EXP3.P we also considered exploration along with exploitation. We argued that in EXP3 we can show the bounds in expectations whereas in EXP3.P we can show the bounds in expectation as well as in high probability. Now the question is can we get this bound in Expectation and in high probability only by doing the exploration and the answer to this question is yes! we can do this and the algorithm that we have for this EXP3-IX.

13.2 Perceptron Algorithm

We now consider a setting for binary classification, where $x_t \in \mathcal{X} \subset \mathbb{R}^d$ and labels $y_t \in \{-1, +1\}$.
$\{w : w \in \mathbb{R}^d\}$ forms the hypothesis space here which is also assumed to be a linear in nature.
Rule for prediction in this algorithm is:
At round t, we have w_t and an instance x_t. We predict our corresponding label y_t^* by considering the dot product of x_t *and* w_t i.e. $< w_t, x_t > \geq 0 \implies y_t^* = +1$
$< w_t, x_t > < 0 \implies y_t^* = -1$
A loss might be incurred depending on the prediction of y_t^* and w_t is updated if a mistake is made.
Rule for updation in this algorithm is:
if $loss(y_t^*, y_t) = \mathbb{1}_{\{y_t^* \neq y_t\}}$ In this setting we have assumed the hypothesis space to be linear separable, which is an assumption close to realizability and states that $\exists\ w_t^* \in \mathbb{R}^d$ *such that* $y_t\ < x_t, w_t > \ \geq\ 0\ \forall t$

Algorithm 1: Perceptron algorithm

Input : Linear Hypothesis Set $\{w : w \in \mathbb{R}^d\}$
Output: Predictions for the labels
initialize w=w_0;
for *t in range* $1, \ldots, T$ **do**
  receive x_t;
  dot=$< x_t, w_t >$;
  if *dot* $\geq$ 0 **then**
    Set $y_t^* = +1$
  end
  else
    Set $y_t^* = -1$
  end
  Evaluate $loss_t(y_t^*, y_t) = \mathbb{1}_{y_t^* \neq y_t}$;
  if $loss_t == 1$ **then**
    $w_{t+1} \leftarrow w_t + y_t x_t$
  end
  else
    $w_{t+1} \leftarrow w_t$
  end
end

Theorem 13.1 ***Mistake bound on Perceptron Algorithm:***
Let $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. *Let* $R = \sup_t\{||x_t||_\infty\} < \infty$. *Let* $\gamma > 0$ *and* $w^* \in \mathbb{R}^d$,$||w^*||_2 < \infty$ *such that* $y_t < x_t, w_t > \geq \gamma \ \forall\ t$ *then*
the number of mistakes, $M \leq \frac{R^2||w^*||^2}{\gamma^2}$

Proof: Let us assume that there is a mistake in round t. Then,

$$\begin{aligned} < w_{t+1}, w^* > - < w_t, w^* > &= < w_t, w^* > + < y_t x_t, w^* > - < w_t, w^* > \\ &=< y_t x_t, w^* > \\ &\geq \gamma \qquad (since \ \ w_{t+1} \leftarrow w_t + y_t x_t) \end{aligned}$$

Summing the above inequality over $t = 1, \ldots, T$:
$< w_{T+1}, w^* > - < w_1, w^* > \geq \gamma M$
Assuming $w_1 = 0$:
$< w_{T+1}, w^* > \geq \gamma M$
We now try to find an upper bound on$< w_{T+1}, w^* >$

$$\begin{aligned} ||w_{t+1}||^2 &= ||w_t + y_t x_t||^2 \\ &= ||w_t|^2 + ||y_t x_t||^2 + 2 < w_t, y_t x_t > \\ &\leq ||w_t||^2 + ||y_t x_t||^2 \end{aligned}$$

Reasoning: if a mistake is made, we can either have:

- $y_t = 1,\ y_t^* = -1$ i.e from the perceptron prediction rule, we can conclude that $< x_t, w_t > \ < 0$ and hence $y_t < x_t, w_t > \ \leq 0$
- $y_t = -1,\ t_t^* = +1$ i.e from the perceptron prediction rule, we can conclude that $< x_t, w_t > \ \geq 0$ and hence $y_t < x_t, w_t > \ \leq 0$

Hence we can introduce the last inequality using the fact that the inner product of w_t and $x_t y_t$ would always be negative whenever a mistake is incurred.

we now have:

$$||w_{t+1}||^2 \leq ||w_t||^2 + ||x_t||^2$$
$$\implies ||w_{t+1}||^2 - ||w_t||^2 \leq R^2$$

Summing this for all $t = 1, 2 \ldots, T$:

$$||w_{T+1}||^2 - ||w_1||^2 \leq MR^2$$
$$||w_{T+1}||^2 \leq MR^2 \qquad (since \;\; ||w_1|| = 0) \text{ So, we now have:}$$

$$\gamma M \leq \; < w_{T+1}, w^* > \; \leq ||w_{T+1}||||w^*||$$
$$\gamma^2 M^2 \leq ||w_{T+1}||^2 ||w^*||^2 \leq MR^2 ||w^*||^2$$
$$M \leq \frac{R^2 ||w^*||^2}{\gamma^2}$$

■

This bound works well for high dimensions and is preferred because of the fact that the dimension does not affect the bound directly. Perceptron algorithm follows additive update rule as compared to the other algorithms which use multiplicative update rule

13.3 Minnow Algorithm

This algorithm works exactly as perceptron algorithm, it differs only in its updation mechanism.

Algorithm 2: Minnow algorithm

Input : Linear Hypothesis Set $\{w : w \in \mathbb{R}^d\}$ and a constant η
Output: Predictions for the labels
initialize w=w_0;
for *t in range* $1, \ldots, T$ **do**
  receive x_t;
  dot=$< x_t, w_t >$;
  if *dot* $\geq$ 0 **then**
    Set $y_t^* = +1$
  end
  else
    Set $y_t^* = -1$
  end
  Evaluate $loss_t(y_t^*, y_t) = \mathbb{1}_{y_t^* \neq y_t}$;
  Update $w_{j,t+1} = \frac{w_{j,t} e^{\eta y_t x_{j,t}}}{\sum_{r=1}^{d} w_{r,t} e^{\eta y_t x_{r,t}}}$
end

Similar to the mistake bound on Perceptron algorithm, we introduce a mistake bound on Minnow Algorithm using the following theorem.

Theorem 13.2 ***Mistake bound on Minnow Algorithm:***
Let $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$. *Let* $R = \sup_t\{||x_t||_\infty\} < \infty$. *Let* $\gamma > 0$ *and* $\exists\ w^* \in \mathbb{R}^d, ||w^*||_2 < \infty$ *such that* $y_t < x_t, w_t > \geq \gamma\ \ \forall\ t$ *then*
the number of mistakes, $M_{minnow} \leq \frac{log_e(d)||w^*||_1}{\eta\gamma - ||w^*||_1 log_e(\frac{\eta R + e^{-\eta R}}{2})}$

Both of the bounds of Perceptron and Minnow Algorithms are determined deterministically. Unlike the Perceptron mistake bound, the minnow mistake bound depends directly on the dimension of w and hence is not very useful for higher dimensional hypothesis spaces.