Machine Learning **Spring 2021**

Lecture 9: Dimensionality Reduction, Online Learning

Lecturer: Liwei Wang *Scribe: Shipeng Cen, Chenqi Zhao, Xuheng Li, Kaiyun Tan*

Disclaimer: *These notes have not been subjected to the usual scrutiny reserved for formal publications. They may be distributed outside this class only with the permission of the Instructor.*

9.1 Dimensionality Reduction

Assume we have N points $x_1, x_2, ..., x_N$ in R^n, and we want to map these points to a space with lower dimension, for example R^d, where $d \ll n$. An intuition is to map these points to the "dim" in which the data share high variance and the loss is related to the projection distance. And to minimize the distance loss, we will use Principle Component Analysis method, as an traditional use of SVD. If we put all the points into a matrix A and use SVD on A, we will get $A = \sum_{i=1}^{r} \sigma_i u_i v_i^T$, then we save the d terms with the largest singular value in the above sum equation, surely the we can also get the form by solving the eigenvalues and eigenvectors of AA^T or $A^T A$. It's not difficult, and details will not be repeated.

Here we introduce another method to realize dimensionality reduction.

9.1.1 Johnson-Lindenstrauss Lemma

The Johnson-Lindenstrauss lemma is a fundamental result in dimensionality reduction that states that any m points in high-dimensional space can be mapped to a much lower dimension, $k \geq O(\frac{\log m}{\epsilon^2})$, without distorting pairwise distance between any two points by more than a factor of $(1 \pm \epsilon)$.

To begin with, the proof of this theorem is an existential proof, not a constructive one.

Lemma 9.1 Let Q be a random variable following a χ^2-squared distribution with k degrees of freedom. Then we have following inequality:

$$P((1-\epsilon)k \leq Q \leq (1+\epsilon)k) \geq 1 - 2e^{-(\epsilon^2-\epsilon^3)k/4}$$

Proof:

$$P(Q \geq (1+\epsilon)k) = P(\exp(tQ) \geq \exp((1+\epsilon)tk)), \quad t > 0 \tag{9.1}$$

According to Markov's inequality, we have:

$$P(\exp(tQ) \geq \exp((1+\epsilon)tk)) \leq \frac{E(\exp(tQ))}{\exp((1+\epsilon)tk)))} = \frac{(1-2t)^{-\frac{k}{2}}}{\exp((1+\epsilon)tk)))} \tag{9.2}$$

The equation above we use is the moment-generating function of a χ^2-squared distribution with k degrees of freedom. According to $\frac{\partial f(t)}{\partial t} = 0$,we choose $t = \frac{\epsilon}{2(1+\epsilon)}$ to minimize the RHS.And we get that:

$$P(Q \geq (1+\epsilon)k) \leq \left(\frac{1+\epsilon}{e^{\epsilon}}\right)^{\frac{k}{2}} \tag{9.3}$$

Inspired by Taylor's formula, we have:

$$1+\epsilon \le e^{\epsilon-\frac{\epsilon^2}{2}+\frac{\epsilon^3}{2}} \tag{9.4}$$

So the above inequality comes into the following form:

$$P(Q \ge (1+\epsilon)k) \le e^{-\frac{k\epsilon^2(1-\epsilon)}{4}} \tag{9.5}$$

The statement of the lemma follows by using similar techniques to bound another situation and by applying the union bound. ■

Lemma 9.2 *Let $x \in \mathbb{R}^N$, define $k \le N$ and assume that entries in $A \in \mathbb{R}^{k\times N}$ are sampled independently from $N(0,1)$. Then for any $0 \le \epsilon \le \frac{1}{2}$, we have following inequality:*

$$P\left[(1-\epsilon)||x||^2 \le ||\frac{1}{\sqrt{k}}Ax||^2 \le (1+\epsilon))||x||^2\right] \ge 1-2e^{-(\epsilon^2-\epsilon^3)k/4}$$

Proof: we notice that, $\frac{||Ax||^2}{||x||^2} = \sum_{i=1}^k T_i^2$, where $T_i \sim N(0,1)$, so we know that $\sum_{i=1}^k T_i^2$ obeys χ^2-squared distribution with k degrees of freedom.

$$P\left[(1-\epsilon)||x||^2 \le ||\frac{1}{\sqrt{k}}Ax||^2 \le (1+\epsilon))||x||^2\right] = P\left[(1-\epsilon)k \le \sum_{i=1}^k T_i^2 \le (1+\epsilon)k\right] \tag{9.6}$$

Then we use Lemma 9.1 and the proof is completed. ■

Now we can prove JL lemma.

Lemma 9.3 For any tolerance $\epsilon \in$ (0,1) and any $m > 4$, $k > \frac{8logm}{\epsilon^2(1-\epsilon)}$, there exists a map f:$R^N \rightarrow R^k$,such that for any u,v $\in$ set V of m points in R^N, we have:

$$(1-\epsilon)||u-v||^2 \le ||f(u)-f(v)||^2 \le (1+\epsilon)||u-v||^2 \tag{9.7}$$

Proof: We choose $f = \frac{1}{\sqrt{k}}$. According to Lemma 9.2, we have $P[(1-\epsilon)||x||^2 \le ||\frac{1}{\sqrt{k}}Ax||^2 \le (1+\epsilon))||x||^2] \ge 1-2e^{-(\epsilon^2-\epsilon^3)k/4}$.
According to union bound, we have:

$$P[\text{fail}] \le \binom{m}{2} * 2e^{-(\epsilon^2-\epsilon^3)k/4} \tag{9.8}$$

Notice $\binom{m}{2} \le m^2$, Finally we get following inequality to guarantee the existence of the mapping:

$$P[\text{fail}] \le m^2 e^{-(\epsilon^2-\epsilon^3)k/4} < 1 \tag{9.9}$$

which is equivalent to following inequality:

$$k > \frac{8\log m}{\epsilon^2(1-\epsilon)} \tag{9.10}$$

■

9.2 Online Learning

Online machine learning is a method of machine learning in which data becomes available in a sequential order and is used to update the best predictor for future data at each step.[1] It differs from our previous methods in that: first, online learning mixes training and test phases; second, with online learning, *no distributional assumption* is made and thus there is no notion of generalization. Instead, the performance of online learning algorithms is measured using a *mistake model* and the notion of *regret.* To derive guarantees in this model, theoretical analyses are based on a worst-case or adversarial assumption.

Let's begin with a traditional setting of online learning:

9.2.1 Prediction with expert advice

Assume that there are T rounds and N experts. At the tth round, every expert $i \in [N]$ makes a prediction, denoted as $\widetilde{y_{t,i}} \in \{0,1\}$, then the learner makes prediction $\widetilde{y_t}$. After that, the adversary reveals $y_t \in \{0,1\}$. The objective is to minimize the accumulate loss (here it is 0-1loss) $\sum\limits_{t=1}^{T} I[\widetilde{y_t} \neq y_t]$.

Our simple intuition might lead us to the "follow the leader" strategy, which means at the tth round, we make the same prediction as best expert, that is, the expert who makes the smallest number of mistakes in the former $t-1$ rounds. However, this strategy gives poor performance both theoretically and practically.

9.2.1.1 Weighted Majority Algorithm

In this section we will introduce a boosting-like algorithm, the Weighted Majority (WM) algorithm, that weights the importance of experts and applies Multiplicative Weight Updating to reduce the weight of incorrect experts at each round.

Algorithm 9.2.1: Weighted Majority Algorithm

```
Initialize: $w_{1,i} = 1, \forall i \in [N]$;
Parameter: $\beta \in (0,1)$;
for $t$=1,2,...,$T$ do
    Learner makes weighted majority vote: $\widetilde{y_t} = \begin{cases} 0 & if \sum\limits_{\widetilde{y_{t,i}}=0} w_{t,i} > \sum\limits_{\widetilde{y_{t,i}}=1} w_{t,i} \\ 1 & otherwise. \end{cases}$;
    if $\widetilde{y_t} = y_t$ then
        $w_{t+1,i} \leftarrow w_{t,i} \quad \forall i \in [N]$ ;
    else
        $w_{t+1,i} \leftarrow \begin{cases} \beta w_{t,i} & \widetilde{y_{t,i}} \neq y_t \\ w_{t,i} & \widetilde{y_{t,i}} = y_t \end{cases} \quad \forall i \in [N]$;
    end
end
```

The following theorem presents a mistake bound of WM algorithm after $T \geq 1$ rounds as a function of the number of mistakes made by the best expert.

Theorem 9.4 *For $\beta \in (0,1)$, after all T rounds, define the loss of learner as $L_T = \sum\limits_{t=1}^{T} I[\widetilde{y_t} \neq y_t]$, define the* *loss of ith expert as* $m_{T,i} = \sum_{t=1}^{T} I[\widetilde{y_{t,i}} \neq y_t], \forall i \in [N]$*, define the loss of best expert as* $m_T^* = \min_{i \in [N]} m_{T,i}$*. We have*

$$L_T \leq \frac{m_T^* \log 1/\beta + \log N}{\log(2/(1+\beta))}$$

Proof: To prove this theorem, we first introduce a *potential function.* For any $t \geq 1$, define the potential function as $W_t = \sum_{i=1}^{N} w_{t,i}$. On the one hand, since predictions are generated using weighted majority vote, if the algorithm makes an error at round t, this implies that

$$W_{t+1} \leq \left(\frac{1}{2} + \frac{1}{2}\beta\right) W_t = \frac{1+\beta}{2} W_t \tag{9.11}$$

Since $W_1 = N$ and L_T mistakes are made after T rounds, we thus have the following upper bound:

$$W_{T+1} \leq \left(\frac{1+\beta}{2}\right)^{L_T} N \tag{9.12}$$

On the other hand, since all weights are non-negative, we have $\forall i \in [N]$

$$W_{T+1} \geq w_{T+1,i} = \beta^{m_{T,i}} \tag{9.13}$$

Applying this lower bound to the best expert and combining it with the upper bound in (9.12) gives us:

$$\begin{aligned}
&\beta^{m_T^*} \leq (\frac{1+\beta}{2})^{L_T} N \\
\Rightarrow &m_T^* \log \beta \leq \log N + L_T \log(\frac{1+\beta}{2}) \\
\Rightarrow &L_T \log(\frac{2}{1+\beta}) \leq m_T^* \log \frac{1}{\beta} + \log N
\end{aligned}$$

■

Thus, the theorem guarantees a bound of the following form for WM algorithm:

$$L_T \leq \text{constant} \times [\text{mistakes of best expert}] + O(\log N)$$

Note that as $\beta \to 1$, the constant $\to 2$ (according to L'Hospital's rule).

9.2.1.2 Randomized Weight Updating

We first introduce another algorithm, which is different from the former one mainly in that the voting process is replace with randomized selection of expert.

Algorithm 9.2.2: Randomized weight updating

```
Initialize: $w_{1,i} = 1, i \in [N]$.
Parameter: $\beta \in [\frac{1}{2}, 1)$.
for $t = 1, \dots, T$ do
    Learner chooses $i_t \in [N]$ with probability $w_{t,i_t} / \sum_j w_{t,j}$ and $\tilde{y}_t \leftarrow \tilde{y}_{t,i_t}$.
    Update $w_{t+1,i} \leftarrow \beta w_{t,i}$ for all $i$ such that $\tilde{y}_{t,i} \neq y_t$.
end
```

The expected loss of the learner at round t is

$$l_t = \frac{\sum_i w_{t,i} |\tilde{y}_{t,i} - y_t|}{\sum_j w_{t,j}}, \tag{9.14}$$

and the total expected loss is

$$L_T = \sum_{t=1}^{T} l_t. \tag{9.15}$$

For Algorithm 2, we have the following theorem:

Theorem 9.5 *For $\beta \in [1/2, 1)$, the total expected loss is bounded by*

$$L_T \leq (2 - \beta) m_T^* + \frac{\log N}{1 - \beta}. \tag{9.16}$$

Proof: Consider again the potential function

$$W_t = \sum_i w_{i,t}. \tag{9.17}$$

Note that

$$\begin{aligned} W_{t+1} &= \sum_{\tilde{y}_{t,i} = y_t} w_{t,i} + \beta \sum_{\tilde{y}_{t,i} \neq y_t} w_{t,i} = \sum_i w_{t,i} + (\beta - 1) \sum_{\tilde{y}_{t,i} \neq y_t} w_{t,i} \\ &= W_t + (\beta - 1) \sum_i w_{t,i} |\tilde{y}_{t,i} - y_t| = W_t + (\beta - 1) W_t l_t = W_t [1 + (\beta - 1) l_t]. \end{aligned}$$

So we have

$$W_{T+1} = W_1 \prod_{t=1}^{T} [1 + (\beta - 1) l_t] = N \prod_{t=1}^{T} [1 + (\beta - 1) l_t].$$

On the other hand,

$$W_{T+1} \geq \max_i w_{T,i} = \beta^{m_T^*},$$

so

$$N \prod_{t=1}^{T} [1 + (\beta - 1) l_t] \geq \beta^{m_T^*},$$

which is

$$\sum_{t=1}^{T} \log(1+(\beta-1)l_t) \geq m_T^* \log\beta - \log N.$$

Note that

$$\log(1+(\beta-1)l_t) \leq (\beta-1)l_t,$$

so

$$m_T^* \log\beta - \log N \leq \sum_{t=1}^{T} (\beta-1)l_t = (\beta-1)L_T.$$

Thus we have a bound for L_T:

$$L_T \leq \frac{\log N}{1-\beta} - \frac{\log\beta}{1-\beta} m_T^*.$$

It suffices to show that

$$-\log\beta \leq (1-\beta)(2-\beta) \tag{9.18}$$

for $\beta \in [1/2, 1)$. Let

$$f(\beta) = \log\beta + (1-\beta)(2-\beta),$$

then

$$f'(\beta) = \frac{1}{\beta} - 3 + 2\beta = \frac{(1-\beta)(1-2\beta)}{\beta} \leq 0.$$

So $f(\beta) \geq f(1) = 0$, and (9.18) is proven. ■

For Algorithm 2, the coefficient before m_T^* is improved to $2-\beta$, strictly smaller than 2.

If

$$\beta = 1 - \sqrt{\frac{\log N}{T}} \geq 1/2,$$

combined with $m_T^* \leq T$, we have

$$L_T \leq m_T^* + \sqrt{\frac{\log N}{T}} m_T^* + \sqrt{T\log N} \leq m_T^* + \sqrt{T\log N} + \sqrt{T\log N} = m_T^* + 2\sqrt{T\log N},$$

and the average loss per round is bounded by

$$\frac{L_T}{T} \leq \frac{m_T^*}{T} + 2\sqrt{\frac{\log N}{T}}.$$

9.2.2 Proof of Von Neumann's Minimax Theorem

9.2.2.1 Preliminary

We first introduce Hedge algorithm, a general case of randomized weight updating. Instead of 0-1 loss (i.e. whether an expert gives the right prediction), this time we meet the loss function $g_t(i) \in [0,1]$, which indicates the loss we undertake if we follow expert i at round t.

Algorithm 9.2.3: Hedge

```
Initialize: $w_{1,i} = 1, i \in [N]$.
Parameter: $\beta \in (0,1)$.
for $t = 1, \ldots, T$ do
    Learner chooses $i_t \in [N]$ with probability $w_{t,i_t}/\sum_j w_{t,j}$ and incurs loss$g_t(i)$.
    Update $w_{t+1,i} \leftarrow w_{t,i} \cdot \beta^{g_t(i)}$ for all $i$.
end
```

Similarly, we define the expected loss of the learner at round t:

$$l_t = \frac{\sum_i w_{t,i} g_t(i)}{\sum_i w_{t,i}}$$

and the total expected loss is

$$L_T = \sum_{t=1}^{T} l_t$$

Theorem 9.6 *Applying Alg 3, we have*

$$\text{Regret}_T = L_T - \min_i \sum_{t=1}^{T} g_t(i) = O(\sqrt{T \log N})$$

Proof: Set $\epsilon = -\ln \beta > 0$ and $W_t = \sum_{t=1}^{T} w_{t,i}$ for all $t \leq T$.
It is easy to verify the following inequalities

$$\forall x \geq 0,\ e^{-x} \leq 1 - x + x^2 \tag{9.19}$$

$$1 + x \leq e^x \tag{9.20}$$

Inspecting the sum of weights

$$\begin{aligned} W_{t+1} &= \sum_i w_{t,i} e^{-\epsilon g_t(i)} \\ &\leq \sum_i w_{t,i}(1 - \epsilon g_t(i) + \epsilon^2 g_t(i)^2) \quad \text{(apply (9.19))} \\ &\leq W_t(1 - \epsilon l_t + \epsilon^2) \quad \text{(notice that } g_t(i) \leq 1) \\ &\leq W_t e^{-\epsilon l_t + \epsilon^2} \quad \text{(apply (9.20))} \end{aligned}$$

Let $i^* = \arg\min_i \sum_{t=1}^{T} g_t(i)$, we have

$$e^{-\epsilon \sum_t g_t(i^*)} = w_{T+1,i^*} \leq W_{T+1} \leq W_1 e^{-\epsilon \sum_t l_t + \epsilon^2 T} = N e^{-\epsilon L_T + \epsilon^2 T}$$

Taking logarithm of both sides we get:

$$-\epsilon \sum_t g_t(i^*) \leq \ln N - \epsilon L_T + \epsilon^2 T$$

It immediately follows that

$$\text{Regret}_T \leq \epsilon T + \frac{\ln N}{\epsilon} \leq 2\sqrt{T \ln N} = O(\sqrt{T \log N})$$

■

9.2.2.2 Minimax Theorem and its proof

Theorem 9.7 (Von Neumann's minimax theorem) *For any two-person zero-sum game defined by matrix $M \in \mathcal{R}^{m \times n}$*

$$\min_{p \in \Delta_m} \max_{q \in \Delta_n} p^T M q = \max_{q \in \Delta_n} \min_{p \in \Delta_m} p^T M q \quad (\Delta_k = \{a \in \mathcal{R}^k : \|a\|_1 = 1 \wedge a \succeq 0\})$$

We have learned that if two players follow pure strategy (i.e. restrict p and q to having only one non-zero entry), the result cannot be strictly better for the one who plays second if the playing order is reversed. (i.e. $\min_i \max_j M_{ij} \geq \max_j \min_i M_{ij}$)
In case of two players adopting mixed strategy, it is also intuitive that playing second is better, because playing second means having more information without any cost. So we need only focus on proving the reverse inequality. We will demonstrate that by adopting online learning algorithm, the first player can reduce the disadvantage of playing first to an infinitesimal as learning time goes to infinity.
Proof: The inequality

$$\min_{p\in\Delta_m} \max_{q\in\Delta_n} p^T Mq \geq \max_{q\in\Delta_n} \min_{p\in\Delta_m} p^T Mq \tag{9.21}$$

is straightforward. (Let $q^* \in \arg\max_{q\in\Delta_n} \min_{p\in\Delta_m} p^T Mq$, $p^* \in \arg\min_{p\in\Delta_m} \max_{q\in\Delta_n} p^T Mq$, $LHS = \max_{q\in\Delta_n} p^{*T} Mq \geq p^{*T} Mq^* \geq \min_{p\in\Delta_m} p^T Mq^* = RHS$)

To show the reverse inequality, consider an online learning setting where Alg 3 is applied. In this case, row player (playing fist) is the learner, who chooses p_t such that $(p_t)_i = \frac{w_{t,i}}{W_t}$. Column player is the adversary who always select the optimal adversarial q_t (i.e. $q_t \in \arg\max_{q\in\Delta_n} p_t^T Mq$). There are m experts who keep suggesting choosing one single row. Thus the loss function $g_t(i) = (Mq_t)_i$ (we can let $M_{ij} \in [0,1]$ without loss of generality).
Then from Theorem 9.6 we have

$$L_T - \min_i \sum_{t=1}^T g_t(i) = \sum_{t=1}^T p_t^T Mq_t - \min_i (M \sum_{t=1}^T q_t)_i = O(\sqrt{T \log m})$$

Then

$$\begin{aligned}
\frac{1}{T}\sum_{t=1}^T p_t^T Mq_t &\leq \frac{1}{T}\min_i \left(M\sum_{t=1}^T q_t\right)_i + O\left(\sqrt{\frac{\log m}{T}}\right) \\
&= \min_{p\in\Delta_m}\left(p^T M(\frac{1}{T}\sum_{t=1}^T q_t)\right) + o(1) \\
&\leq \max_{q\in\Delta_n}\min_{p\in\Delta_m} p^T Mq + o(1)
\end{aligned}$$

And we have

$$\min_{p\in\Delta_m}\max_{q\in\Delta_n} p^T Mq \leq \max_{q\in\Delta_n}\left(\frac{1}{T}\sum_{t=1}^T p_t^T\right) Mq \leq \frac{1}{T}\sum_{t=1}^T \max_{q\in\Delta_n} p_t^T Mq = \frac{1}{T}\sum_{t=1}^T p_t^T Mq_t \leq \max_{q\in\Delta_n}\min_{p\in\Delta_m} p^T Mq + o(1)$$

So we have

$$\min_{p\in\Delta_m}\max_{q\in\Delta_n} p^T Mq \leq \max_{q\in\Delta_n}\min_{p\in\Delta_m} p^T Mq$$

Combined with (9.21), the proof is completed. ■

References

[1] https://en.wikipedia.org/wiki/Online_machine_learning

[2] https://www.cs.princeton.edu/~rlivni/cos511/lectures/lect18.pdf