Online learning of quantum processes

Sumeet Khatri

https://sumeetkhatri.com

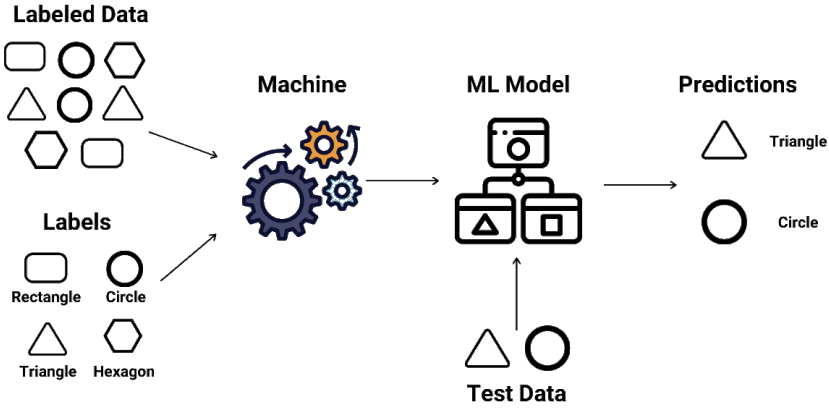
skhatri@vt.edu

November 27, 2024

Online learning of quantum processes

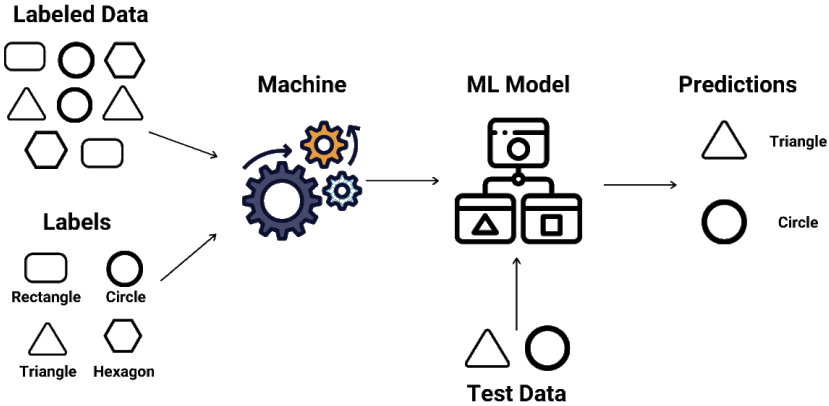
Asad Raza[1], Matthias C. Caro[1], Jens Eisert[1,2], and Sumeet Khatri[1]

[arXiv:2406.04250]



[Adapted from https://medium.com/@ngneha090/a-guide-to-supervised-learning-f2ddf1018ee0

Supervised Learning

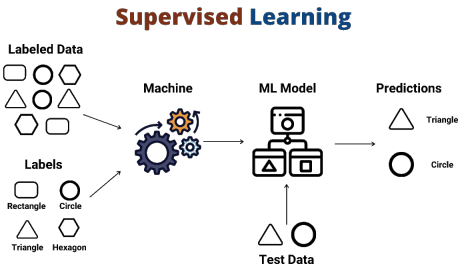


[Adapted from https://medium.com/@ngneha090/a-guide-to-supervised-learning-f2ddf1018ee0

Supervised learning

- ⋆ Distinct training and testing phase
- ⋆ "Batch" learning: hypothesis based on looking at all training data at once
- ⋆ Data assumed IID
- ⋆ Goal: obtain good generalization/prediction on new data points
- ⋆ Analyzed using the PAC framework

[L. G. Valiant, Comm. ACM **27**(11), 1134 (1984)]



[Adapted from

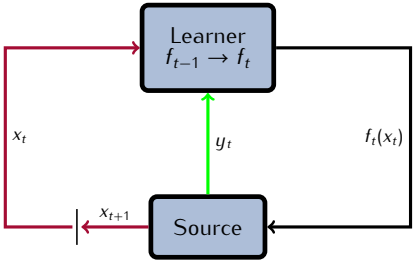
https://medium.com/@ngneha090/a-guide-to-supervised-learning-f2ddf1018ee0

Online learning [N. Littlestone, Machine Learning 2(4), 285 (1988)]

- No training or testing phase
- Data is given sequentially
- Data can be non-IID, adversarial
- Goal: minimize the *number of mistakes* and the *regret*:

$$R_T := \sum_{t=1}^{T} \ell_t(f_t(x_t)) - \min_{f \in \mathcal{F}} \sum_{t=1}^{T} \ell_t(f(x_t)),$$

where ℓ_t are loss functions, e.g., $\ell_t(\cdot) = |(\cdot) - y_t|$.



Learning quantum processes

Our goal: learn expectation values of quantum processes:

$$f(x) \equiv \mathrm{Tr}[M\mathcal{N}(\rho)] = \mathrm{Tr}[(\rho^T \otimes M)C(\mathcal{N})],$$

where $C(\mathcal{N})$ is the *Choi matrix/representation*

$$C(\mathcal{N}) = \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|).$$

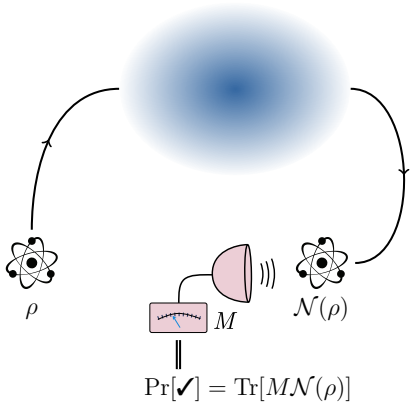
More generally, we aim to learn

$$\mathrm{Tr}[EC(\mathcal{N})], \quad \text{for} \quad 0 \leq E \leq \sigma \otimes \mathbb{1}.$$

The regret is

$$R_T = \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}\mathcal{N}^{(t)}]) - \min_{\mathcal{N}\in\mathrm{CPTP}} \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}C(\mathcal{N})]).$$

$\star$ We aim for a regret upper bound of $\mathcal{O}\left(\sqrt{T\,\mathrm{poly}(\log(d))}\right)$.



Learning quantum processes

Our goal: learn expectation values of quantum processes:

$$f(x) \equiv \mathrm{Tr}[M\mathcal{N}(\rho)] = \mathrm{Tr}[(\rho^T \otimes M)C(\mathcal{N})],$$

where $C(\mathcal{N})$ is the *Choi matrix/representation*

$$C(\mathcal{N}) = \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|).$$

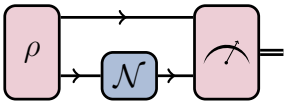
More generally, we aim to learn

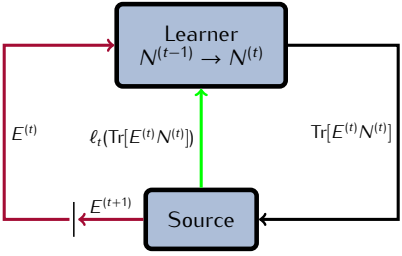
$$\mathrm{Tr}[EC(\mathcal{N})], \quad \text{for} \quad 0 \leq E \leq \sigma \otimes \mathbb{1}.$$

The regret is

$$R_T = \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}\mathcal{N}^{(t)}]) - \min_{\mathcal{N}\in\mathrm{CPTP}} \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}C(\mathcal{N})]).$$

$\star$ We aim for a regret upper bound of $\mathcal{O}\left(\sqrt{T\,\mathrm{poly}(\log(d))}\right)$.





Learning quantum processes

Our goal: learn expectation values of quantum processes:

$$f(x) \equiv \mathrm{Tr}[M\mathcal{N}(\rho)] = \mathrm{Tr}[(\rho^T \otimes M)C(\mathcal{N})],$$

where $C(\mathcal{N})$ is the *Choi matrix/representation*

$$C(\mathcal{N}) = \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|).$$

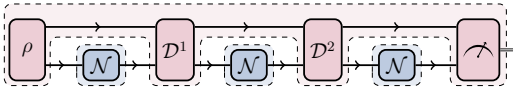
More generally, we aim to learn

$$\mathrm{Tr}[EC(\mathcal{N})], \quad \text{for} \quad 0 \leq E \leq \sigma \otimes \mathbb{1}.$$

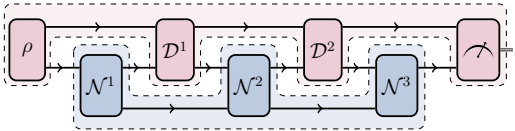
The regret is

$$R_T = \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}\mathcal{N}^{(t)}]) - \min_{\mathcal{N}\in\mathrm{CPTP}} \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}C(\mathcal{N})]).$$

$\star$ We aim for a regret upper bound of $\mathcal{O}\left(\sqrt{T\,\mathrm{poly}(\log(d))}\right)$.



$\Pr[i] = \mathrm{Tr}[E^{(i)}C(\mathcal{N})^{\otimes 3}]$



$\Pr[i] = \mathrm{Tr}[E^{(i)}C(\mathcal{N}^{[3]})]$

Results

	G-gate channels	Pauli channels
Regret upper bounds	$\mathcal{O}\left(L\sqrt{TG\log(Gn)}\right)$	$\mathcal{O}\left(L\sqrt{Tn}\right)$
Mistake upper bounds	$\mathcal{O}\left(\frac{L^2 G\log(Gn)}{\varepsilon^2}\right)$	$\mathcal{O}\left(\frac{L^2 n}{\varepsilon^2}\right)$

- ⋆ Exponential mistake lower bounds for arbitrary channels.
- **G-gate channels:** Channels decomposed as a sequence of at most G two-qubit channels.
- **Pauli channels:** Kraus operators are Paulis.
- ⋆ Results extend to arbitrary convex combinations of known channels.
- ⋆ Results extend to arbitrary convex combinations of multi-time (non-Markovian) processes.
- ⋆ We also have computational complexity lower bounds.

Online learning of quantum states*

Scott Aaronson,[1,6], Xinyi Chen[2,3], Elad Hazan[2,3],
Satyen Kale[4] and Ashwin Nayak[5,7]

[arXiv:1802.09025]

★ Regret upper bound: $\mathcal{O}(\sqrt{Tn})$.

Algorithm Matrix multiplicative weights (MMW) algorithm

Require: $\eta \in (0, 1]$; Initialize $W^{(1)} = \mathbb{1}_d$

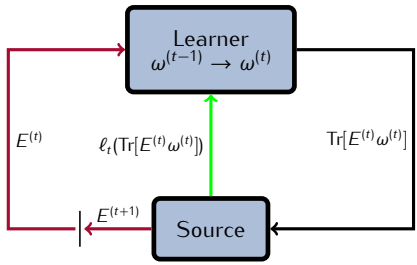
1: **for** $t = 1, 2, \ldots, T$ **do**

2: Output the decision/estimate $\omega^{(t)} = \frac{W^{(t)}}{\mathrm{Tr}[W^{(t)}]}$.

3: Receive the cost matrix $L^{(t)}$, $-\mathbb{1}_d \leq L^{(t)} \leq \mathbb{1}_d$.

4: Update the weight matrix: $W^{(t+1)} = \exp\left[-\eta \sum_{t'=1}^{t} L^{(t')}\right] = \exp[\log(W^{(t)}) - \eta L^{(t)}]$.

5: **end for**



For an arbitrary state ρ:

$$\sum_{t=1}^{T} \mathrm{Tr}[L^{(t)}\omega^{(t)}] \leq \mathrm{Tr}\left[\rho\left(\sum_{t=1}^{T} L^{(t)}\right)\right] + \eta T + \frac{\log d - H(\rho)}{\eta}$$

Why can't we learn the Choi state?

Our goal: learn expectation values of quantum processes:

$$\mathrm{Tr}[EC(\mathcal{N})],$$

where $C(\mathcal{N})$ is the *Choi matrix/representation*

$$C(\mathcal{N}) = \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|).$$

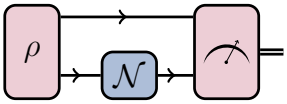
The *Choi state* is

$$\Phi(\mathcal{N}) := \frac{1}{d} \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|) = \frac{1}{d} C(\mathcal{N}),$$

which implies that

$$\mathrm{Tr}[EC(\mathcal{N})] = d\,\mathrm{Tr}[E\Phi(\mathcal{N})],$$

where $d = 2^n$.



Why can't we learn the Choi state?

Our goal: learn expectation values of quantum processes:

$$\mathrm{Tr}[EC(\mathcal{N})],$$

where $C(\mathcal{N})$ is the *Choi matrix/representation*

$$C(\mathcal{N}) = \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|).$$

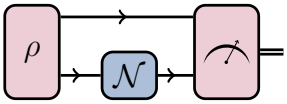
The *Choi state* is

$$\Phi(\mathcal{N}) := \frac{1}{d} \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|) = \frac{1}{d} C(\mathcal{N}),$$

which implies that

$$\mathrm{Tr}[EC(\mathcal{N})] = d\,\mathrm{Tr}[E\Phi(\mathcal{N})],$$

where $d = 2^n$.



⋆ Good regret and mistake bounds for states lead to exponential bounds for channels!

Why can't we learn the Choi state?

Our goal: learn expectation values of quantum processes:

$$\mathrm{Tr}[EC(\mathcal{N})],$$

where $C(\mathcal{N})$ is the *Choi matrix/representation*

$$C(\mathcal{N}) = \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|).$$

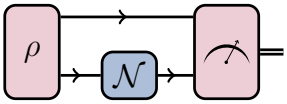
The *Choi state* is

$$\Phi(\mathcal{N}) := \frac{1}{d} \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|) = \frac{1}{d} C(\mathcal{N}),$$

which implies that

$$\mathrm{Tr}[EC(\mathcal{N})] = d\,\mathrm{Tr}[E\Phi(\mathcal{N})],$$

where $d = 2^n$.



★ Good regret and mistake bounds for states lead to exponential bounds for channels!

★ The Choi state also satisfies

$$\mathrm{Tr}_{\mathrm{out}}[\Phi(\mathcal{N})] = \frac{1}{d} \mathbb{1}_{\mathrm{in}}$$

So state online learning algorithms cannot be directly applied.

Why can't we learn the Choi state?

Our goal: learn expectation values of quantum processes:

$$\mathrm{Tr}[EC(\mathcal{N})],$$

where $C(\mathcal{N})$ is the *Choi matrix/representation*

$$C(\mathcal{N}) = \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|).$$

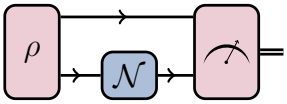
The *Choi state* is

$$\Phi(\mathcal{N}) := \frac{1}{d} \sum_{i,j=0}^{d-1} |i\rangle\langle j| \otimes \mathcal{N}(|i\rangle\langle j|) = \frac{1}{d} C(\mathcal{N}),$$

which implies that

$$\mathrm{Tr}[EC(\mathcal{N})] = d\mathrm{Tr}[E\Phi(\mathcal{N})],$$

where $d = 2^n$.



★ Good regret and mistake bounds for states lead to exponential bounds for channels!

★ The Choi state also satisfies

$$\mathrm{Tr}_{\mathrm{out}}[\Phi(\mathcal{N})] = \frac{1}{d} \mathbb{1}_{\mathrm{in}}$$

So state online learning algorithms cannot be directly applied.

★ Our actual task is to learn the *Choi matrix*!

Regret upper bound for Pauli channels

Theorem

Let $\mathcal{P}$ be an n-qubit Pauli channel. There exists a sequence of Pauli-channel Choi matrices $N^{(t)}$ such that

$$\sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}N^{(t)}]) - \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}C(\mathcal{P})]) = \mathcal{O}(L\sqrt{nT}).$$

Pauli channels have the form

$$\mathcal{P}(\rho) = \sum_{\boldsymbol{z},\boldsymbol{x}\in\{0,1\}^n} p_{\boldsymbol{z},\boldsymbol{x}} P^{\boldsymbol{z},\boldsymbol{x}} \rho P^{\boldsymbol{z},\boldsymbol{x}},$$

where

$$\begin{aligned} P^{\boldsymbol{z},\boldsymbol{x}} &:= (+\mathrm{i})^{\boldsymbol{z}\cdot\boldsymbol{x}} Z^{\boldsymbol{z}} X^{\boldsymbol{x}}, \quad \boldsymbol{x},\boldsymbol{z} \in \{0,1\}^n, \\ Z^{\boldsymbol{z}} &:= Z^{z_1} \otimes Z^{z_2} \otimes \cdots \otimes Z^{z_n}, \\ X^{\boldsymbol{x}} &:= X^{x_1} \otimes X^{x_2} \otimes \cdots \otimes X^{x_n}, \end{aligned}$$

and $\{p_{\boldsymbol{z},\boldsymbol{x}}\}_{\boldsymbol{z},\boldsymbol{x}\in\{0,1\}^n}$ is a probability distribution.

Proof technique: the (matrix) multiplicative weights algorithm

Algorithm Multiplicative weights update (MWU) algorithm [Arora et al., Theory of Computing **8**(6), 121 (2012)]

Require: $\eta \in (0, \frac{1}{2}]$; $T \in \mathbb{N}$ rounds; $d \in \mathbb{N}$ decisions; initial weights $\boldsymbol{w}^{(1)} = (1, 1, \ldots, 1)$.

1: **for** $t = 1, 2, \ldots T$ **do**

2: Output the decision/estimate $\boldsymbol{p}^{(t)} = \frac{\boldsymbol{w}^{(t)}}{\phi^{(t)}}$, $\phi^{(t)} = \sum_{i=1}^{d} w_i^{(t)}$.

3: Receive the cost vector $\boldsymbol{m}^{(t)} = (m_1^{(t)}, m_2^{(t)}, \ldots, m_d^{(t)})$, $m_i^{(t)} \in [-1, 1]$, $i \in \{1, 2, \ldots, d\}$.

4: Update the weights as $w_i^{(t+1)} = w_i^{(t)}(1 - \eta m_i^{(t)})$, $i \in \{1, 2, \ldots, d\}$.

5: Set $\boldsymbol{w}^{(t+1)} = (w_1^{(t+1)}, w_2^{(t+1)}, \ldots, w_d^{(t+1)})$.

6: **end for**

For an arbitrary probability vector $\boldsymbol{q}$:

$$\sum_{t=1}^{T} \boldsymbol{m}^{(t)} \cdot \boldsymbol{p}^{(t)} \leq \sum_{t=1}^{T} (\boldsymbol{m}^{(t)} + \eta |\boldsymbol{m}^{(t)}|) \cdot \boldsymbol{q} + \frac{\log d}{\eta} \leq \sum_{t=1}^{T} \boldsymbol{m}^{(t)} \cdot \boldsymbol{q} + \eta T + \frac{\log d}{\eta}.$$

Proof of the regret upper bound for Pauli channels

Theorem

Let $\mathcal{P}$ be an n-qubit Pauli channel. There exists a sequence of Pauli-channel Choi matrices $N^{(t)}$ such that

$$\sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)} N^{(t)}]) - \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)} C(\mathcal{P})]) = \mathcal{O}(L\sqrt{nT}).$$

Algorithm Multiplicative weights update (MWU) algorithm

[Arora et al., Theory of Computing **8**(6), 121 (2012)]

Require: $\eta \in (0, \frac{1}{2}]$; $T \in \mathbb{N}$ rounds; $d \in \mathbb{N}$ decisions; initial weights $\boldsymbol{w}^{(1)} = (1, 1, \ldots, 1)$.

1: **for** $t = 1, 2, \ldots T$ **do**

2: Output the decision/estimate $\boldsymbol{p}^{(t)} = \frac{\boldsymbol{w}^{(t)}}{\phi^{(t)}}$, $\phi^{(t)} = \sum_{i=1}^{d} w_i^{(t)}$.

3: Receive the cost vector $\boldsymbol{m}^{(t)} = (m_1^{(t)}, m_2^{(t)}, \ldots, m_d^{(t)})$, $m_i^{(t)} \in [-1, 1]$, $i \in \{1, 2, \ldots, d\}$.

4: Update the weights as $w_i^{(t+1)} = w_i^{(t)}(1 - \eta m_i^{(t)})$, $i \in \{1, 2, \ldots, d\}$.

5: Set $\boldsymbol{w}^{(t+1)} = (w_1^{(t+1)}, w_2^{(t+1)}, \ldots, w_d^{(t+1)})$.

6: **end for**

For an arbitrary probability vector $\boldsymbol{q}$:

$$\sum_{t=1}^{T} \boldsymbol{m}^{(t)} \cdot \boldsymbol{p}^{(t)} \leq \sum_{t=1}^{T} \boldsymbol{m}^{(t)} \cdot \boldsymbol{q} + \eta T + \frac{\log d}{\eta}.$$

Proof of the regret upper bound for Pauli channels

Theorem

Let $\mathcal{P}$ be an n-qubit Pauli channel. There exists a sequence of Pauli-channel Choi matrices $N^{(t)}$ such that

$$\sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}N^{(t)}]) - \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}C(\mathcal{P})]) = \mathcal{O}(L\sqrt{nT}).$$

Proof: Map to the MWU algorithm as follows:

$$N^{(t)} := \sum_{\boldsymbol{z},\boldsymbol{x}\in\{0,1\}^n} p^{(t)}_{\boldsymbol{z},\boldsymbol{x}}\Gamma^{\boldsymbol{z},\boldsymbol{x}},$$

$$\boldsymbol{m}^{(t)} := (m^{(t)}_{\boldsymbol{z},\boldsymbol{x}})_{\boldsymbol{z},\boldsymbol{x}\in\{0,1\}^n},$$

$$m^{(t)}_{\boldsymbol{z},\boldsymbol{x}} := \frac{1}{L}\ell'_t(\mathrm{Tr}[E^{(t)}N^{(t)}])\mathrm{Tr}[E^{(t)}\Gamma^{\boldsymbol{z},\boldsymbol{x}}],$$

$$\Gamma^{\boldsymbol{z},\boldsymbol{x}} = (\mathbb{1}\otimes P^{\boldsymbol{z},\boldsymbol{x}})|\Gamma\rangle\langle\Gamma|(\mathbb{1}\otimes P^{\boldsymbol{z},\boldsymbol{x}}),$$

$$|\Gamma\rangle = \sum_{\boldsymbol{\alpha}\in\{0,1\}^n} |\boldsymbol{\alpha},\boldsymbol{\alpha}\rangle.$$

Algorithm Multiplicative weights update (MWU) algorithm

[Arora et al., Theory of Computing **8**(6), 121 (2012)]

Require: $\eta \in (0, \frac{1}{2}]$; $T \in \mathbb{N}$ rounds; $d \in \mathbb{N}$ decisions; initial weights $\boldsymbol{w}^{(1)} = (1, 1, \ldots, 1)$.

1: **for** $t = 1, 2, \ldots T$ **do**

2: Output the decision/estimate $\boldsymbol{p}^{(t)} = \frac{\boldsymbol{w}^{(t)}}{\phi^{(t)}}$, $\phi^{(t)} = \sum_{i=1}^{d} w_i^{(t)}$.

3: Receive the cost vector $\boldsymbol{m}^{(t)} = (m_1^{(t)}, m_2^{(t)}, \ldots, m_d^{(t)})$, $m_i^{(t)} \in [-1, 1]$, $i \in \{1, 2, \ldots, d\}$.

4: Update the weights as $w_i^{(t+1)} = w_i^{(t)}(1 - \eta m_i^{(t)})$, $i \in \{1, 2, \ldots, d\}$.

5: Set $\boldsymbol{w}^{(t+1)} = (w_1^{(t+1)}, w_2^{(t+1)}, \ldots, w_d^{(t+1)})$.

6: **end for**

For an arbitrary probability vector $\boldsymbol{q}$:

$$\sum_{t=1}^{T} \boldsymbol{m}^{(t)} \cdot \boldsymbol{p}^{(t)} \leq \sum_{t=1}^{T} \boldsymbol{m}^{(t)} \cdot \boldsymbol{q} + \eta T + \frac{\log d}{\eta}.$$

$$C(\mathcal{P}) = \sum_{\boldsymbol{z},\boldsymbol{x}\in\{0,1\}^n} q_{\boldsymbol{z},\boldsymbol{x}}\Gamma^{\boldsymbol{z},\boldsymbol{x}}.$$

Proof of the regret upper bound for Pauli channels

Theorem

Let $\mathcal{P}$ be an n-qubit Pauli channel. There exists a sequence of Pauli-channel Choi matrices $N^{(t)}$ such that

$$\sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}N^{(t)}]) - \sum_{t=1}^{T} \ell_t(\mathrm{Tr}[E^{(t)}C(\mathcal{P})]) = \mathcal{O}(L\sqrt{nT}).$$

$$N^{(t)} := \sum_{\boldsymbol{z},\boldsymbol{x}\in\{0,1\}^n} p^{(t)}_{\boldsymbol{z},\boldsymbol{x}}\Gamma^{\boldsymbol{z},\boldsymbol{x}},$$

$$\boldsymbol{m}^{(t)} := (m^{(t)}_{\boldsymbol{z},\boldsymbol{x}})_{\boldsymbol{z},\boldsymbol{x}\in\{0,1\}^n},$$

$$m^{(t)}_{\boldsymbol{z},\boldsymbol{x}} := \frac{1}{L}\ell'_t(\mathrm{Tr}[E^{(t)}N^{(t)}])\mathrm{Tr}[E^{(t)}\Gamma^{\boldsymbol{z},\boldsymbol{x}}],$$

$$C(\mathcal{P}) = \sum_{\boldsymbol{z},\boldsymbol{x}\in\{0,1\}^n} q_{\boldsymbol{z},\boldsymbol{x}}\Gamma^{\boldsymbol{z},\boldsymbol{x}},$$

$$\sum_{t=1}^{T} \boldsymbol{m}^{(t)}\cdot\boldsymbol{p}^{(t)} \leq \sum_{t=1}^{T} \boldsymbol{m}^{(t)}\cdot\boldsymbol{q} + \eta T + \frac{\log d}{\eta}.$$

Lemma: For ℓ_t convex and differentiable,

$$\sum_{t=1}^{T}\Big(\ell_t(f_t(x_t)) - \ell_t(f(x_t))\Big) \leq \sum_{t=1}^{T}\Big(\ell'_t(f_t(x_t))(f_t(x_t) - f(x_t))\Big).$$

Recall: $f_t(x_t) \equiv \mathrm{Tr}[E^{(t)}N^{(t)}]$. Then,

$$\frac{1}{L}\ell'_t(\mathrm{Tr}[E^{(t)}_{A,B}N^{(t)}_{A,B}])\mathrm{Tr}[E^{(t)}_{A,B}C^{\mathcal{P}}_{A,B}] = \boldsymbol{m}^{(t)}\cdot\boldsymbol{q},$$

$$\frac{1}{L}\ell'_t(\mathrm{Tr}[E^{(t)}_{A,B}N^{(t)}_{A,B}])\mathrm{Tr}[E^{(t)}_{A,B}N^{(t)}_{A,B}] = \boldsymbol{m}^{(t)}\cdot\boldsymbol{p}^{(t)},$$

Therefore, setting $\eta = \sqrt{n/T}$:

$$\sum_{t=1}^{T}\Big(\ell_t(\mathrm{Tr}[E^{(t)}_{A,B}N^{(t)}_{A,B}]) - \ell_t(\mathrm{Tr}[E^{(t)}_{A,B}C^{\mathcal{P}}_{A,B}])\Big) \leq L\left(\eta T + \frac{n\log 4}{\eta}\right)$$
$$= \mathcal{O}(L\sqrt{nT}).$$

Summary and outlook

- We consider learning channel expectation values in the online setting: observables can be provided in a sequential, adaptive, and adversarial manner.
- For general channels, exponential mistake and regret bounds.
- For special classes of channels, we get good mistake and regret bounds (e.g., Pauli channels).
- **Open question:** MMW-style algorithm for quantum channels? (Like a max-entropy principle for quantum channels.)

Thanks!