Follow the Leader with Dropout Perturbations

Tim van Erven

COLT, 2014

Joint work with: Wojciech Kotłowski
Manfred Warmuth
Neural Network

\[
\begin{align*}
  y &= f(x_1, x_2, x_3) \\
  x_1, x_2, x_3 &= \text{input variables} \\
  y &= \text{output variable}
\end{align*}
\]
Neural Network
Dropout Training

- Stochastic gradient descent
- Randomly remove every hidden/input unit with probability $1/2$ before each gradient descent update

[Hinton et al., 2012]
Dropout Training

- Very successful in e.g. image classification, speech recognition
- Many people trying to analyse why it works

[Wager, Wang, Liang, 2013]
Prediction with Expert Advice

- Every round $t = 1, \ldots, T$:
  1. (Randomly) choose expert $\hat{k}_t \in \{1, \ldots, K\}$
  2. Observe expert losses $l_{t,1}, \ldots, l_{t,K} \in [0, 1]$
  3. Our loss is $l_{t,\hat{k}_t}$

Goal: minimize expected regret

$$R_T = \sum_{t=1}^{T} \mathbb{E}[l_{t,\hat{k}_t}] - L^*$$ where

$$L^* = \min_k \sum_{t=1}^{T} l_{t,k}$$
Follow-the-Leader

- Deterministically choose the expert that has predicted best in the past:

\[
\hat{k}_t = \arg \min_k \sum_{s=1}^{t-1} \ell_{t,k} \quad \text{is the leader.}
\]

- Can be fooled: regret grows linearly in T for adversarial data
Dropout Perturbations

\[ \mathbb{E}[\ell_{t, \hat{k}_t}] \]

\[ \ell_{t, 1} \quad \ell_{t, 2} \quad \ell_{t, 3} \quad \ell_{t, 4} \]

\[ \tilde{\ell}_{t, k} = \begin{cases} 
\ell_{t, k} & \text{with probability } 1 - \alpha \\
0 & \text{with probability } \alpha 
\end{cases} \]

\[ \hat{k}_t = \arg \min_k \sum_{s=1}^{t-1} \tilde{\ell}_{t, k} \text{ is the perturbed leader} \]
Dropout Perturbations for Binary Losses

- For losses in \( \{0, 1\} \) it works: for any dropout probability \( \alpha \in (0, 1) \)

\[
\mathcal{R}_T = O \left( \sqrt{L^* \ln K + \ln K} \right)
\]

- No tuning required!
Dropout Perturbations for Binary Losses

- For losses in \( \{0, 1\} \) it works: for any dropout probability \( \alpha \in (0, 1) \)

\[
\mathcal{R}_T = O \left( \sqrt{L^* \ln K} + \ln K \right)
\]

- **No tuning** required!

- But it does **not** work for continuous losses in \([0,1]\): there exist losses such that

\[
\mathcal{R}_T = \Omega(K)
\]
**Binarized Dropout Perturbations: Continuous Losses**

\[ \tilde{\ell}_{t,k} = \begin{cases} 
1 & \text{with probability } (1 - \alpha)\ell_{t,k}, \\
0 & \text{otherwise.} 
\end{cases} \]

- The right generalization: for losses in \([0,1]\)

\[ R_T = O \left( \sqrt{L^* \ln K} + \ln K \right) \]
Small Regret for IID Data

If loss vectors are

- independent, identically distributed between trials,
- with a gap between expected loss of best expert and the rest,

then regret is constant:

$$R_T = O(\ln K) \quad \text{w.h.p.}$$

- Algorithms that rely on doubling trick for $T$ or $L^*$ do not get this.
Instance of Follow-the-Perturbed Leader

- **Follow-the-Perturbed-Leader** [Kalai, Vempala, 2005]:
  \[
  \hat{k}_t = \arg\min_k \sum_{s=1}^{t-1} \ell_{t,k} + \xi_{t-1,k}
  \]

  We have **data-dependent perturbations** $\xi_{t-1,k}$ that **differ between experts**.

- **Standard analysis**: bound probability of leader change in the be-the-leader lemma.

- **Elegant simple bound** for perturbations of Kalai & Vempala, but not for us.
Related Work: RWP

- **Random walk perturbation** [Devroye et al. 2013]:
  \[
  \tilde{\ell}_{t,k} = \ell_{t,k} + Z_{t,k}
  \]
  for \( Z_{t,k} \) a centered Bernoulli variable

  \[
  \mathcal{R}_T = O(\sqrt{T \ln K})
  \]

- Equivalent to dropout if \( \ell_{t,k} = 1 \)
- But perturbations do not adapt to data, so no \( L^* \)-bound
Proof Outline

- Find **worst-case loss sequence**
Proof Outline

- Find **worst-case loss sequence**: e.g. for 3 experts with cumulative losses 1, 3 and 5

\[
\begin{align*}
(1, 0, 0), & \quad (0, 1, 0) & \quad (0, 1, 1) & \quad (0, 0, 1) & \quad (0, 0, 0) \\
(0, 0, 0), & \quad (1, 0, 1) & \quad (0, 1, 0) & \quad (0, 0, 1) & \quad (0, 1, 1) \\
(0, 1, 0), & \quad (1, 0, 1) & \quad (0, 1, 0) & \quad (0, 0, 1) & \quad (1, 1, 1) \\
\end{align*}
\]

- All experts get losses
- Expert 1 reached budget
- Experts 1 and 2 reached budget
Proof Outline

- Find **worst-case loss sequence**: e.g. for 3 experts with cumulative losses 1, 3 and 5

\[
\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}
\]

- **all experts get losses**
- **expert 1 reached budget**
- **experts 1 and 2 reached budget**

1. Cumulative losses approximately equal: apply lemma from RWP roughly once per K rounds
2. Expert 1 much smaller cum. loss: Hoeffding
Summary

- Simple algorithm: Follow-the-leader on losses that are perturbed by binarized dropout
- **No tuning** necessary
- On any losses:
  \[ R_T = O \left( \sqrt{L^* \ln K} + \ln K \right) \]
- On i.i.d. loss vectors with gap between best expert and rest:
  \[ R_T = O(\ln K) \text{ w.h.p.} \]
Many Open Questions

To discuss at the poster!

• Can we use dropout for:
  – Tracking the best expert?
  – *Combinatorial settings* (e.g. online shortest path)?

• Need to reuse randomness between experts

• What about variations on the dropout perturbations?
  – Drop the whole loss vector at once?
References