The Forgetron: 
A Kernel-Based Perceptron on a Fixed Budget

Ofer Dekel    Shai Shalev-Shwartz    Yoram Singer
School of Computer Science & Engineering
The Hebrew University, Jerusalem 91904, Israel
{oferd,shais,singer}@cs.huji.ac.il

• Presented by: Mourad GUEZZOU
Contents

• Introduction
• Problem setting
• Forgetron algorithm
• Experiments
• Conclusion
Introduction

• The online classifications algorithms store a subset of observed example in its internal memory
• It continually changes as learning progresses (new hypothesis are added)
• A rapid growth of active set + Bounded memory → Risk to require more memory than physically available
• Problem specially eminent in cases where the online Algorithm is implemented in hardware with small memory such as mobile telephone

• FORGETRON: since its update builds on that of the perceptron and since it gradually FORGETs active example as learning progresses
Problem Setting

**Online learning:**
- Choose an initial hypothesis $f_0$
- For $t=1,2,...$
  - Receive an instance $x_t$ and predict $\text{sign}(f_t(x_t))$ determined by a hypothesis, stored in internal memory and updated from round to round
  - If $(y_t f_t(x_t) \leq 0)$ ($f_t$ denote the hypothesis used in round $t$)
    - Update the hypothesis $f$

**Goal:** minimize the number of prediction mistakes

**Kernel-based hypotheses**

$$f^I_t(x) = \sum_{i \in I_t} \sigma_{i,t} y_i K(x_i, x)$$

**Example: the dual Perceptron**
- $\sigma_{i,t}$ is always 1
- Initial hypothesis: $I_1 = \emptyset$
- Update rule: $I_{t+1} = I_t \cup \{t\}$

<table>
<thead>
<tr>
<th>K</th>
<th>kernel Operator</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>subset of ${1,...,(t-1)}$</td>
</tr>
<tr>
<td>$X_i$</td>
<td>is active on round $t$ if $i$ in $I_t$</td>
</tr>
<tr>
<td>$Y_i$</td>
<td>in ${-1, +1}$</td>
</tr>
<tr>
<td>B</td>
<td>positif integer, refer budget parameter</td>
</tr>
</tbody>
</table>
The Forgetron

- Initialize: $I_1 = \emptyset$; $Q_1 = 0$; $M_0 = 0$ with $M$ number of mistakes

- For $t=1,2$...

  define $f_t(x) = \sum_{i \in I_t} \sigma_{i,t} y_i K(x_i, x)$

Receive an instance $x_t$, predict $\text{sign}(f_t(x_t))$, and then receive $y_t$

If $y_t f_t(x_t) \leq 0$ set $M_t = M_{t-1} + 1$ and update
The Forgetron

Step (1) - Perceptron

\[ I'_t = I_t \cup \{t\} \]
\[ \text{define} \quad f'_t = f_t + y_t K(x_t, \cdot) \]

- If \(|I'_t| \leq B\) skip the next two steps
- \[ \text{define } r_t = \min I_t \]
The Forgetron

Step (2) - Shrinking

\[ \phi_t = \max\{\phi \in (0, 1] : \Psi(\phi, \sigma_{i,t}, \mu_t) + Q_t \leq (15/32) M_t\} \]

\[ \forall i \in I_t', \quad \sigma_{i,t+1} = \phi_t \sigma_{i,t} \]

define \[ f''_t = \phi_t f'_t \]
The Forgetron

Step (3) - Removal

\[ I_{t+1} = I_t \setminus \{r_t\} \]
Experiments

Note that the Forgetron outperforms CKS on both datasets, especially when the value of $B$ is small.
Conclusion

• Describe the FORGETRON algorithm which is kernel-based online learning with a fixed memory budget

• The analysis presented in this paper can be used to derive a family of online algorithms of which the Forgetron is only one special case.