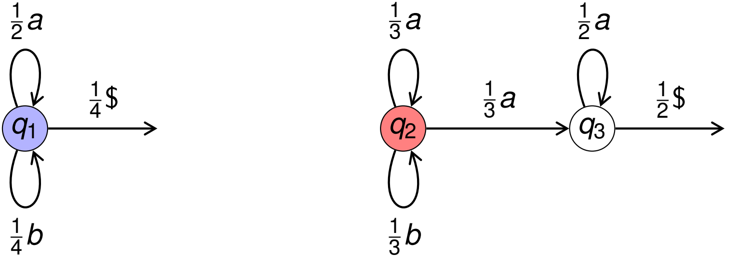
On Computing the Total Variation Distance of Hidden Markov Models

Stefan Kiefer

University of Oxford, UK

ICALP 2018
Prague, 10 July 2018

Hidden Markov Models = Labelled Markov Chains



$\Pr_1(aa) = \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{4}$ $\qquad$ $\Pr_2(aa) = \frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{2} + \frac{1}{3} \cdot \frac{1}{2} \cdot \frac{1}{2}$

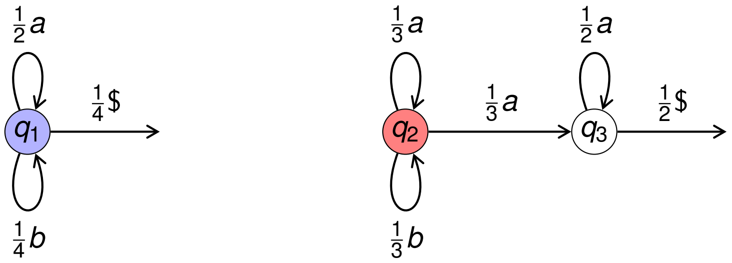
Each Labelled Markov Chain (LMC) generates a probability distribution over Σ^*.

Hidden Markov Models = Labelled Markov Chains

Very widely used:

- speech recognition
- gesture recognition
- signal processing
- climate modelling
- computational biology
  - DNA modelling
  - biological sequence analysis
  - structure prediction
- probabilistic model checking: see tools like Prism or Storm

Hidden Markov Models = Labelled Markov Chains



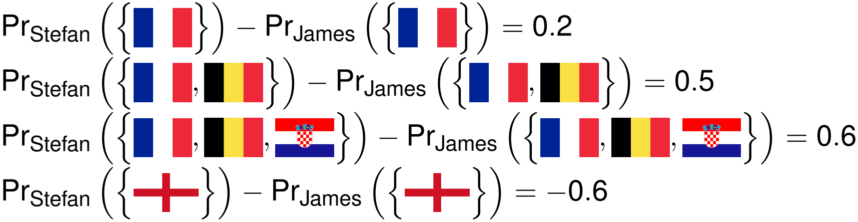
$\Pr_1(aa) = \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{4}$ $\Pr_2(aa) = \frac{1}{3} \cdot \frac{1}{3} \cdot \frac{1}{2} + \frac{1}{3} \cdot \frac{1}{2} \cdot \frac{1}{2}$

Each LMC generates a probability distribution over Σ^*.

Equivalence problem:
Are the two distributions equal?

Solvable in $O(|Q|^3|\Sigma|)$ with linear algebra [Schützenberger'61].
Direct applications in the verification of anonymity properties.

$\Pr_{\text{James}}$	0.1	0.1	0.8	0.0
$\Pr_{\text{Stefan}}$	0.3	0.4	0.2	0.1



Total Variation Distance for Words

Let $\Pr_1, \Pr_2$ be two probability distributions over Σ^*.

$$d(\Pr_1, \Pr_2) \;:=\; \max_{W \subseteq \Sigma^*} \left| \Pr_1(W) - \Pr_2(W) \right|$$

The maximum is attained by
$W_1 := \{ w \in \Sigma^* : \Pr_1(w) \ge \Pr_2(w) \}$.

As in the football case:

$$d(\Pr_1, \Pr_2) \;=\; \frac{1}{2} \sum_{w \in \Sigma^*} \left| \Pr_1(w) - \Pr_2(w) \right|$$

Total Variation Distance for Words

Let $\Pr_1, \Pr_2$ be two probability distributions over Σ^*.

$$d(\Pr_1, \Pr_2) \;:=\; \max_{W \subseteq \Sigma^*} \left| \Pr_1(W) - \Pr_2(W) \right|$$

The maximum is attained by
$W_1 := \{w \in \Sigma^* : \Pr_1(w) \ge \Pr_2(w)\}$.

As in the football case:

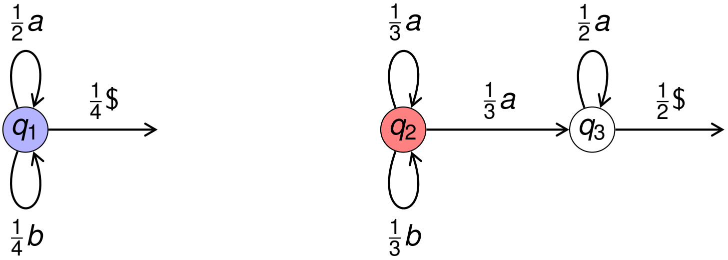
$$d(\Pr_1, \Pr_2) \;=\; \frac{1}{2} \sum_{w \in \Sigma^*} \left| \Pr_1(w) - \Pr_2(w) \right|$$

By a simple calculation:

$$1 + d(\Pr_1, \Pr_2) \;=\; \Pr_1(W_1) + \Pr_2(W_2)$$

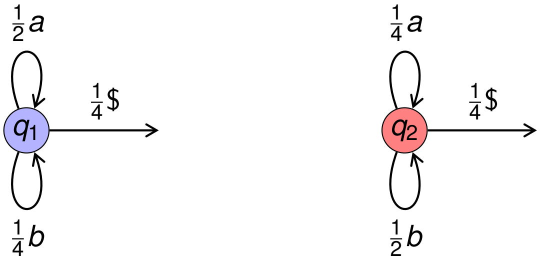
for $W_2 := \{w \in \Sigma^* : \Pr_1(w) < \Pr_2(w)\}$.

Verification View



$$\forall \varphi \ : \ \mathrm{Pr}_2(\varphi) \in [\mathrm{Pr}_1(\varphi) - d, \mathrm{Pr}_1(\varphi) + d]$$

Small distance saves verification work.
Especially for parameterised models.



$$d = \frac{\sqrt{2}}{4} \approx 0.35$$

Given two LMCs and a threshold $\tau \in [0, 1]$.
Is $d > \tau$? strict distance-threshold problem
Is $d \geq \tau$? non-strict distance-threshold problem

NP-hard: [Lyngsø,Pedersen'02], [Cortes,Mohri,Rastogi'07], [Chen,K.'14]

Decidability of the Distance-Threshold Problem

Theorem (K.'18)

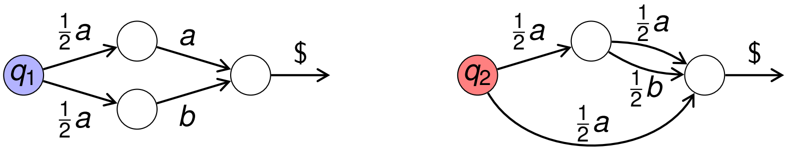
The strict distance-threshold problem is undecidable.

Reduction from emptiness of probabilistic automata.

What about the non-strict distance-threshold problem?
It is sqrt-sum-hard [Chen,K.'14] and PP-hard [K.'18].

Decidability status "strict vs. non-strict" similar as for the joint spectral radius of a set of matrices.

Acyclic LMCs



Theorem (K.'18)

For acyclic LMCs:

- *Computing the distance is #P-complete.*
- *Approximating the distance is #P-complete.*
- *The strict and non-strict distance-threshold problems are PP-complete.*

Reduction from #NFA:

Given an NFA $\mathcal{A}$ and $n \in \mathbb{N}$ in unary, compute $|L(\mathcal{A}) \cap \Sigma^n|$.

Probably simpler than previous NP-hardness reductions.

Approximation

Theorem (K.'18)

Given two LMCs and an error bound $\varepsilon > 0$ in binary, one can compute in PSPACE a number $x \in [d - \varepsilon, d + \varepsilon]$.

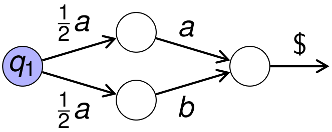
$$\begin{aligned} 1 + d(\Pr_1, \Pr_2) &= \Pr_1(W_1) + \Pr_2(W_2) \qquad \text{where} \\ W_1 &= \{w \in \Sigma^* : \Pr_1(w) \geq \Pr_2(w)\} \\ W_2 &= \{w \in \Sigma^* : \Pr_1(w) < \Pr_2(w)\} \end{aligned}$$

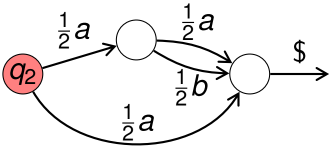
Approximation

Theorem (K.'18)

Given two LMCs and an error bound $\varepsilon > 0$ in binary, one can compute in PSPACE a number $x \in [d - \varepsilon, d + \varepsilon]$.

$$
\begin{aligned}
1 + d(\Pr_1, \Pr_2) &= \Pr_1(W_1) + \Pr_2(W_2) \qquad \text{where} \\
W_1 &= \{w \in \Sigma^* : \Pr_1(w) \geq \Pr_2(w)\} \\
W_2 &= \{w \in \Sigma^* : \Pr_1(w) < \Pr_2(w)\}
\end{aligned}
$$

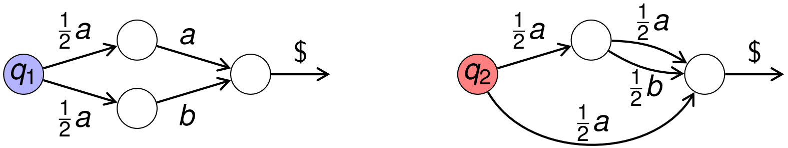




Theorem (K.'18)

Given two LMCs and an error bound $\varepsilon > 0$ in binary, one can compute in PSPACE a number $x \in [d - \varepsilon, d + \varepsilon]$.

$$
\begin{aligned}
1 + d(\Pr_1, \Pr_2) &= \Pr_1(W_1) + \Pr_2(W_2) \qquad \text{where} \\
W_1 &= \{w \in \Sigma^* : \Pr_1(w) \geq \Pr_2(w)\} \\
W_2 &= \{w \in \Sigma^* : \Pr_1(w) < \Pr_2(w)\}
\end{aligned}
$$



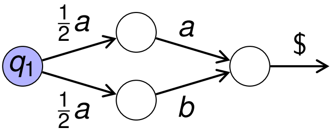
In the cyclic case: we have to sample exponentially long words.

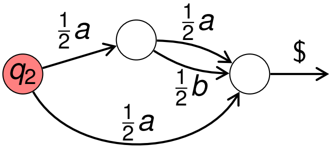
Approximation

Theorem (K.'18)

Given two LMCs and an error bound $\varepsilon > 0$ in binary, one can compute in PSPACE a number $x \in [d - \varepsilon, d + \varepsilon]$.

$$\begin{aligned} 1 + d(\Pr_1, \Pr_2) &= \Pr_1(W_1) + \Pr_2(W_2) \qquad \text{where} \\ W_1 &= \{w \in \Sigma^* : \Pr_1(w) \geq \Pr_2(w)\} \\ W_2 &= \{w \in \Sigma^* : \Pr_1(w) < \Pr_2(w)\} \end{aligned}$$





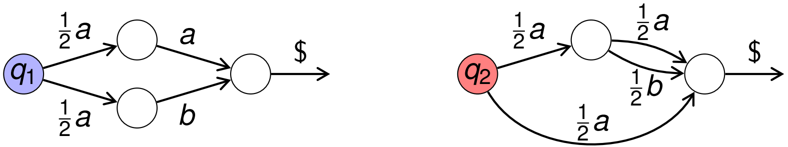
In the cyclic case: we have to sample exponentially long words.
Floating-point arithmetic computes $\Pr_1(w), \Pr_2(w)$ up to small relative error.

Approximation

Theorem (K.'18)

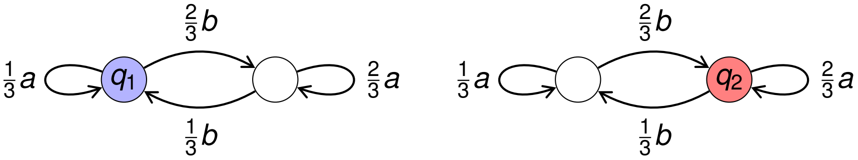
Given two LMCs and an error bound $\varepsilon > 0$ in binary, one can compute in PSPACE a number $x \in [d - \varepsilon, d + \varepsilon]$.

$$
\begin{aligned}
1 + d(\mathrm{Pr}_1, \mathrm{Pr}_2) &= \mathrm{Pr}_1(W_1) + \mathrm{Pr}_2(W_2) \qquad \text{where} \\
W_1 &= \{w \in \Sigma^* : \mathrm{Pr}_1(w) \geq \mathrm{Pr}_2(w)\} \\
W_2 &= \{w \in \Sigma^* : \mathrm{Pr}_1(w) < \mathrm{Pr}_2(w)\}
\end{aligned}
$$



In the cyclic case: we have to sample exponentially long words.
Floating-point arithmetic computes $\mathrm{Pr}_1(w), \mathrm{Pr}_2(w)$ up to small relative error.
Use Ladner's result on counting in polynomial space.

Infinite-Word LMCs



E.g., if $W = \{aw : w \in \Sigma^\omega\}$ then $\Pr_1(W) = \frac{1}{3}$ and $\Pr_2(W) = \frac{2}{3}$.

$$\begin{aligned} d(\Pr_1, \Pr_2) &:= \max_{W \subseteq \Sigma^\omega} \left| \Pr_1(W) - \Pr_2(W) \right| \\ &= \max_{W \subseteq \Sigma^\omega} \left(\Pr_1(W) - \Pr_2(W) \right) \end{aligned}$$

Theorem (Chen,K.'14)

One can decide in polynomial time if $d(\Pr_1, \Pr_2) = 1$.

One can also decide in polynomial time if $\Pr_1 = \Pr_2$.
Finite-word LMCs are a special case of infinite-word LMCs.

Summary

Theorem (main results again)

The strict distance-threshold problem is undecidable.
Approximating the distance is #P-hard and in PSPACE.

Open problems:

- decidability of the non-strict distance-threshold problem
- complexity of approximating the distance of
  - infinite-word LMCs
  - non-hidden/deterministic LMCs