From ranking to intransitive preference learning: rock-paper-scissors and beyond

Tapio Pahikkala[1] Willem Waegeman[2] Evgeni Tsivtsivadze[3]
Tapio Salakoski[1] Bernard De Baets[2]

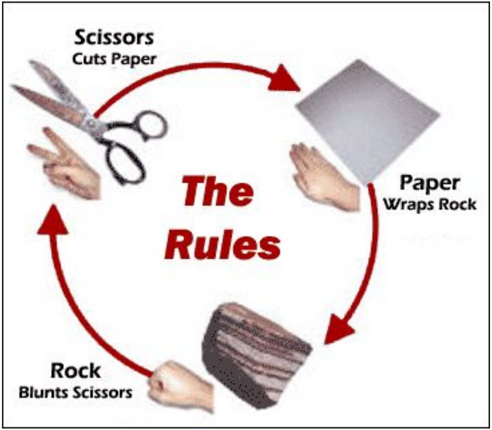
[1]TUCS, University of Turku, Finland

[2]KERMIT, Ghent University, Belgium

[3]Institute for Computing and Information Sciences
Radboud University Nijmegen, The Netherlands

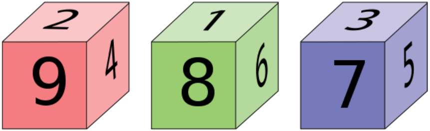
Outline

The transitivity property: a classical example



Examples of intransitivity are found in many fields...

Outline

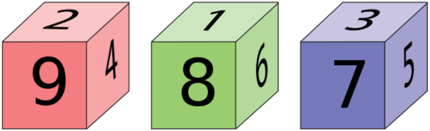


$$Q(\mathbf{x},\mathbf{x}') = 5/9 \quad Q(\mathbf{x}',\mathbf{x}'') = 5/9 \quad Q(\mathbf{x}'',\mathbf{x}) = 5/9$$
$$Q(\mathbf{x}',\mathbf{x}) = 4/9 \quad Q(\mathbf{x}'',\mathbf{x}') = 4/9 \quad Q(\mathbf{x},\mathbf{x}'') = 4/9$$

Proposition

A relation $Q : \mathcal{X}^2 \rightarrow [0,1]$ *is called a reciprocal relation if*

$$Q(\mathbf{x},\mathbf{x}') + Q(\mathbf{x}',\mathbf{x}) = 1 \quad \forall(\mathbf{x},\mathbf{x}') \in \mathcal{X}^2 .$$



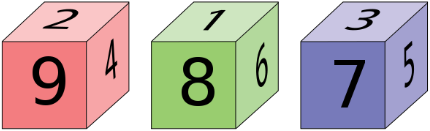
$$Q(\mathbf{x}, \mathbf{x}') = 5/9 \quad Q(\mathbf{x}', \mathbf{x}'') = 5/9 \quad Q(\mathbf{x}'', \mathbf{x}) = 5/9$$

$$Q(\mathbf{x}', \mathbf{x}) = 4/9 \quad Q(\mathbf{x}'', \mathbf{x}') = 4/9 \quad Q(\mathbf{x}, \mathbf{x}'') = 4/9$$

Proposition

A relation $Q : \mathcal{X}^2 \rightarrow [0, 1]$ is called a reciprocal relation if

$$Q(\mathbf{x}, \mathbf{x}') + Q(\mathbf{x}', \mathbf{x}) = 1 \quad \forall (\mathbf{x}, \mathbf{x}') \in \mathcal{X}^2 .$$

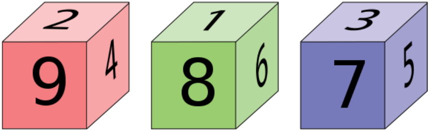


$$Q(\mathbf{x}, \mathbf{x}') = 5/9 \quad Q(\mathbf{x}', \mathbf{x}'') = 5/9 \quad Q(\mathbf{x}'', \mathbf{x}) = 5/9$$
$$Q(\mathbf{x}', \mathbf{x}) = 4/9 \quad Q(\mathbf{x}'', \mathbf{x}') = 4/9 \quad Q(\mathbf{x}, \mathbf{x}'') = 4/9$$

Proposition

A relation $Q : \mathcal{X}^2 \to [0, 1]$ is called a reciprocal relation if

$$Q(\mathbf{x}, \mathbf{x}') + Q(\mathbf{x}', \mathbf{x}) = 1 \quad \forall (\mathbf{x}, \mathbf{x}') \in \mathcal{X}^2 .$$



$$Q(\mathbf{x},\mathbf{x}') = 5/9 \quad Q(\mathbf{x}',\mathbf{x}'') = 5/9 \quad Q(\mathbf{x}'',\mathbf{x}) = 5/9$$
$$Q(\mathbf{x}',\mathbf{x}) = 4/9 \quad Q(\mathbf{x}'',\mathbf{x}') = 4/9 \quad Q(\mathbf{x},\mathbf{x}'') = 4/9$$

Proposition

A relation $Q : \mathcal{X}^2 \to [0,1]$ *is called a reciprocal relation if*

$$Q(\mathbf{x},\mathbf{x}') + Q(\mathbf{x}',\mathbf{x}) = 1 \quad \forall(\mathbf{x},\mathbf{x}') \in \mathcal{X}^2 .$$

Ranking representability

Definition

A reciprocal relation $Q : \mathcal{X}^2 \to [0,1]$ is called weakly ranking representable if there exists a ranking function $f : \mathcal{X} \to \mathbb{R}$ such that for any $(\mathbf{x}, \mathbf{x}') \in \mathcal{X}^2$ it holds that

$$Q(\mathbf{x}, \mathbf{x}') \leq \frac{1}{2} \Leftrightarrow f(\mathbf{x}) \leq f(\mathbf{x}')\,.$$

$$Q(\mathbf{x}, \mathbf{x}') = 5/9 \Leftrightarrow \mathbf{x} \succ \mathbf{x}'$$
$$Q(\mathbf{x}', \mathbf{x}'') = 5/9 \Leftrightarrow \mathbf{x}' \succ \mathbf{x}''$$
$$Q(\mathbf{x}'', \mathbf{x}) = 5/9 \Leftrightarrow \mathbf{x}'' \succ \mathbf{x}$$

Ranking representability

Definition

A reciprocal relation $Q : \mathcal{X}^2 \to [0,1]$ is called weakly ranking representable if there exists a ranking function $f : \mathcal{X} \to \mathbb{R}$ such that for any $(\mathbf{x}, \mathbf{x}') \in \mathcal{X}^2$ it holds that

$$Q(\mathbf{x}, \mathbf{x}') \leq \frac{1}{2} \Leftrightarrow f(\mathbf{x}) \leq f(\mathbf{x}')\,.$$

$$Q(\mathbf{x}, \mathbf{x}') = 5/9 \Leftrightarrow \mathbf{x} \succ \mathbf{x}'$$
$$Q(\mathbf{x}', \mathbf{x}'') = 5/9 \Leftrightarrow \mathbf{x}' \succ \mathbf{x}''$$
$$Q(\mathbf{x}'', \mathbf{x}) = 5/9 \Leftrightarrow \mathbf{x}'' \succ \mathbf{x}$$

Weak stochastic transitivity

Proposition (Luce and Suppes, 1965)

A reciprocal relation Q is weakly ranking representable if and only if it satisfies weak stochastic transitivity, i.e., for any $(\mathbf{x}, \mathbf{x}', \mathbf{x}'') \in \mathcal{X}^3$ *it holds that*

$$Q(\mathbf{x}, \mathbf{x}') \geq 1/2 \wedge Q(\mathbf{x}', \mathbf{x}'') \geq 1/2 \Rightarrow Q(\mathbf{x}, \mathbf{x}'') \geq 1/2\,.$$

$$Q(\mathbf{x}, \mathbf{x}') = 6/9 \quad Q(\mathbf{x}', \mathbf{x}'') = 5/9 \quad Q(\mathbf{x}'', \mathbf{x}) = 2/9$$

$$\Leftrightarrow$$

$$\mathbf{x} \succ \mathbf{x}' \succ \mathbf{x}''$$

Weak stochastic transitivity

Proposition (Luce and Suppes, 1965)

A reciprocal relation Q is weakly ranking representable if and only if it satisfies weak stochastic transitivity, i.e., for any $(\mathbf{x}, \mathbf{x}', \mathbf{x}'') \in \mathcal{X}^3$ *it holds that*

$$Q(\mathbf{x}, \mathbf{x}') \geq 1/2 \wedge Q(\mathbf{x}', \mathbf{x}'') \geq 1/2 \Rightarrow Q(\mathbf{x}, \mathbf{x}'') \geq 1/2\,.$$

$$Q(\mathbf{x}, \mathbf{x}') = 6/9 \quad Q(\mathbf{x}', \mathbf{x}'') = 5/9 \quad Q(\mathbf{x}'', \mathbf{x}) = 2/9$$

$$\Leftrightarrow$$

$$\mathbf{x} \succ \mathbf{x}' \succ \mathbf{x}''$$

Outline

Definition of our framework

- Training data $E = (\mathbf{e}_i, y_i)_{i=1}^{N}$
- Training data are here couples: $\mathbf{e} = (\mathbf{x}, \mathbf{x}')$
- Labels $y_i = 2Q(\mathbf{x}_i, \mathbf{x}'_i) + 1$
- Minimizing the regularized empirical error:

$$\mathcal{A}(E) = \operatorname*{argmin}_{h \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^{N} L(h(\mathbf{e}_i), y_i) + \lambda \|h\|_{\mathcal{F}}^2$$

- Least-squares loss function: **regularized least-squares**

Reciprocal relations are learned by defining a specific kernel construction

Consider the following joint feature representation for a couple:

$$\Phi(\mathbf{e}_i) = \Phi(\mathbf{x}_i, \mathbf{x}'_i) = \Psi(\mathbf{x}_i, \mathbf{x}'_i) - \Psi(\mathbf{x}'_i, \mathbf{x}_i),$$

This yields the following kernel defined on couples:

$$\begin{aligned} K^{\Phi}(\mathbf{e}_i, \mathbf{e}_j) &= K^{\Phi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}_j, \mathbf{x}'_j) \\ &= \langle \Psi(\mathbf{x}_i, \mathbf{x}'_i) - \Psi(\mathbf{x}'_i, \mathbf{x}_i), \Psi(\mathbf{x}_j, \mathbf{x}'_j) - \Psi(\mathbf{x}'_j, \mathbf{x}_j) \rangle \\ &= K^{\Psi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}_j, \mathbf{x}'_j) + K^{\Psi}(\mathbf{x}'_i, \mathbf{x}_i, \mathbf{x}'_j, \mathbf{x}_j) \\ &\quad - K^{\Psi}(\mathbf{x}'_i, \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}'_j) - K^{\Psi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}'_j, \mathbf{x}_j) \,. \end{aligned}$$

And the model becomes:

$$h(\mathbf{x}, \mathbf{x}') = \langle \mathbf{w}, \Psi(\mathbf{x}, \mathbf{x}') - \Psi(\mathbf{x}', \mathbf{x}) \rangle = \sum_{i=1}^{N} a_i K^{\Phi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}, \mathbf{x}') \,.$$

Reciprocal relations are learned by defining a specific kernel construction

Consider the following joint feature representation for a couple:

$$\Phi(\mathbf{e}_i) = \Phi(\mathbf{x}_i, \mathbf{x}'_i) = \Psi(\mathbf{x}_i, \mathbf{x}'_i) - \Psi(\mathbf{x}'_i, \mathbf{x}_i),$$

This yields the following kernel defined on couples:

$$\begin{aligned} K^{\Phi}(\mathbf{e}_i, \mathbf{e}_j) &= K^{\Phi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}_j, \mathbf{x}'_j) \\ &= \langle \Psi(\mathbf{x}_i, \mathbf{x}'_i) - \Psi(\mathbf{x}'_i, \mathbf{x}_i), \Psi(\mathbf{x}_j, \mathbf{x}'_j) - \Psi(\mathbf{x}'_j, \mathbf{x}_j) \rangle \\ &= K^{\Psi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}_j, \mathbf{x}'_j) + K^{\Psi}(\mathbf{x}'_i, \mathbf{x}_i, \mathbf{x}'_j, \mathbf{x}_j) \\ &\quad - K^{\Psi}(\mathbf{x}'_i, \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}'_j) - K^{\Psi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}'_j, \mathbf{x}_j)\,. \end{aligned}$$

And the model becomes:

$$h(\mathbf{x}, \mathbf{x}') = \langle \mathbf{w}, \Psi(\mathbf{x}, \mathbf{x}') - \Psi(\mathbf{x}', \mathbf{x}) \rangle = \sum_{i=1}^{N} a_i K^{\Phi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}, \mathbf{x}')\,.$$

Reciprocal relations are learned by defining a specific kernel construction

Consider the following joint feature representation for a couple:

$$\Phi(\mathbf{e}_i) = \Phi(\mathbf{x}_i, \mathbf{x}'_i) = \Psi(\mathbf{x}_i, \mathbf{x}'_i) - \Psi(\mathbf{x}'_i, \mathbf{x}_i),$$

This yields the following kernel defined on couples:

$$\begin{aligned} K^{\Phi}(\mathbf{e}_i, \mathbf{e}_j) &= K^{\Phi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}_j, \mathbf{x}'_j) \\ &= \langle \Psi(\mathbf{x}_i, \mathbf{x}'_i) - \Psi(\mathbf{x}'_i, \mathbf{x}_i), \Psi(\mathbf{x}_j, \mathbf{x}'_j) - \Psi(\mathbf{x}'_j, \mathbf{x}_j) \rangle \\ &= K^{\Psi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}_j, \mathbf{x}'_j) + K^{\Psi}(\mathbf{x}'_i, \mathbf{x}_i, \mathbf{x}'_j, \mathbf{x}_j) \\ &\quad - K^{\Psi}(\mathbf{x}'_i, \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}'_j) - K^{\Psi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}'_j, \mathbf{x}_j) \,. \end{aligned}$$

And the model becomes:

$$h(\mathbf{x}, \mathbf{x}') = \langle \mathbf{w}, \Psi(\mathbf{x}, \mathbf{x}') - \Psi(\mathbf{x}', \mathbf{x}) \rangle = \sum_{i=1}^{N} a_i K^{\Phi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}, \mathbf{x}') \,.$$

Ranking can be considered as a specific case in this framework

Consider the following joint feature representation Ψ for a couple:

$$\Psi_T(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}) .$$

This yields the following kernel K^Ψ:

$$K_T^\Psi(\mathbf{x}_i, \mathbf{x}_i', \mathbf{x}_j, \mathbf{x}_j') = K^\phi(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle ,$$

And the model becomes:

$$h(\mathbf{x}, \mathbf{x}') = \langle \mathbf{w}, \phi(\mathbf{x}) \rangle - \langle \mathbf{w}, \phi(\mathbf{x}') \rangle = f(\mathbf{x}) - f(\mathbf{x}') ,$$

Ranking can be considered as a specific case in this framework

Consider the following joint feature representation Ψ for a couple:

$$\Psi_T(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}) \,.$$

This yields the following kernel K^Ψ:

$$K_T^\Psi(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}_j, \mathbf{x}'_j) = K^\phi(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle \,,$$

And the model becomes:

$$h(\mathbf{x}, \mathbf{x}') = \langle \mathbf{w}, \phi(\mathbf{x}) \rangle - \langle \mathbf{w}, \phi(\mathbf{x}') \rangle = f(\mathbf{x}) - f(\mathbf{x}') \,,$$

Ranking can be considered as a specific case in this framework

Consider the following joint feature representation Ψ for a couple:

$$\Psi_T(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})\,.$$

This yields the following kernel K^Ψ:

$$K^\Psi_T(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}_j, \mathbf{x}'_j) = K^\phi(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle\,,$$

And the model becomes:

$$h(\mathbf{x}, \mathbf{x}') = \langle \mathbf{w}, \phi(\mathbf{x}) \rangle - \langle \mathbf{w}, \phi(\mathbf{x}') \rangle = f(\mathbf{x}) - f(\mathbf{x}')\,,$$

Using the Kronecker-product intransitive relations can be learned, unlike the existing approaches

Consider the following joint feature representation Ψ for a couple:

$$\Psi_I(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}) \otimes \phi(\mathbf{x}'),$$

where $\otimes$ denotes the Kronecker-product:

$$A \otimes B = \begin{pmatrix} A_{1,1}B & \cdots & A_{1,n}B \\ \vdots & \ddots & \vdots \\ A_{m,1}B & \cdots & A_{m,n}B \end{pmatrix},$$

This yields the following kernel K^Ψ:

$$\begin{aligned} K_I^\Psi(\mathbf{x}_i, \mathbf{x}_i', \mathbf{x}_j, \mathbf{x}_j') &= \langle \phi(\mathbf{x}_i) \otimes \phi(\mathbf{x}_i'), \phi(\mathbf{x}_j) \otimes \phi(\mathbf{x}_j') \rangle \\ &= \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle \otimes \langle \phi(\mathbf{x}_i'), \phi(\mathbf{x}_j') \rangle \\ &= K^\phi(\mathbf{x}_i, \mathbf{x}_j) K^\phi(\mathbf{x}_i', \mathbf{x}_j'), \end{aligned}$$

Using the Kronecker-product intransitive relations can be learned, unlike the existing approaches

Consider the following joint feature representation Ψ for a couple:

$$\Psi_I(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}) \otimes \phi(\mathbf{x}')\,,$$

where $\otimes$ denotes the Kronecker-product:

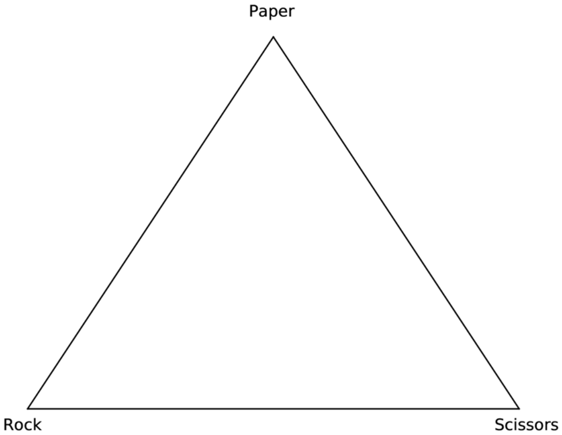
$$A \otimes B = \begin{pmatrix} A_{1,1}B & \cdots & A_{1,n}B \\ \vdots & \ddots & \vdots \\ A_{m,1}B & \cdots & A_{m,n}B \end{pmatrix},$$

This yields the following kernel K^{Ψ}:

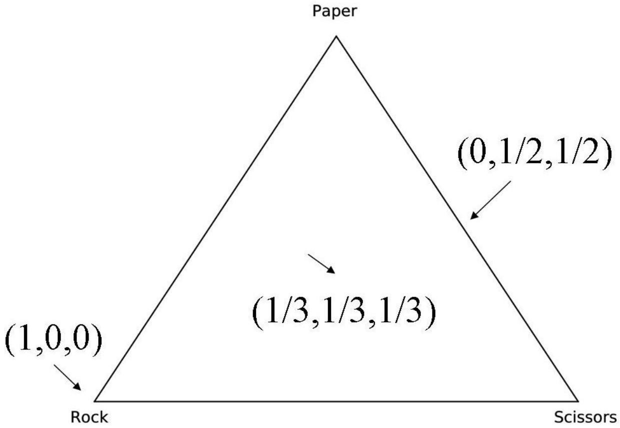
$$\begin{aligned} K_I^{\Psi}(\mathbf{x}_i, \mathbf{x}'_i, \mathbf{x}_j, \mathbf{x}'_j) &= \langle \phi(\mathbf{x}_i) \otimes \phi(\mathbf{x}'_i), \phi(\mathbf{x}_j) \otimes \phi(\mathbf{x}'_j) \rangle \\ &= \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle \otimes \langle \phi(\mathbf{x}'_i), \phi(\mathbf{x}'_j) \rangle \\ &= K^{\phi}(\mathbf{x}_i, \mathbf{x}_j) K^{\phi}(\mathbf{x}'_i, \mathbf{x}'_j), \end{aligned}$$

Outline

Reciprocal relations in rock-paper-scissors



Reciprocal relations in rock-paper-scissors



Reciprocal relations in rock-paper-scissors

Convert probabilities to a reciprocal relation:

$$\begin{aligned} Q(\mathbf{x},\mathbf{x}') &= P(r \mid \mathbf{x})_i P(s \mid \mathbf{x}') + \frac{1}{2} P(r \mid \mathbf{x}) P(r \mid \mathbf{x}') \\ &\quad + P(p \mid \mathbf{x}) P(r \mid \mathbf{x}') + \frac{1}{2} P(p \mid \mathbf{x}) P(p \mid \mathbf{x}') \\ &\quad + P(s \mid \mathbf{x}) P(p \mid \mathbf{x}') + \frac{1}{2} P(s \mid \mathbf{x}) P(s \mid \mathbf{x}'). \end{aligned}$$

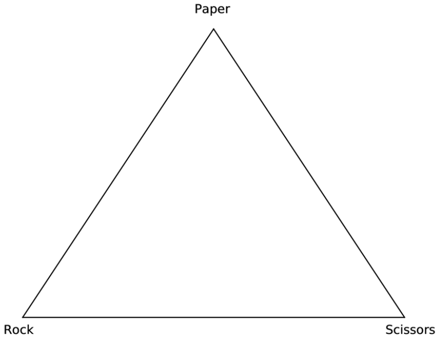
Example:

$$\text{Player1}: \mathbf{x} = (r = 1/2, p = 1/2, s = 0)$$

$$\text{Player2}: \mathbf{x}' = (r = 0, p = 1/2, s = 1/2)$$

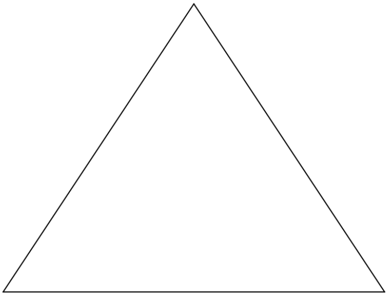
$$\Rightarrow Q(\mathbf{x},\mathbf{x}') = 1/2(1/2 + 0/2) + 1/2(0 + 1/4) + 0(1/2 + 1/4) = 3/8$$

Rock-paper-scissors: experimental setup



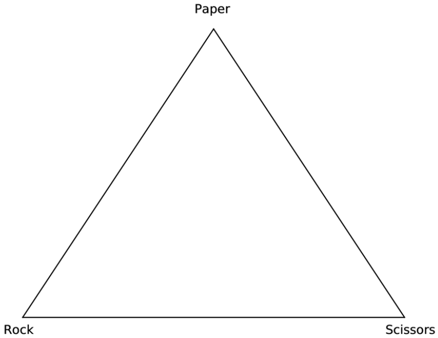
- 100 players for training (100 games)
- 100 players for testing (1000 games)
- features are the mixed strategies
- training labels $y \in \{-1, 0, 1\}$
- test labels $y \in [0, 1]$
- K^{ϕ} linear kernel
- three different settings

Rock-paper-scissors: experimental setup



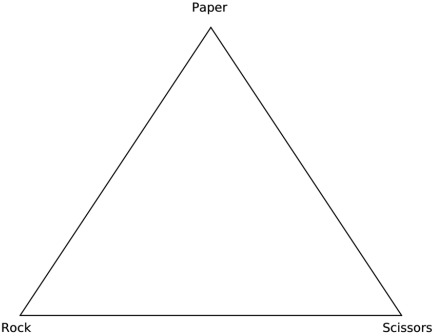
- 100 players for training (100 games)
- 100 players for testing (1000 games)
- features are the mixed strategies
- training labels $y \in \{-1, 0, 1\}$
- test labels $y \in [0, 1]$
- K^ϕ linear kernel
- three different settings

Rock-paper-scissors: experimental setup



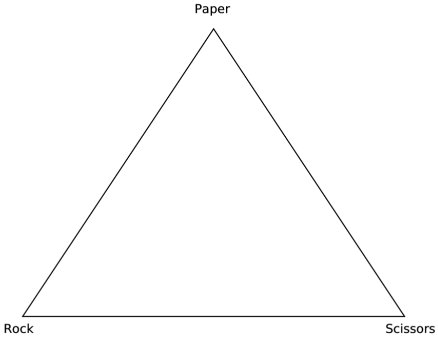
- 100 players for training (100 games)
- 100 players for testing (1000 games)
- features are the mixed strategies
- training labels $y \in \{-1, 0, 1\}$
- test labels $y \in [0, 1]$
- K^{ϕ} linear kernel
- three different settings

Rock-paper-scissors: experimental setup



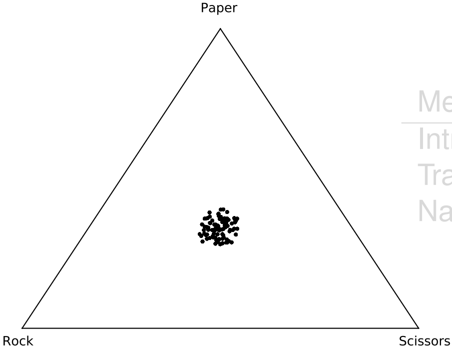
- 100 players for training (100 games)
- 100 players for testing (1000 games)
- features are the mixed strategies
- training labels $y \in \{-1, 0, 1\}$
- test labels $y \in [0, 1]$
- K^{ϕ} linear kernel
- three different settings

Rock-paper-scissors: experimental setup



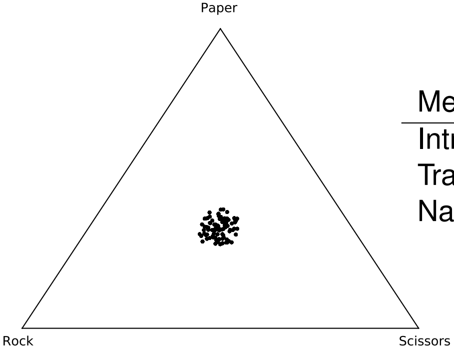
- 100 players for training (100 games)
- 100 players for testing (1000 games)
- features are the mixed strategies
- training labels $y \in \{-1, 0, 1\}$
- test labels $y \in [0, 1]$
- K^{ϕ} linear kernel
- three different settings

Rock-paper-scissors: setting 1



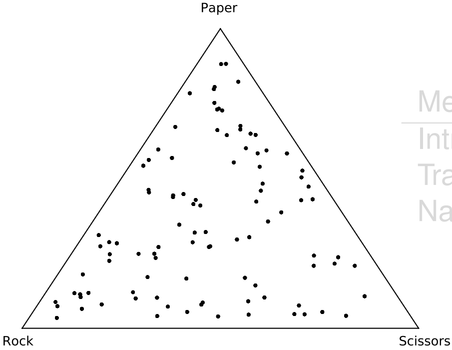
Method	MSE
Intrans.	0.000209
Trans.	0.000162
Naive	0.000001

Rock-paper-scissors: setting 1



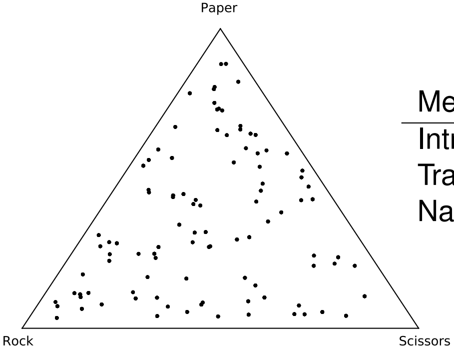
Method	MSE
Intrans.	0.000209
Trans.	0.000162
Naive	0.000001

Rock-paper-scissors: setting 2



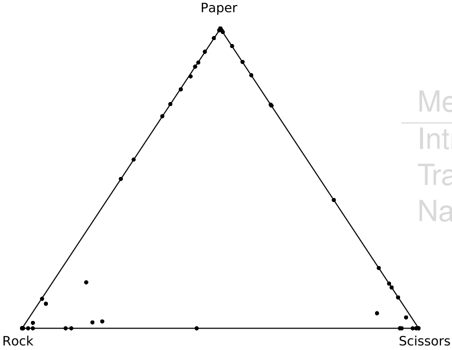
Method	MSE
Intrans.	0.000445
Trans.	0.006804
Naive	0.006454

Rock-paper-scissors: setting 2



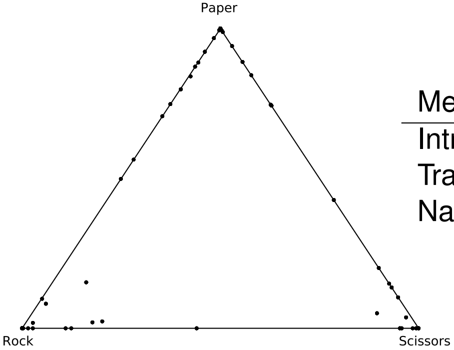
Method	MSE
Intrans.	0.000445
Trans.	0.006804
Naive	0.006454

Rock-paper-scissors: setting 3



Method	MSE
Intrans.	0.000076
Trans.	0.131972
Naive	0.125460

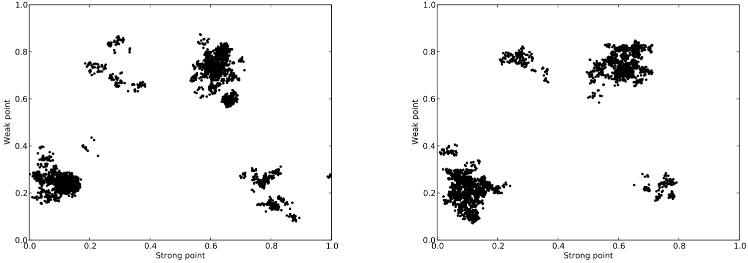
Rock-paper-scissors: setting 3



Method	MSE
Intrans.	0.000076
Trans.	0.131972
Naive	0.125460

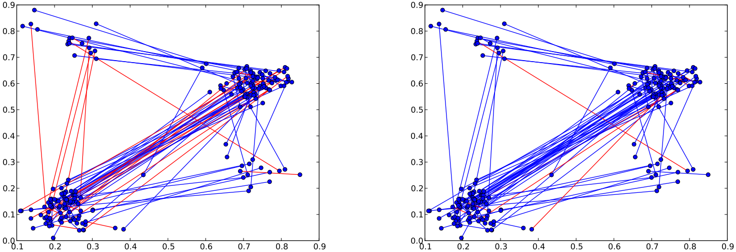
Simulation of competition between species results in stable populations after many iterations

Experiment 2: competition between species in theoretical biology



$$y = \mathrm{sign}(d(s(x'), w(x)) - d(s(x), w(x')))$$

The intransitive kernel clearly beats the traditional transitive kernel



Trans. Accuracy = 0.615 ⇔ Intrans. Accuracy = 0.850

Discussion

- Existing kernel-based ranking methods cannot predict intransitive relations.
- With our framework it is possible to represent and predict intransitive relations in an adequate way.
- Empirical results on two problems confirm that our framework is able to learn intransitive relations, unlike ranking methods.
- Many applications possible (e.g. in the life sciences), but no publicly available datasets.

http://staff.cs.utu.fi/˜ aatapa/software/RPS