A model of random walk with varying transition probabilities

Tomáš Kouřim [1] Petr Volf [2]

[1]Faculty of Nuclear Sciences and Physical Engineering, CTU Prague

[2]Institute of Information Theory and Automation, CAS CR Prague

SMTDA 2020

Random walk

Definition

A man starts from a point O and walks l yards in a straight line; he then turns through any angle whatever and walks another l yards in a second straight line. He repeats this process n times. I require the probability that after these n stretches he is at a distance between r and $r + \delta r$ from his starting point, O.

[Karl Pearson: The problem of the random walk.(1905)]

Where is the *drunken sailor*?

Random walk

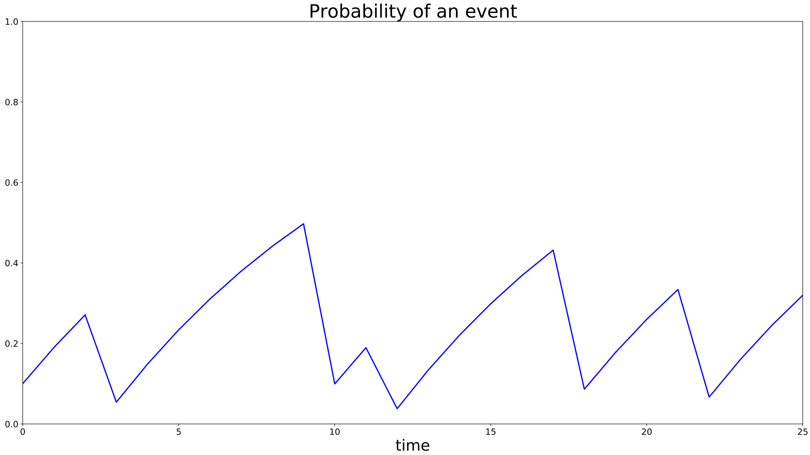
Definition

A man starts from a point O and walks l yards in a straight line; he then turns through any angle whatever and walks another l yards in a second straight line. He repeats this process n times. I require the probability that after these n stretches he is at a distance between r and $r + \delta r$ from his starting point, O.

[Karl Pearson: The problem of the random walk.(1905)]

Where is the *drunken sailor*?

Motivation - Random process with varying probability



Motivation

- Failure of a machine
  - repair after failure
  - preventive maintenance
- Occurrence of a disease
  - cure of the disease
  - prevention (i.e. lifestyle change)
- Development of sports match
  - goal scored, point achieved
  - period won

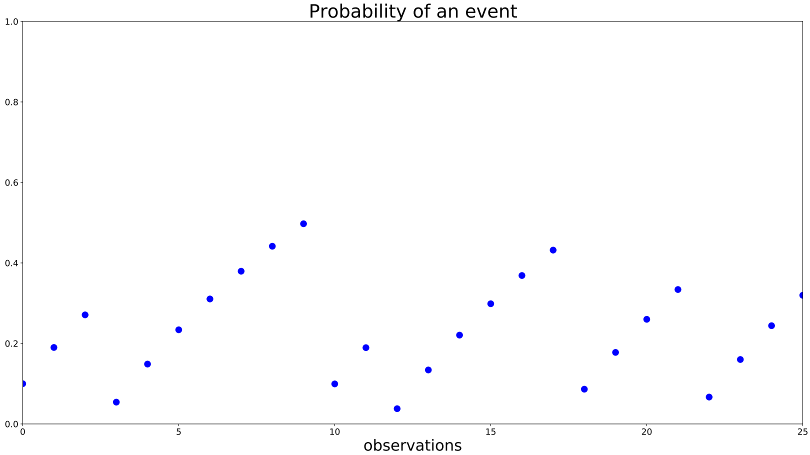
Motivation

- Failure of a machine
  - repair after failure
  - preventive maintenance
- Occurrence of a disease
  - cure of the disease
  - prevention (i.e. lifestyle change)
- Development of sports match
  - goal scored, point achieved
  - period won

Motivation

- Failure of a machine
  - repair after failure
  - preventive maintenance
- Occurrence of a disease
  - cure of the disease
  - prevention (i.e. lifestyle change)
- Development of sports match
  - goal scored, point achieved
  - period won

Motivation - Random process with varying probability



Random walk with varying probabilities

- Random walk with memory based the on standard Bernoulli random walk
- Given starting probability p_0
- $X_t \in \{-1, 1\}$ with $X_t \sim \mathrm{Bernoulli}(p_{t-1})$
- Memory coefficient $\lambda \in (0, 1)$ affecting the development of probabilities p_t as

$$X_t = 1 \rightarrow p_t = \lambda p_{t-1} \qquad X_t = -1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1})$$

$\Rightarrow$ "Success punished"

$$X_t = 1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1}) \qquad X_t = -1 \rightarrow p_t = \lambda p_{t-1}$$

$\Rightarrow$ "Success rewarded"

Random walk with varying probabilities

- Random walk with memory based the on standard Bernoulli random walk
- Given starting probability p_0
- $X_t \in \{-1, 1\}$ with $X_t \sim \mathrm{Bernoulli}(p_{t-1})$
- Memory coefficient $\lambda \in (0, 1)$ affecting the development of probabilities p_t as

$$X_t = 1 \rightarrow p_t = \lambda p_{t-1} \qquad X_t = -1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1})$$

$\Rightarrow$ "Success punished"

$$X_t = 1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1}) \qquad X_t = -1 \rightarrow p_t = \lambda p_{t-1}$$

$\Rightarrow$ "Success rewarded"

Random walk with varying probabilities

- Random walk with memory based the on standard Bernoulli random walk
- Given starting probability p_0
- $X_t \in \{-1, 1\}$ with $X_t \sim \mathrm{Bernoulli}(p_{t-1})$
- Memory coefficient $\lambda \in (0,\, 1)$ affecting the development of probabilities p_t as

$$X_t = 1 \rightarrow p_t = \lambda p_{t-1} \qquad X_t = -1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1})$$

$\Rightarrow$ "Success punished"

$$X_t = 1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1}) \qquad X_t = -1 \rightarrow p_t = \lambda p_{t-1}$$

$\Rightarrow$ "Success rewarded"

Random walk with varying probabilities

- Random walk with memory based the on standard Bernoulli random walk
- Given starting probability p_0
- $X_t \in \{-1, 1\}$ with $X_t \sim \mathrm{Bernoulli}(p_{t-1})$
- Memory coefficient $\lambda \in (0,\ 1)$ affecting the development of probabilities p_t as

$$X_t = 1 \rightarrow p_t = \lambda p_{t-1} \qquad X_t = -1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1})$$

–> "Success punished"

$$X_t = 1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1}) \qquad X_t = -1 \rightarrow p_t = \lambda p_{t-1}$$

–> "Success rewarded"

Random walk with varying probabilities

- Random walk with memory based the on standard Bernoulli random walk
- Given starting probability p_0
- $X_t \in \{-1, 1\}$ with $X_t \sim \mathrm{Bernoulli}(p_{t-1})$
- Memory coefficient $\lambda \in (0,\ 1)$ affecting the development of probabilities p_t as

$$X_t = 1 \rightarrow p_t = \lambda p_{t-1} \qquad X_t = -1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1})$$

–> "Success punished"

$$X_t = 1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1}) \qquad X_t = -1 \rightarrow p_t = \lambda p_{t-1}$$

–> "Success rewarded"

Random walk with varying probabilities

- Random walk with memory based the on standard Bernoulli random walk
- Given starting probability p_0
- $X_t \in \{-1, 1\}$ with $X_t \sim \mathrm{Bernoulli}(p_{t-1})$
- Memory coefficient $\lambda \in (0, 1)$ affecting the development of probabilities p_t as

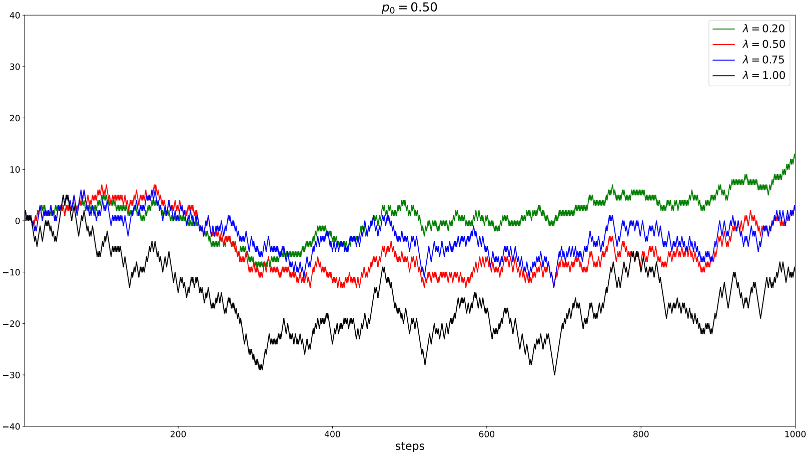
$$X_t = 1 \rightarrow p_t = \lambda p_{t-1} \qquad X_t = -1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1})$$

–> "Success punished"

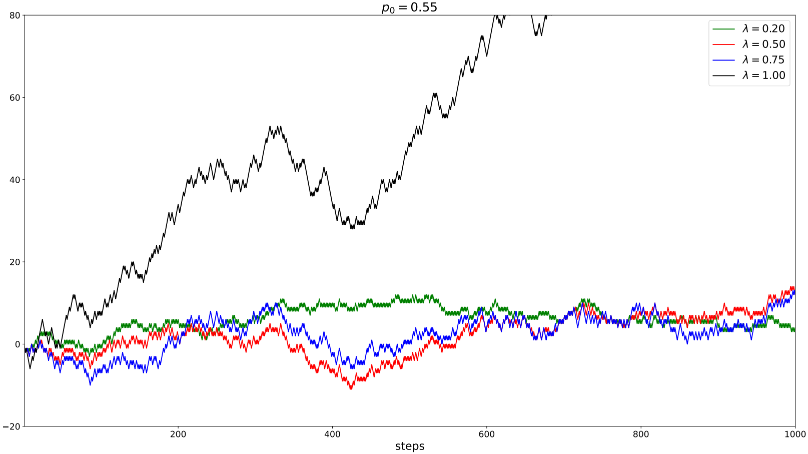
$$X_t = 1 \rightarrow p_t = 1 - \lambda(1 - p_{t-1}) \qquad X_t = -1 \rightarrow p_t = \lambda p_{t-1}$$

–> "Success rewarded"

Example - RW development



Example - RW development



Walk steps properties

$$EX_t = (2\lambda - 1)^{t-1}(2p_0 - 1)$$

$$\lim_{t \to +\infty} EX_t = 0$$

$$Var\, X_t = 1 - (2\lambda - 1)^{2(t-1)}(2p_0 - 1)^2$$

$$\lim_{t \to +\infty} Var\, X_t = 1$$

Walk steps properties

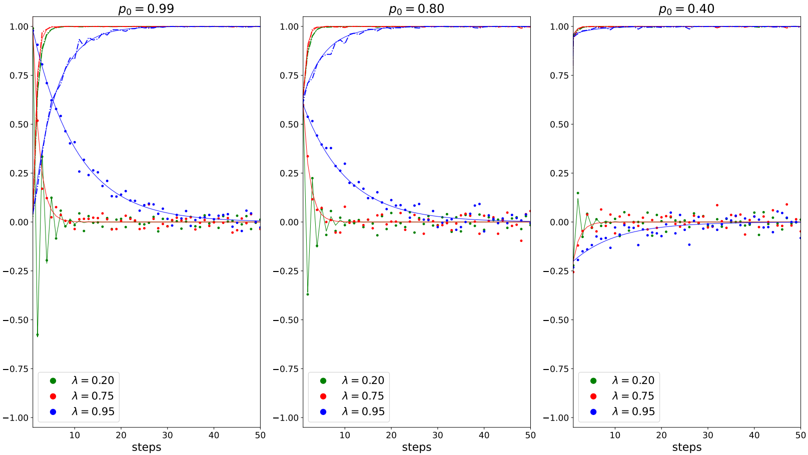
$$EX_t = (2\lambda - 1)^{t-1}(2p_0 - 1)$$

$$\lim_{t \to +\infty} EX_t = 0$$

$$Var\, X_t = 1 - (2\lambda - 1)^{2(t-1)}(2p_0 - 1)^2$$

$$\lim_{t \to +\infty} Var\, X_t = 1$$

Example - RW steps



Walk probabilities properties

$$EP_t = (2\lambda - 1)^t p_0 + \frac{1 - (2\lambda - 1)^t}{2}$$

$$\lim_{t \to +\infty} EP_t = \frac{1}{2}$$

$$Var\ P_t = (3\lambda^2 - 2\lambda)^t p_0^2 + \sum_{i=0}^{t-1} K(i; p_0, \lambda)(3\lambda^2 - 2\lambda)^{t-1-i} - k(t; p_o, \lambda)^2$$

$$\lim_{t \to +\infty} Var\ P_t = \frac{\frac{1}{2}(1 - \lambda^2)}{-3\lambda^2 + 2\lambda + 1} - \frac{1}{4}$$

Walk probabilities properties

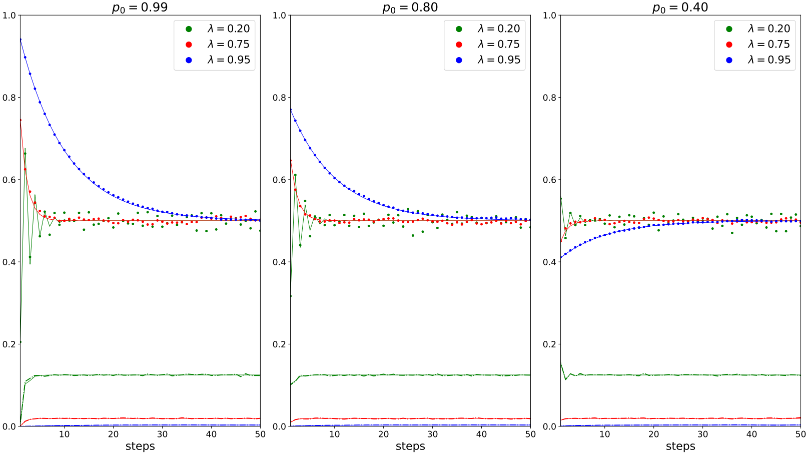
$$EP_t = (2\lambda - 1)^t p_0 + \frac{1 - (2\lambda - 1)^t}{2}$$

$$\lim_{t \to +\infty} EP_t = \frac{1}{2}$$

$$Var\, P_t = (3\lambda^2 - 2\lambda)^t p_0^2 + \sum_{i=0}^{t-1} K(i; p_0, \lambda)(3\lambda^2 - 2\lambda)^{t-1-i} - k(t; p_o, \lambda)^2$$

$$\lim_{t \to +\infty} Var\, P_t = \frac{\frac{1}{2}(1 - \lambda^2)}{-3\lambda^2 + 2\lambda + 1} - \frac{1}{4}$$

Example - RW probabilities



Walk position properties

$$ES_t = S_0 + (2p_0 - 1)\frac{1 - (2\lambda - 1)^t}{2(1 - \lambda)}$$

$$\lim_{t \to +\infty} ES_t = S_0 + \frac{(2p_0 - 1)}{2(1 - \lambda)}$$

$$Var\, S_t = t + 4\sum_{i=0}^{t-1} \sigma(i; p_0, 0, \lambda) - a(t; p_0, \lambda)$$

$$\lim_{t \to +\infty} Var\, S_t = c_1(p_0, \lambda)t + c_2(p_0, \lambda)$$

Walk position properties

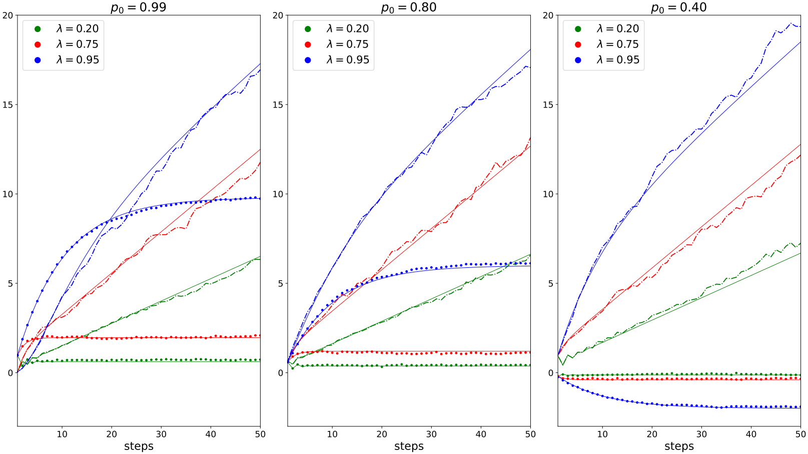
$$ES_t = S_0 + (2p_0 - 1)\frac{1 - (2\lambda - 1)^t}{2(1 - \lambda)}$$

$$\lim_{t \to +\infty} ES_t = S_0 + \frac{(2p_0 - 1)}{2(1 - \lambda)}$$

$$Var\ S_t = t + 4\sum_{i=0}^{t-1} \sigma(i; p_0, 0, \lambda) - a(t; p_0, \lambda)$$

$$\lim_{t \to +\infty} Var\ S_t = c_1(p_0, \lambda)t + c_2(p_0, \lambda)$$

Example - RW position



Success rewarding model

$$EX_t = 2p_0 - 1$$

$$Var\, X_t = 4p_0(1 - p_0)$$

$$EP_t = p_0$$

$$Var\, P_t = (2\lambda - \lambda^2)^t p_0^2 + p_0(1 - \lambda)^2 \sum_{i=0}^{t-1} (2\lambda - \lambda^2)^i - p_0^2$$

$$ES_t = S_0 + t(2p_0 - 1)$$

$$Var\, S_t = 4p_0(1 - p_0)t^2 + a(p_0, \lambda)t - a(p_0, \lambda)\frac{1 - (2\lambda - \lambda^2)^t}{(1 - \lambda)^2}$$

Success rewarding model

$$EX_t = 2p_0 - 1$$

$$Var\,X_t = 4p_0(1-p_0)$$

$$EP_t = p_0$$

$$Var\,P_t = (2\lambda - \lambda^2)^t p_0^2 + p_0(1-\lambda)^2 \sum_{i=0}^{t-1} (2\lambda - \lambda^2)^i - p_0^2$$

$$ES_t = S_0 + t(2p_0 - 1)$$

$$Var\,S_t = 4p_0(1-p_0)t^2 + a(p_0,\lambda)t - a(p_0,\lambda)\frac{1-(2\lambda-\lambda^2)^t}{(1-\lambda)^2}$$

More complex models

- Two λ parameters each affecting one direction of the walk
- Again two variants – success punishing and success rewarding

$$X_{t-1} = 1 \rightarrow p_t = \lambda_0 p_{t-1} \quad X_{t-1} = -1 \rightarrow p_t = 1 - \lambda_1(1 - p_{t-1})$$

–> "Two-parameter success punishing model"

$$X_{t-1} = 1 \rightarrow p_t = 1 - \lambda_0(1 - p_{t-1}) \quad X_{t-1} = -1 \rightarrow p_t = \lambda_1 p_{t-1}$$

–> "Two-parameter success rewarding model"

- M steps, $\lambda(t)$, n-dimensional walk

More complex models

- Two λ parameters each affecting one direction of the walk
- Again two variants – success punishing and success rewarding

$$X_{t-1} = 1 \rightarrow p_t = \lambda_0 p_{t-1} \quad X_{t-1} = -1 \rightarrow p_t = 1 - \lambda_1(1 - p_{t-1})$$

–> "Two-parameter success punishing model"

$$X_{t-1} = 1 \rightarrow p_t = 1 - \lambda_0(1 - p_{t-1}) \quad X_{t-1} = -1 \rightarrow p_t = \lambda_1 p_{t-1}$$

–> "Two-parameter success rewarding model"

- M steps, $\lambda(t)$, n-dimensional walk

More complex models

- Two λ parameters each affecting one direction of the walk
- Again two variants – success punishing and success rewarding

$$X_{t-1} = 1 \rightarrow p_t = \lambda_0 p_{t-1} \quad X_{t-1} = -1 \rightarrow p_t = 1 - \lambda_1 (1 - p_{t-1})$$

–> "Two-parameter success punishing model"

$$X_{t-1} = 1 \rightarrow p_t = 1 - \lambda_0 (1 - p_{t-1}) \quad X_{t-1} = -1 \rightarrow p_t = \lambda_1 p_{t-1}$$

–> "Two-parameter success rewarding model"

- M steps, $\lambda(t)$, n-dimensional walk

More complex models

- Two λ parameters each affecting one direction of the walk
- Again two variants – success punishing and success rewarding

$$X_{t-1} = 1 \rightarrow p_t = \lambda_0 p_{t-1} \quad X_{t-1} = -1 \rightarrow p_t = 1 - \lambda_1 (1 - p_{t-1})$$

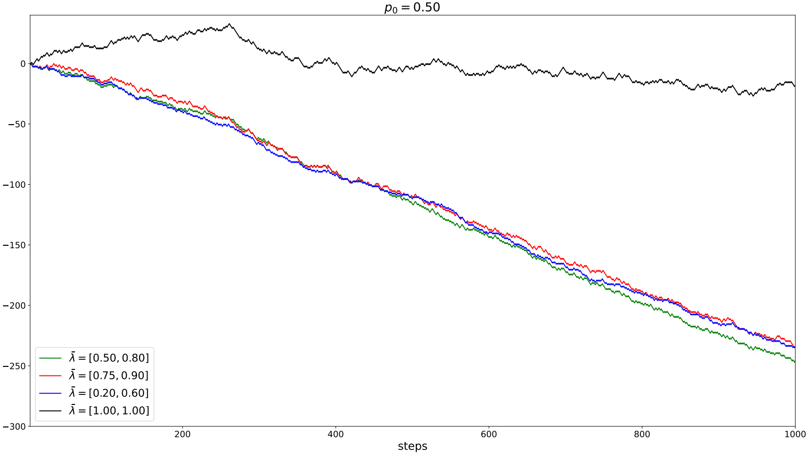
–> "Two-parameter success punishing model"

$$X_{t-1} = 1 \rightarrow p_t = 1 - \lambda_0 (1 - p_{t-1}) \quad X_{t-1} = -1 \rightarrow p_t = \lambda_1 p_{t-1}$$

–> "Two-parameter success rewarding model"

- M steps, $\lambda(t)$, n-dimensional walk

Example - two-parameter success punishing model



Model fitting

1. Find λ with known p_0, model type
2. Find p_0 with known λ, model type
3. Find p_0, λ with known model type
4. Find model type without any prior knowledge

Model fitting

1. Find $\overrightarrow{\lambda}$ with known p_0, model type
2. Find p_0 with known $\overrightarrow{\lambda}$, model type
3. Find p_0, $\overrightarrow{\lambda}$ with known model type
    - (1-3) maximal likelihood estimate & numerical optimization
4. Find model type without any prior knowledge

Model fitting

1. Find $\vec{\lambda}$ with known p_0, model type
2. Find p_0 with known $\vec{\lambda}$, model type
3. Find p_0, $\vec{\lambda}$ with known model type
   - (1-3) maximal likelihood estimate & numerical optimization
4. Find model type without any prior knowledge
   - (4) Akaike information criterion & numerical optimization

Real life application

- Success rewarding model well suited for modelling tennis matches
- Model trained on 2009–2018 men Grand Slam tournaments
- Model applied on 2019 US Open
  - Live betting against bookmaker
  - Bankroll: 0.52 units
  - Profit: 2.24 units $\rightarrow$ 430% ROI
  - Only 128 bets placed

Real life application

- Success rewarding model well suited for modelling tennis matches
- Model trained on 2009–2018 men Grand Slam tournaments
- Model applied on 2019 US Open
    - Live betting against bookmaker
    - Bankroll: 0.52 units
    - Profit: 2.24 units $\rightarrow$ 430% ROI
    - Only 128 bets placed

Real life application

- Success rewarding model well suited for modelling tennis matches
- Model trained on 2009–2018 men Grand Slam tournaments
- Model applied on 2019 US Open
  - Live betting against bookmaker
  - Bankroll: 0.52 units
  - Profit: 2.24 units $\rightarrow$ 430% ROI
  - Only 128 bets placed

Real life application

- Success rewarding model well suited for modelling tennis matches
- Model trained on 2009–2018 men Grand Slam tournaments
- Model applied on 2019 US Open
  - Live betting against bookmaker
  - Bankroll: 0.52 units
  - Profit: 2.24 units $\rightarrow$ 430% ROI
  - Only 128 bets placed

Real life application

- Success rewarding model well suited for modelling tennis matches
- Model trained on 2009–2018 men Grand Slam tournaments
- Model applied on 2019 US Open
  - Live betting against bookmaker
  - Bankroll: 0.52 units
  - Profit: 2.24 units $\rightarrow$ 430% ROI
  - Only 128 bets placed

Real life application

- Success rewarding model well suited for modelling tennis matches
- Model trained on 2009–2018 men Grand Slam tournaments
- Model applied on 2019 US Open
  - Live betting against bookmaker
  - Bankroll: 0.52 units
  - Profit: 2.24 units $\rightarrow$ 430% ROI
  - Only 128 bets placed

Summary

- A specific model of a random walk with memory
- Model properties derived
- Possible applications in a set of real life scenarios
- Initial results show big potential of the model

Thank you.

tom@skourim.com