# Elinkaarimallin ratkaiseminen dynaamisella ohjelmoinnilla ja reinforced learning-menetelmällä

Antti J. Tanskanen

27.4.2020

## Abstrakti

Elinkaarimallilla arvioimme työllisyysvaikutuksia elinkaarimallilla, joissa yksilöt optimoivat omaa hyötyään elinkaaren yli. Perinteisesti tällaisia malleja on ratkaistu dynaamisen ohjelmoinnilla tai maximum likehood-menetelmillä, mutta keinotekoisia neuroverkkoja funktioapprosimaattoreina käyttävät ”tekoälyratkaisut” mahdollistavat laajempien elinkaarimallien ratkaisemisen. Näytämme, että suuren osan koko sosiaaliturvaa ja työntekovalintoja sisältävä elinkaarimalli on mahdollista ratkaista malliriippumattomalla reinforced learning-menetelmällä. Ajamme mallin standardoidussa ympäristössä, jolloin tuleva algoritmikehitys on suoraan käytettävissä mallin ratkaisuun. Tässä artikkelissa kuvataan mallin rakenne.

# Johdanto

Elinkaarimalleja on käytetty Suomessa mm. eläkeuudistuksen vaikutusten arviointiin (Määttänen, 2013) ja (Hakola & Määttänen, 2005). Tässä tutkimuksessa tarkastellaan artikkeliin (Määttänen, 2013) pohjautuvaa elinkaarimallia. Mallia on laajennettu huomioimaan sosiaaliturvaa aiempaa laajemmin.

Elinkaarimallissa tarkastellaan henkilön elinkaarta pitkän aikaa, tässä tutkimuksessa 20-70-vuotiaana. Tällöin joka hetki vakuutettu arvioi, millaista nettotuloa työskentely tuottaa verrattuna sosiaaliturvaetuuksiin, ja tekee prioriteettiensä ja tulojen perusteella tekee päätöksen työskentelystään tai etuudella olosta. Mukana tarkastelussa tulee olla myös vapaa-ajan arvostus ja pidemmän aikavälin diskontatut hyödyt.

Työhön osallistumista voi pitää optimointiongelmana: vakuutettu maksimoi hyödyn nykyarvoa säätelemällä työhön osallistumista (Hakola & Määttänen, 2007). Optimointiongelmaan voi soveltaa esimerkiksi dynaamista ohjelmointia optimaalisen ratkaisun etsimisessä (Hakola & Määttänen, 2007).

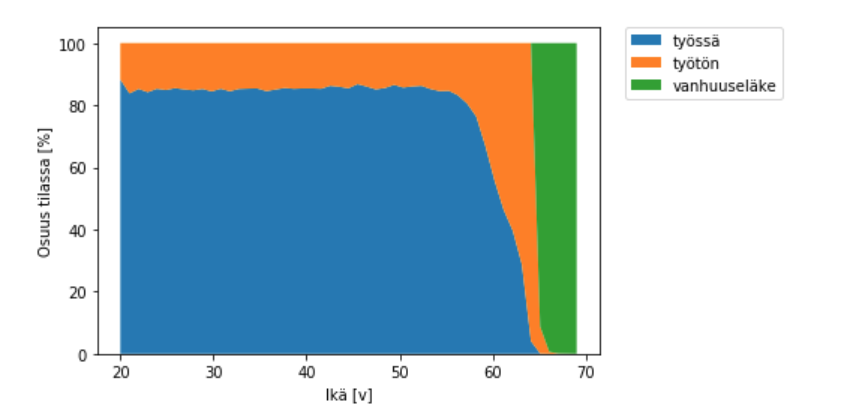
## Tämä tutkimus

Tässä tutkimuksessa työhön osallistumispäätöksiä ratkaistaan Reinforced Learning-menetelmällä, joka oppii yrityksen ja erehdyksen kautta. Algoritmi yrittää erilaisia toimia, tarkastelee millaisia palkkioita niistä seurasi, ja sillä perusteella pyrkii parantamaan toimiaan. Rajatuissa ongelmissa menetelmällä päästään usein erittäin hyviin, ihmistason ylittäviin suorituksiin (esim. Karpathy, 2016).

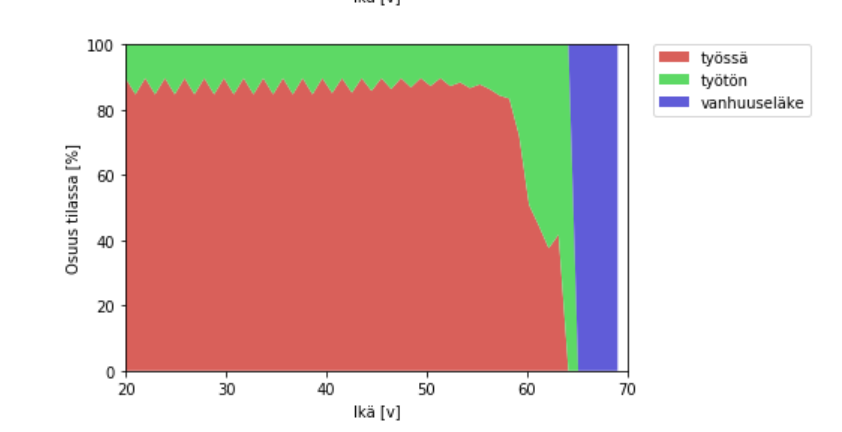
Malli on rakennettu Suomen sosiaaliturvan ja verotuksen toteuttavan kirjaston päälle käyttäen standardoitua tekoäly-ympäristöä. Tavoitteena on toisaalta tarkastella politiikkavaikutusten työllisyysvaikutuksia ja toisaalta pyrkiä laajentamaan mallikirjoa, jota elinkaarimalleilla voi ratkaista.

## Vertailu dynaamiseen ohjelmointiin

Vertailu dynaaminen optimointi ja RL *unemployment-v0* -mallin ratkaisussa.

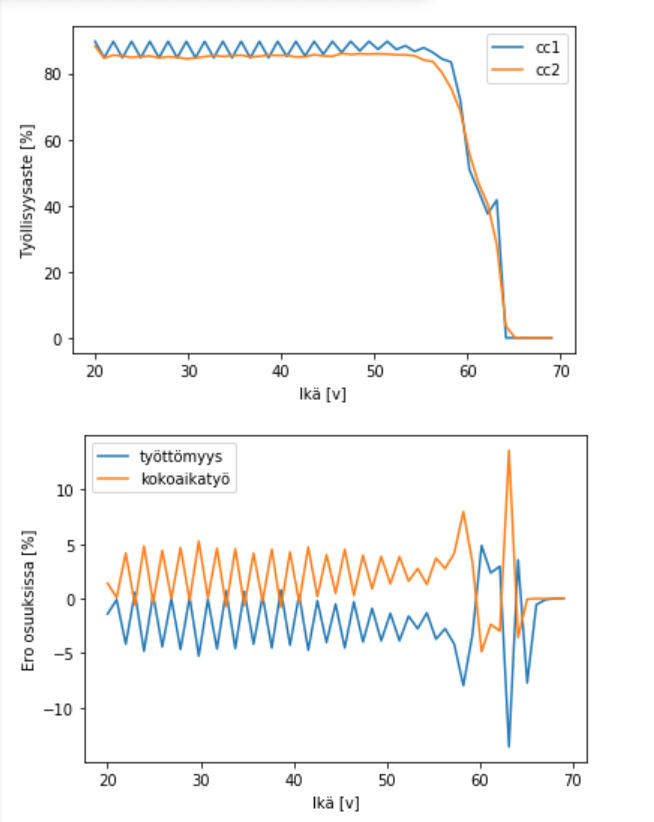


Kuva 5. Softmax



Kuva 6. Deterministic

Kuva 8. Dynaaminen optimointi



Kuva 7. Vertailu

# Diskussio

Elinkaarimalli on rakennettu modulaariseksi. Tämä mahdollistaa mallin helpon muokkaamisen, osin myös koska Python mahdollistaa objektien periytymisen ja jo aiemman koodin päälle rakentamisen. Esimerkiksi Ruotsin sosiaaliturva olisi melko helppo toteuttaa malliin korvaamalla *fin\_benefits* vastaavalla Ruotsin sosiaaliturvaa kuvaavalla mallilla.

# Kirjallisuus

Agarwal, Alekh, Sham M. Kakade, Jason D. Lee, and Gaurav Mahajan. "Optimality and approximation with policy gradient methods in markov decision processes." arXiv preprint arXiv:1908.00261 (2019).

Arel, I., Deep Reinforcement Learning as Foundation for Artificial General Intelligence. In: Wang P., Goertzel B. (eds) Theoretical Foundations of Artificial General Intelligence. Atlantis Thinking Machines, vol 4. Atlantis Press, Paris. 2012.

Brockman, G., Cheung, V., Pettersson, L, Schneider, J,, Schulman, J., Zaremba, W. OpenAI Gym, arXiv:1606.01540, 2016.

Brown, N., Sandholm, T. Superhuman AI for multiplayer poker. Science 365, s. 885-890, 2019

Hakola, Tuulia, and Niku Määttänen. Vuoden 2005 eläkeuudistuksen vaikutus eläkkeelle siirtymiseen ja eläkkeisiin: arviointia stokastisella elinkaarimallilla. Eläketurvakeskus, 2007.

Hessel, M. Rainbow: Combining Improvements in Deep Reinforcement Learning. arXiv:1710.02298 (cs.AI), 2017

Hill, Ashley, and Raffin, Antonin and Ernestus, Maximilian and Gleave, Adam and Kanervisto, Anssi and Traore, Rene and Dhariwal, Prafulla and Hesse, Christopher and Klimov, Oleg and Nichol, Alex and Plappert, Matthias and Radford, Alec and Schulman, John and Sidor, Szymon and Wu, Yuhuai,

Stable Baselines, Github repository, <https://github.com/hill-a/stable-baselines>, 2018.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. Neural networks, 4(2), 251-257.

Jiang, Z., Xu, D., & Liang, J. (2017). A deep reinforcement learning framework for the financial portfolio management problem. arXiv preprint arXiv:1706.10059.

Karpathy, A., Deep Reinforcement Learning: Pong from Pixels. 2016 http://karpathy.github.io/2016/05/31/rl/

Kober, Jens; Bagnell, J. Andrew; Peters, Jan. Reinforcement learning in robotics: A survey. The International Journal of Robotics Research, 2013, 32.11: 1238-1274.

Määttänen, Niku. "Eläkepoliittisten uudistusvaihtoehtojen arviointia stokastisen elinkaarimallin avulla." Kirjassa Lassila, Määttänen ja Valkonen (2013): Eläkeiän sitominen elinaikaan–miten käy työurien ja tulonjaon (2013).

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., ... & Lillicrap, T. (2018). A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. Science, 362(6419), 1140-1144.

Silver, D., Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, Demis Hassabis Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm, <https://arxiv.org/abs/1712.01815>, 2017

Sutton, Richard S., and Andrew G. Barto. Reinforcement learning: An introduction. MIT press, 2018 http://incompleteideas.net/book/bookdraft2017nov5.pdf

Tanskanen, Antti. Elinkaarimalli. Github repository. <https://github.com/ajtanskanen/lifecycle_rl> 2019a

Tanskanen, Antti. Suomen sosiaaliturva ja verotus Gym-ympäristönä. Github repository. <https://github.com/ajtanskanen/econogym> 2019b

Tanskanen, Antti. Suomen sosiaaliturva ja verotus Python-toteutuksena. Github repository. https://github.com/ajtanskanen/benefits 2019c

Tanskanen, Antti, Unelmoivatko robotit ansiosidonnaisesta sosiaaliturvasta? Submitted to KAK, 2019d

Wang, Z., Schaul, T., Hessel, M., Van Hasselt, H., Lanctot, M., & De Freitas, N. (2015). Dueling network architectures for deep reinforcement learning. arXiv preprint arXiv:1511.06581.

Wu, Y., Mansimov, E., Grosse, R. B., Liao, S., & Ba, J. (2017). Scalable trust-region method for deep reinforcement learning using kronecker-factored approximation. In Advances in neural information processing systems (pp. 5279-5288).

Ziebart, B. D., Maas, A., Bagnell, J. A., & Dey, A. K. (2008). Maximum entropy inverse reinforcement learning.