Task Sheet 1
Getting From NNs to Q-Values

Context. In the *Deep Learning & Neural Networks* course you saw how feed-forward nets learn fixed maps. When an agent interacts with a market, however, decisions are sequential and rewards lie in the future. This week we *simulate* a value function to bridge from supervised learning to reinforcement learning.

Learning goals

- Refresh `keras` basics in R.
- Build a dense network that outputs two Q-values (long vs. flat).
- Understand the data pipeline from *state* $\rightarrow$ network $\rightarrow$ (fake) supervised target.

Exercises

0. (**Glossary**) Download the glossary for this seminar from Moodle and make yourself familiar with its contents. Each new vocabulary you find here shall be investigated by you using that glossary and - if needed - additional information from elsewhere.
1. (**Markov Chain**) Describe briefly what a Markov chain is and find a suitable example in finance where it typically occurs.
2. (**Market data**) Download daily prices of `SPY` from 1 Jan 2015 onward using e.g. the package `quantmod` and turn them into percentage returns. Why are *returns* a better input than raw prices? *(Glossary keyword: "Reward")*
3. (**State vector**) Write `make_state(t_index)` that returns a `window_size` (=10) vector of the returns up to time t. Check: for $t = 15$ the function must look at bars 6–15. *(Glossary keyword: "State")*
4. (**Q-network**) Design a dense network with one hidden layer (`hidden_units` = 32, ReLU) and two linear outputs. Annotate the code: Which *action* does each output correspond to?
5. (**Targets**) Create 256 random sample states and compute artificial target y-values. y shall have two columns $y^{(\text{long})}$ and $y^{(\text{flat})}$ which are 1 if that action has taken place and 0 otherwise. Thus:

   $$y^{(\text{long})} = 1 \text{ if } ret_{t+1} > ret_t, \text{ else } 0; \quad y^{(\text{flat})} = 1 \text{ if } ret_{t+1} \leq ret_t, \text{ else } 0.$$

   How would you describe an agent, who is capable of implementing this strategy as his own trading policy?
6. (**Compile & train**) Train for 20 epochs (`mse` loss, Adam, $lr = 10^{-3}$) and plot the loss curve.
7. (**Inspection**) Pass the most recent state to your network and interpret the two numbers. Relate them to the definition of a Q-function from the glossary.

Reflection and research questions

- What is a window in this context?
- Match the terms {X, y, feed-forward NN} to the terms used above. To which terms do they correspond the most to?
- Explain in your own words the difference between a *Q-value* and a *policy*.
- Which Markov property do we *assume* when using only the last 10 daily returns as state?
- How will transaction costs enter the problem once the agent can change positions?