

## Week 4

---

- j) Het geïmplementeerde algoritme had het pad exact goed (0 fouten).
- k) Dit is te zien in lijn 201 van `model.py`. Hier is te zien dat een `for` lus doorlopen wordt voor elke mogelijke toestand. Per toestand wordt vervolgens een `for` lus doorlopen om de huidige toestand aan te passen op basis van de vorige toestand.
- l) Als je dit doet loop je het risico dat deze toestand cruciaal was voor het daadwerkelijke pad. Zolang de kans niet 0 is, is er een kans dat deze toestand onderdeel is van het pad. Het is dus niet verstandig om deze toestand te verwijderen.
- m) Evenals bij vraag j is het pad exact goed (0 fouten).
- n) Zoals uitgelegd in vraag j en m is het pad exact goed (0 fouten) bij het gebruik van het geïmplementeerde algoritme.
- o) De complexiteit van het Viterbi algoritme is  $O(n^2 \cdot m)$ , waarbij  $n$  het aantal toestanden is en  $m$  het aantal observaties. Dijkstra's algoritme heeft een complexiteit van  $O(n^2)$ , waarbij  $n$  het aantal knopen is.
- p) Die zijn er zeker. Eén manier om dat te doen is door de kansen van `observation_model` welke uit komen op 0 te over te slaan. In plaats van voor elke toestand de kans te berekenen, kunnen ook enkel de toestanden berekend worden waarbij de kans niet 0 is vanuit het observatie model.