Construction of a Markovian Model using the Viterbi Algorithm to predict Aerodynamic Control Laws of an Aircraft

Objective of the experiment: To build a computational stochastic model based on Markov chains to predict the most likely sequence of events using the Viterbi algorithm.

Learning concepts: Conditional probability, Markov property, stochastic optimizaion, dynamic programming.

Theoretical Concepts

I. **Introduction & overview**: We will consider a certain stochastic process with the following state space of dimension K, $S = \{s_1, s_2, ..., s_K\}$. Associated with this process is a T dimensional observation set $\mathbf{Y} = \{y_1, y_2, ..., y_T\}$ from amongst a possible N dimensional observation space $O = \{o_1, o_2, ..., o_N\}$. Note: $y_n \in O$. Further, consider an initial probability distribution given by $\mathbf{\Pi} = \{\pi_1, \pi_2, ..., \pi_K\}$. The probability transition matrix $\mathbb{P}$ is a $K \times K$ matrix with entries

$$p_{ij}(t) := \text{probability of transitioning from state } s_i \text{ to state } s_j = Prob(x_t = s_j \big| x_{t-1} = s_i),$$

and the emission matrix $\mathbb{E}$ is a $K \times N$ matrix with entries

$$e_{ij}(t) := \text{probability of observing } o_j \text{ from state } s_i = Prob(y_t = o_j \big| x_{t-1} = s_i).$$

Succinctly, we will often write $s_i \equiv i$ and $o_j \equiv j$ where it must be understood that $x_t = i$ refers to the random variable x_t taking the state s_i and $y_t = j$ refers to the random variable y_t being assigned the observable o_j. The goal of the prediction algorithm is to forecast the most likely sequence of states (events) $\mathbf{X} = \{x_1, x_2, ..., x_T\}, x_n \in S$ given a prescribed sequence of observables $\mathbf{Y}$, i.e. we need to compute

$$\text{argmax}_{\mathbf{X}} Prob(\mathbf{X} \big| \mathbf{Y}) = \text{argmax}_{\mathbf{X}} Prob(\mathbf{Y} \big| \mathbf{X}) Prob(\mathbf{X}) = \text{argmax}_{\mathbf{X}} Prob(\mathbf{Y}, \mathbf{X}).$$

Here argmax$\big(f(x)\big)$ returns the value of x at which the function $f(x)$ attains its maximum.

For convenience, you may think of a state space $S = \{$rainy, cloudy, sunny$\}$, an observational space $O = \{$walk, shop, clean$\}$ and a sequence of observations of activity patterns of Billoo, the handyman as $Y = \{$walk, walk, shop, walk, clean, walk, shop$\}$. The objective here is to find the most likely sequence of (hidden) states X corresponding to the sequence of observables Y. E.g., one possible likely outcome may be $X = \{$sunny, sunny, cloudy, sunny, rainy, sunny, cloudy$\}$. In this experiment, we will implement the Viterbi algorithm to predict the most likely sequence of states that corresponds to a sequence of associated observables assuming a Markovian stochastic model (also known as the *Hidden Markov Model* (HMM)).

II. **Construction and essential calculations of the Viterbi algorithm**:

In what follows, we will fix the notation $Prob(X_1 = x_1) \equiv Prob(x_1) \equiv \pi_1$. Note that if $T = 2$, then

$$\begin{aligned}
Prob(\mathbf{Y}, \mathbf{X}) &\equiv Prob(y_1, y_2, x_1, x_2) \\
&= Prob(y_1, y_2, x_2 \big| x_1) Prob(x_1) \\
&= Prob(y_1, y_2 \big| x_2, x_1) Prob(x_2 \big| x_1) Prob(x_1) \\
&= Prob(y_1 \big| y_2, x_2, x_1) Prob(y_2 \big| x_2, x_1) p_{12} \pi_1 \\
&= Prob(y_1 \big| x_1, x_2, y_2) Prob(y_2 \big| x_2) p_{12} \pi_1 \\
&= Prob(y_1 \big| x_1) Prob(y_2 \big| x_2) p_{12} \pi_1 \qquad (1)
\end{aligned}$$

In general, we have

$$\begin{aligned}
Prob(\mathbf{Y}, \mathbf{X}) &\equiv Prob(\mathbf{Y} = y_1, ..., y_T, \mathbf{X} = x_1, ..., x_T) \\
&= \underbrace{Prob(x_1)}_{\pi_1} Prob(y_1 \big| x_1) \underbrace{Prob(x_2 \big| x_1)}_{p_{12}} Prob(y_2 \big| x_2) \cdots Prob(y_T \big| x_T)
\end{aligned} \qquad (2)$$

The Viterbi algorithm involves *recursively* computing the Viterbi entries $V_{k,t}$

$$\begin{aligned}
V_{k,t} &:= \max Prob\big((y_1, ..., y_t), (x_1, ..., x_t = k)\big) \\
&= \text{probability of the best (most likely) sequence of states (ending with state } k\text{, i.e. } x_t = k) \\
&\quad \text{corresponding to the sequence of observables } (y_1, ..., y_t).
\end{aligned}$$

II.1. **Recursive computation of $V_{k,t}$**:

By comparing the terms on the right hand side of eq. (2) and the definition of the Viterbi entries above, we see that $V_{k,t}$ can be obtained recursively and consequently using the argmax function, we can find the most likely sequence of events. The algorithm includes calculation of the following three important terms.

- $V_{k,t} = \max\limits_{\alpha \in S} \big(Prob(y_t = j \big| x_t = k) p_{\alpha k} V_{\alpha, t-1}\big) = \max\limits_{\alpha \in S} \big(e_{kj} p_{\alpha k} V_{\alpha, t-1}\big)$
  with $V_{k,1} \overset{set}{=} Prob(y_1 = o_m \big| x_1 = k) \pi_k = e_{km} \pi_k$, and
- $x_T = \underset{\alpha \in S}{\text{argmax}} \big(V_{\alpha, T}\big)$.
- x_{t-1} = back_pointer(x_t, t) = value of x used to compute $V_{k,t}$ $\forall t > 1$.

Software Implementation

Pseudocode of the Viterbi algorithm:

INPUT: $S, \mathbf{\Pi}, \mathbb{E}, \mathbb{P}, \mathbf{Y} = \{y_1, y_2, ..., y_T\}$.

Part I: Initialization.

```
for each i of K states
   viterbi_prob(i,1) = π_i * e_{iy_1}
   viterbi_path(i,1) = 0
end for
```

Part II: Compute Viterbi probabilities and Viterbi path.

```
for each j of T-1 observations starting with T=2
   for each i of K states
      viterbi_prob(i,j) = max_{α∈S}(e_{iy_j} * p_{αi} * viterbi_prob(α, j-1))
      viterbi_path(i,j) = argmax_{α∈S}(e_{iy_j} * p_{αi} * viterbi_prob(α, j-1))
   end for
end for
x_T = s_{z_T} where z_T := argmax_{α∈S}(viterbi_prob(α, T))
```

The appearance of e_{ij} in the computation of `viterbi_path(i,j)` is unnecessary because it is non-negative and independent of α (so you may choose to skip it).

Part III: Retracking the most likely path $\mathbf{X}$.

```
for each j of T-1 observations from T to 2
   x_{j-1} = s_{z_{j-1}} where z_{j-1} = viterbi_path(z_j, j)
end for
```

OUTPUT: $X = \{x_1, x_2, ..., x_T\}$

III. **Questions**: Implement the above algorithm in MATLAB and use your program to answer the following questions.

1. Consider there are only two specific types of weather states, viz., *rainy*, *sunny*. Our friend Billoo, the handyman decides to either go *walking*, *shopping* or undertake *cleaning* depending on the type of weather on a given day. Let us say that we have recorded his daily chores over the past five days and observed that he undertook the following sequence of activities on subsequent days: *walking*, *walking*, *shopping*, *walking*, *cleaning*. Use the Markovian model explained above to predict the weather for the last five days. Assume the initial weather distri-

bution $\mathbf{\Pi} = \{0.43, 0.57\}$, the probability transition matrix $\mathbb{P} = \begin{pmatrix} 0.2 & 0.8 \\ 0.4 & 0.6 \end{pmatrix}$, where state 1 is *rainy* and state 2 is *sunny*, and the probability emission matrix $\mathbb{E} = \begin{pmatrix} 0.2 & 0.4 & 0.4 \\ 0.3 & 0.25 & 0.45 \end{pmatrix}$, where the columns (observations) are labelled in order of *walking*, *shopping* and *cleaning*, respectively.

2. Aircraft sensor data from the Airbus A330 is used to predict the flight characteristics and accordingly modify control inputs. One such flight characteristic is pitch up and pitch down motions (observables) measured by the angle of attack sensors. Any error in the pitch measurements may inadvertently affect the primary flight control laws (state variables) and have major consequences in the aerodynamic performance of the plane. In any aircraft there are three primary control laws, viz., *normal*, *alternate* and *direct*, each of which demand distinct inputs by the pilot and the on-board flight computer system. The flight envelope and failure protection modes are also distinctly different depending on the type of control law governing the flight at any given instant, e.g., normal law may have automated low-speed anti-stall protection whereas the same may not be available while the aircraft is operated under direct law. Therefore, accurate real-time prediction of the prevailing control law is essential for continuing safe flight and is monitored carefully by the company at the Airbus engineering systems headquarters. At a certain time, the company receives the following sequence of pitch measurements at 5 minute intervals. Devise a model using the Viterbi algorithm to predict the corresponding sequence of control laws that will likely be activated during the same time instant.

   Pitch data: 'up', 'down', 'down', 'down', 'down', 'up', 'up', 'down', 'down', 'down', 'down'.

   Consider the following probability transition matrix $\mathbb{P}$ and probability emission matrix $\mathbb{E}$ that is available from the Airbus database.

$$\mathbb{P} = \begin{pmatrix} 0.7 & 0.1 & 0.2 \\ 0.4 & 0.5 & 0.1 \\ 0.2 & 0.3 & 0.5 \end{pmatrix}, \mathbb{E} = \begin{pmatrix} 0.6 & 0.4 \\ 0.3 & 0.7 \\ 0.2 & 0.8 \end{pmatrix} \text{ and } \mathbf{\Pi} = \{0.8,\ 0.1,\ 0.1\}.$$

□