Predicting Healthcare Insurance Fraud Using Markov Observation Models

University of Alberta

STAT 499, Winter 2024

Abhimanyu Nag

Introduction

Motivating Example

In 2011, cardiologist John R. McLean was convicted of health care fraud, including unnecessary cardiac procedures, costing \$711,583.

Introduction

- Huge rise in healthcare fraud
- Markov Observation Model (MOM) offers promise
- The MOM expands upon the Hidden Markov Model framework

Background Concepts

- Stochastic Process
- Markov Chain
- Hidden Markov Model (HMM)
- Markov Observation Model (MOM)

Markov Observation Model Structure

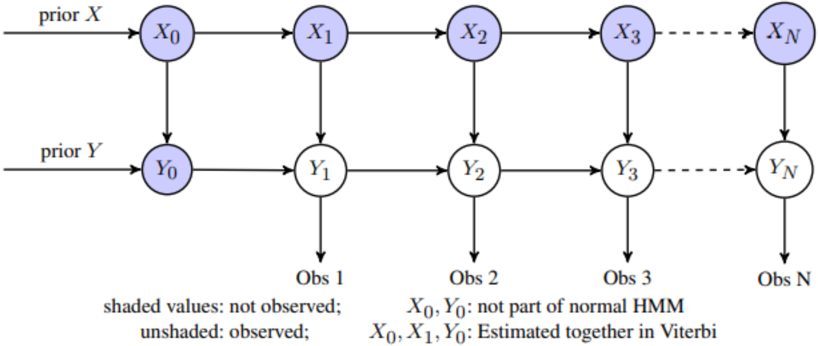

Figure: Markov Observation Model Structure

Formal Treatment of MOM

Suppose N is some positive integer (representing the final time) and O is some discrete observation space.

Assumption : observation Space is a Markov Chain too

In our model, like HMM, the hidden state is a homogeneous Markov chain X on some discrete (finite or countable) state space but in contrast to HMM, we allow self-dependence in the observations.

Formal Treatment of MOM

Suppose N is some positive integer (representing the final time) and O is some discrete observation space.

Assumption : observation Space is a Markov Chain too

In our model, like HMM, the hidden state is a homogeneous Markov chain X on some discrete (finite or countable) state space but in contrast to HMM, we allow self-dependence in the observations.

$P\,(Y_n\,|\,X_n, X_{n-1}, \ldots, X_1; Y_{n-1}, \ldots, Y_1) = P(Y_n \mid Y_{n-1}, X_n) = \sum_{y \in A} q_{y_{n-1} \to y}(x_n)$

Formal Treatment of MOM

MOM generalizes HMM by just taking $q_{y_{n-1}\to y}(x_n) = b_{X_n}(y)$, a state-dependent probability mass function. (X_n, Y_n) is a MOM with some noise ϵ_n

$$\begin{pmatrix} Y_n \\ Y_{n-1} \\ Y_{n-2} \\ \vdots \\ Y_{n-p+1} \end{pmatrix} = \begin{pmatrix} \beta_1^{X_n} & \beta_2^{X_n} & \beta_3^{X_n} & \cdots & \beta_{p-1}^{X_n} & \beta_p^{X_n} \\ 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{pmatrix} \begin{pmatrix} Y_{n-1} \\ Y_{n-2} \\ Y_{n-3} \\ \vdots \\ Y_{n-p} \end{pmatrix} + \begin{pmatrix} \beta(X_n) + \varepsilon_n \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Expectation Maximization for MOM

Algorithm 1: EM algorithm for MOM

Data: Observation sequence: $Y_1, \ldots, Y_N$
Input: Initial Estimates: $\{p_{x\to x'}\}, \{q_{y\to y'}(x)\}, \{\mu(x,y)\}$
Output: Final Estimates: $\{p_{x\to x'}\}, \{q_{y\to y'}(x)\}, \{\mu(x,y)\}$ `// Characterize MOM models`

```
/* Initalization.                                                      */
```

while *p, q, and μ have not converged* **do**

```
/* Forward propagation.                                                */
```

$\pi_0(x,y) = \mu(x,y)\ \forall x \in E, y \in O$;

$\rho_1(x) = \sum\limits_{x_0 \in E} \sum\limits_{y_0 \in O} \mu(x_0, y_0) p_{x_0 \to x} q_{y_0 \to Y_1}(x)\ \forall x \in E$;

$a_1 = \sum_x \rho_1(x)$

$\pi_1(x) = \frac{\rho_1(x)}{a_1}$.

for $n = 2, 3, \ldots, N$ **do**

$\rho_n(x) = q_{Y_{n-1} \to Y_n}(x) \sum\limits_{x_{n-1} \in E} \pi_{n-1}(x_{n-1}) p_{x_{n-1} \to x}\ \forall x \in E$.

$a_n = \sum_x \rho_n(x)$.

$\pi_n(x) = \frac{\rho_n(x)}{a_n}$.

```
/* Backward propagation.                                               */
```

$\chi_{N-1}(x) = q_{Y_{N-1} \to Y_N}(x)\ \forall x \in E$.

for $n = N-2, N-3, \ldots, 1$ **do**

$\chi_n(x) = \frac{q_{Y_n \to Y_{n+1}}(x)}{a_{n+1}} \sum\limits_{x' \in E} \chi_{n+1}(x') p_{x \to x'}\ \forall x \in E$.

$\chi_0(x,y) = \frac{q_{y \to Y_1}(x)}{a_1} \sum\limits_{x' \in E} \chi_1(x') p_{x \to x'}\ \forall x \in E, y \in O$.

```
/* Probability Update.                                                 */
```

$$q_{y \to y'}(x) = \frac{\sum\limits_{\xi} p_{\xi \to x} \left[1_{Y_1 = y'} \chi_0(x,y) \pi_0(\xi, y) + \sum\limits_{n=1}^{N-1} 1_{Y_n = y, Y_{n+1} = y'} \chi_n(x) \pi_n(\xi) \right]}{\sum\limits_{\xi} p_{\xi \to x} \left[\chi_0(x,y) \pi_0(\xi, y) + \sum\limits_{n=1}^{N-1} 1_{Y_n = y} \chi_n(x) \pi_n(\xi) \right]}$$

$\forall x \in E; y, y' \in O$.

$$\mu(x,y) = \frac{\mu(x,y) \sum_{x_1} \chi_0(x_1, y) p_{x \to x_1}}{\sum_\xi \sum_\theta \mu(\xi, \theta) \sum_{x_1} \chi_0(x_1, \theta) p_{\xi \to x_1}}\ \forall x \in E; y \in O.$$

$$p_{x \to x'} = \frac{p_{x \to x'} \left[\sum\limits_{y} \pi_0(x,y) \chi_0(x', y) + \sum\limits_{n=1}^{N-1} \pi_n(x) \chi_n(x') \right]}{\sum\limits_{x_1} p_{x \to x_1} \left[\sum\limits_{y} \pi_0(x,y) \chi_0(x_1, y) + \sum\limits_{n=1}^{N-1} \chi_n(x_1) \pi_n(x) \right]}\ \forall x, x' \in E.$$

Figure: Expectation Maximization for MOM

Viterbi Algorithm for MOM

Algorithm 2: Viterbi algorithm for MOM

Input: Observation sequence: $Y_1, ..., Y_N$
Output: Most likely Hidden state sequence: $P^*; y_0^*; x_0^*, x_1^*, ..., x_N^*$
Data: Probabilities $\{p_{x \to x'}\}, \{q_{y \to y'}(x)\}, \{\mu(x, y)\}$ // Distinguish MOM models

$\delta_{0,1}(y_0, x_0, x_1) = \mu(x_0, y_0) p_{x_0 \to x_1} q_{y_0 \to Y_1}(x_1), \ \forall y_0, x_0, x_1$ // Initialize joint distribution.

/* Substitute Marginal $\delta_1(x_1) = \sum_{y_0, x_0} \delta_{0,1}(y_0, x_0, x_1)$ if want $x_1^*, ..., x_N^*$. */

$\delta_2(x_2) = \max_{y_0 \in O; x_0, x_1 \in E} \left[\delta_{0,1}(y_0, x_0, x_1) p_{x_1 \to x_2}\right] q_{Y_1 \to Y_2}(x_2)$

/* Replace δ_n with normalized γ_n given below to avoid small number problem. */

$\psi_2(x_2) = \arg\max_{y_0 \in O; x_0, x_1 \in E} \left[\delta_{0,1}(y_0, x_0, x_1) p_{x_1 \to x_2}\right]$

/* Now propagate maximums, keeping track where they occur. */

for *n:=3 to N* **do**

- $\delta_n(x_n) = \max_{x_{n-1} \in E} \left[\delta_{n-1}(x_{n-1}) p_{x_{n-1} \to x_n}\right] q_{Y_{n-1} \to Y_n}(x_n)$ // Maximums
- $\psi_n(x_n) = \arg\max_{x_{n-1} \in E} \left[\delta_{n-1}(x_{n-1}) p_{x_{n-1} \to x_n}\right]$ // Maximum Locations

/* Termination */

$P^* = \max_{x_N \in E} [\delta_N(x_N)]$

$x_N^* = \arg\max_{x_N \in E} [\delta_N(x_N)]$

/* Path Back Tracking */

for *n:=N-1 down to 2* **do**

- $x_n^* = \psi_{n+1}(x_{n+1}^*)$

$(y_0^*; x_0^*, x_1^*) = \psi_2(x_2^*)$

Figure: Viterbi Algorithm for MOM

Research Question

Can we use the Markov Observation Model (MOM) on a large scale healthcare insurance dataset to quantify and predict fraudulent insurance claims and even model mindsets of the exact perpetrators of the crime?

Research Question

Can we use the Markov Observation Model (MOM) on a large scale healthcare insurance dataset to quantify and predict fraudulent insurance claims and even model mindsets of the exact perpetrators of the crime?

We claimed that we could

Research Question

Can we use the Markov Observation Model (MOM) on a large scale healthcare insurance dataset to quantify and predict fraudulent insurance claims and even model mindsets of the exact perpetrators of the crime?

We claimed that we could

I will now show how we did it

How?

Utilization of a dataset comprising simulated data with 9 columns and 6,000 data points

How?

Utilization of a dataset comprising simulated data with 9 columns and 6,000 data points

Table: Provider Claims Data Snippet

Month	Provider_ID	Patient_ID	Services_Rendered	Claim_Amount	Age_Group	Gender_Patient	Race_Patient	Fraud_Type	Claim_Category
1	1	1	Imaging	1443.0	Greater than 84	Female	Hispanic	Normal	Between 900 and 1700
1	1	2	Surgery	2330.0	Less than 65	Female	Other	Normal	Greater than 1700
1	1	3	Consultation	104.0	Less than 65	Female	Black	Normal	Between 100 and 900
1	1	4	Imaging	1257.0	Less than 65	Male	Black	Normal	Between 900 and 1700
1	1	5	Surgery	2452.0	Less than 65	Female	Asian	Normal	Greater than 1700
1	1	6	Medication	245.0	Less than 65	Male	Asian	Normal	Between 100 and 900
1	1	7	Medication	216.0	Less than 65	Female	White	Normal	Between 100 and 900
1	1	8	Surgery	2062.0	Greater than 84	Male	Asian	Normal	Greater than 1700
1	1	9	Consultation	121.0	Between 65 and 74	Female	Black	Normal	Between 100 and 900
1	1	10	Laboratory	524.0	Less than 65	Female	Other	Normal	Between 100 and 900

How?

Table: Fraudulent Provider Data Snippet

Month	Provider_ID	Patient_ID	Services_Rendered	Claim_Amount	Age_Group	Gender_Patient	Race_Patient	Fraud_Type	Claim_Category
1	5	151	Medication	220.0	Greater than 84	Male	Hispanic	Misrepresenting Surgical Complexity	Between 100 and 900
1	5	152	Medication	292.0	Greater than 84	Female	White	Pharmacy Kickbacks	Between 100 and 900
1	5	153	Consultation	103.0	Less than 65	Male	Hispanic	Pharmacy Kickbacks	Between 100 and 900
1	5	154	Imaging	1484.0	Less than 65	Female	Hispanic	Upcoding	Between 900 and 1700
1	5	155	Surgery	2126.0	Less than 65	Female	White	Pharmacy Kickbacks	Greater than 1700
1	5	156	Imaging	1133.0	Between 75 and 84	Male	Hispanic	Misrepresenting Surgical Complexity	Between 900 and 1700
1	5	157	Laboratory	502.0	Between 75 and 84	Female	Hispanic	Misrepresenting Surgical Complexity	Between 100 and 900
1	5	158	Surgery	2114.0	Between 75 and 84	Male	Other	Pharmacy Kickbacks	Greater than 1700
1	5	159	Consultation	115.0	Greater than 84	Male	White	Upcoding	Between 100 and 900
1	5	160	Imaging	1063.0	Less than 65	Male	White	Pharmacy Kickbacks	Between 900 and 1700

How?

- Define a set U of transaction marks and partition transactions into bins.

How?

- Define a set U of transaction marks and partition transactions into bins.
- Define event types E_i for each claim and create observations Y_t representing mark occurrences up to time t through event counting.

How?

- Define a set U of transaction marks and partition transactions into bins.
- Define event types E_i for each claim and create observations Y_t representing mark occurrences up to time t through event counting.
- Let E_i be the (random) event type of the ith claim, so

$$1_{E_i}(u) = \begin{cases} 1 & \text{if } E_i = u \\ 0 & \text{otherwise} \end{cases}$$

and $y_t(u) = \sum_{i=1}^{t} 1_{E_i}(u)$ counts the events with mark type u up to time t. Notice we process one transaction at a time.

Creation of Bins and Input Data

```
Y = Any[
 1:  [17, "Consultation", "Between 100 and 900", 1, 1]
 2:  [0, "Surgery", "Between 100 and 900", 1, 1]
 3:  [7, "Laboratory", "Between 100 and 900", 1, 1]
 4:  [0, "Imaging", "Between 100 and 900", 1, 1]
 5:  [6, "Medication", "Between 100 and 900", 1, 1]
 6:  [0, "Consultation", "Between 900 and 1700", 1, 1]
 7:  [0, "Surgery", "Between 900 and 1700", 1, 1]
 8:  [0, "Laboratory", "Between 900 and 1700", 1, 1]
 9:  [14, "Imaging", "Between 900 and 1700", 1, 1]
10:  [0, "Medication", "Between 900 and 1700", 1, 1]
11:  [0, "Consultation", "Greater than 1700", 1, 1]
12:  [6, "Surgery", "Greater than 1700", 1, 1]
13:  [0, "Laboratory", "Greater than 1700", 1, 1]
14:  [0, "Imaging", "Greater than 1700", 1, 1]
15:  [0, "Medication", "Greater than 1700", 1, 1]
16:  [14, "Consultation", "Between 100 and 900", 2, 1]
17:  [0, "Surgery", "Between 100 and 900", 2, 1]
18:  [14, "Laboratory", "Between 100 and 900", 2, 1]
19:  [0, "Imaging", "Between 100 and 900", 2, 1]
20:  [6, "Medication", "Between 100 and 900", 2, 1]
```

Figure: Structure of the bins; Counts extracted

EM Results

```
[array([[0.02637331, 0.04783824, 0.010629  , ..., 0.00878691, 0.04034541,
         0.04620467],
        [0.05376534, 0.01598379, 0.03890696, ..., 0.01841219, 0.01615797,
         0.00931618],
        [0.03396005, 0.00115832, 0.01877873, ..., 0.01623156, 0.02901965,
         0.03644356],
        ...,
        [0.05674344, 0.07405833, 0.00679279, ..., 0.00072537, 0.02246913,
         0.04659564],
        [0.0482747 , 0.0117708 , 0.00835729, ..., 0.04064329, 0.03701374,
         0.05518775],
        [0.0553257 , 0.0460259 , 0.0141991 , ..., 0.04606983, 0.01535844,
         0.01082162]]),
 array([[[0.48724221, 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         ...,
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ]],

        [[0.60107443, 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         ...,
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ]],

        [[0.56797694, 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         ...,
         [0.        , 0.        , 0.        , ..., 0.        ,
          0.        , 0.        ],
         [0.        , 0.        , 0.        , ..., 0.        ,
```

Figure: EM Algorithm Results; Probability Convergence

EM Results

```
 0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 ...,
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ]],

[[0.36240399, 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 ...,
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ]],

[[0.63421871, 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 ...,
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ],
 [0.        , 0.        , 0.        , ..., 0.        ,
  0.        , 0.        ]]]),
array([[0.00485714, 0.        , 0.        , ..., 0.        , 0.        ,
        0.        ],
       [0.05656153, 0.        , 0.        , ..., 0.        , 0.        ,
        0.        ],
       [0.03456379, 0.        , 0.        , ..., 0.        , 0.        ,
        0.        ],
       ...,
       [0.06002384, 0.        , 0.        , ..., 0.        , 0.        ,
        0.        ],
       [0.01580304, 0.        , 0.        , ..., 0.        , 0.        ,
        0.        ],
       [0.03913511, 0.        , 0.        , ..., 0.        , 0.        ,
        0.        ]])]
```

Figure: EM Algorithm Results; Probability Convergence

Viterbi Results and MOM fit to Data

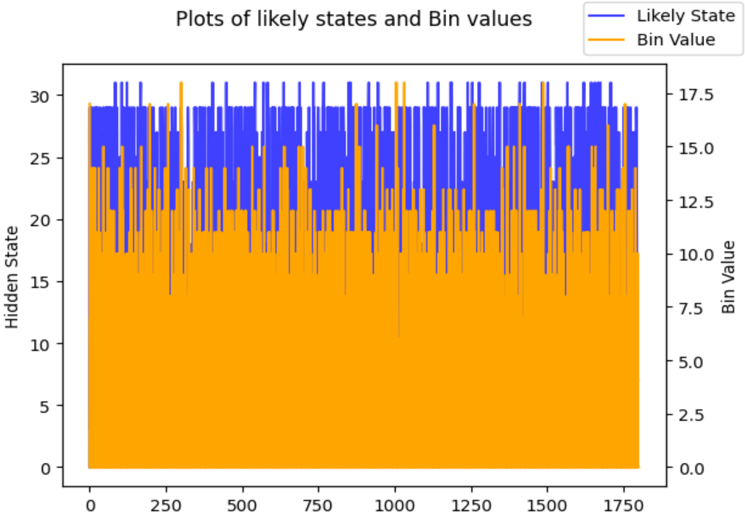

Figure: MOM fit to data

Interpretation of Results

- In a MOM framework, the *predictor distribution* represents the probability distribution of the next hidden state given the observed data up to the current time step.
- At time step 0, the predictor distribution is initialized based on the initial distribution and the first observation.
- For subsequent time steps, the predictor distribution is updated recursively based on the previous predictor distribution and the current observation.

Interpretation of Results

- In Bayesian model comparison, the Bayes factor measures the relative likelihood of two competing models given the observed data.
- In the context of MOM, the Bayes factor at each time step quantifies the evidence in favor of the model based on the observed data up to that time step.
- It is computed by summing over the predictor distributions at each time step.

Bayes Factor Results

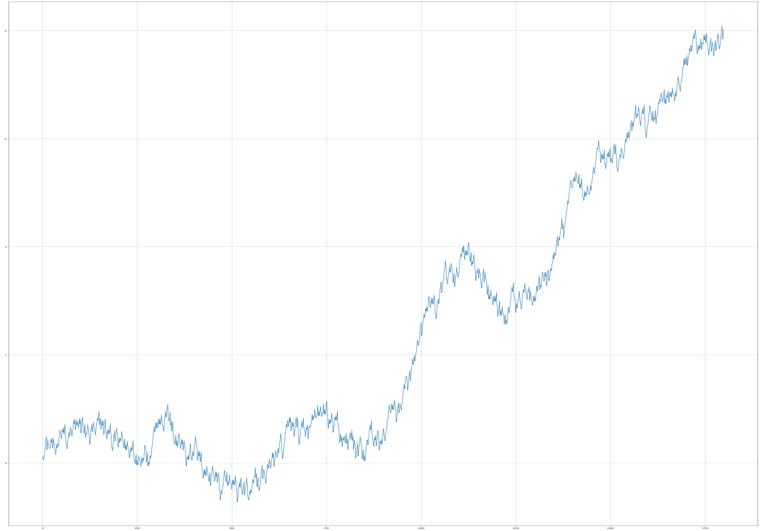

Figure: Bayes Factor Computation Results

Limitations

- Very sensitive to initial conditions
- Chances of false positives and negatives
- Sparcity
- Difficult to assess due to subjective mindsets

Conclusion and Future Work

The Markov Observation Model tells us which service providers have been defrauding insurance providers through modelling the difference in mindsets over data.

Future work would be based on better engineering and real world viability.

Aim to motivate and popularize the use of this algorithm in more areas such as network security and economic modelling

References

Kouritzin, Michael A. "Markov Observation Models." *arXiv preprint arXiv:2208.06368* (2022)

Baum, L. E. and Petrie, T. (1966). Statistical Inference for Probabilistic Functions of Finite State Markov Chains. The Annals of Mathematical Statistics. 37 (6): 1554-1563. doi:10.1214/aoms/1177699147.