CMPT 318

TERM PROJECT

ALI ALDEN 301302901

MICHAEL VANDERVELDEN 301386669

For Dr. Uwe Glasser

Simon Fraser University

Fall 2021

**Anomaly Detection and Data Analysis**

**using Hidden Markov Models**

Abstract:

This project utilizes Hidden Markov Models (HMMs) to perform anomaly detection on large datasets. Hidden Markov Models are statistical models in which the system in question is considered to be a Markov process. The data used was electricity consumption over a four-year period. After standardizing the data, the project’s methodology was comprised of three major tasks in relation to building and training Hidden Markov Models and using it for anomaly detection this was done using the R programming language. The major tasks are as follows: feature engineering, Hidden Markov modeling, and interpreting the results for the purpose of anomaly detection. Through principal component analysis (PCA), it was determined that the features global active power and global intensity were optimal for model training. Using the model, the degree of anomalies present in the data was determined by computing the log-likelihood for each observation sequence found in the data. The expectation was that a Hidden Markov model would perform adequately at minimum capacity; the Hidden Markov model will detect the anomaly in the injected dataset. These expectations were met accordingly, with the model facilitating anomaly detection in all three of the datasets.

**Table of Contents**

[Introduction3](#Part1)

Part 1: Feature engineering4

1.1: PCA Analysis

1.2: Feature Selection and Observation Time Window

Part 2: Building a Hidden Markov Model6

2.1: HMM Training

Part 3: Anomaly Detection8

3.1: Feeding the model

3.2: Anomalous Analysis

Part 4: Conclusion9

**Figures**

Table 1.1.1 Variability PCA components table3

Figure 1.1.1 Variability Bar plot4

Table 1.1.2 4

Table 1.1.35

Figure 1.2.16

Figure 1.2.26

Table 2.1.17

Table 2.1.28

Table 2.1.38

**Introduction**

This project is derived from an actual problem in North America, the electric power grid that covers most cities in the United States are operated by almost 500 different companies. North America needs power to make sure the economy runs as well as it should, power grid disruption will not just inconvenience the public but also disrupt the economy. One example of this is the 2003 northeast Blackout (Minkel, 2008). The blackout gave a significance in the need of cybersecurity measures for any electric power grid. One such technique that we can be used is the Hidden Markov Model type anomaly detection, this kind of anomaly detection technique not only is beneficial to the well being of the system, but also gives us a deeper understanding of how the data works and moves. We can then predict future patterns and take care of the problem before it has even happened.

This project works with what we consider a normal and anomalous state. Both which has been ran to a Hidden Markov Model algorithm in R. The log likelihood and the BIC was then scaled and compared with some threshold that the data should be under. HMM Anomaly detection usage ranges from a wide variety of different types of systems. Such range of systems are the water utilities system (Uwe Glasser, 2016). This paper focused on using HMM Anomaly Detection techniques to prevent disruption on the main waterline of the city of Surrey, British Columbia, Canada; to anomaly detection in Network Systems (Juan J. Flores) that detects and prevents network anomalous behaviour by computer systems. In both cases, HMM is used basically in the same concept in such a way that we have an observable output data from a set of input data. We can then label the normal behaviour of such system and detect anomalous behaviour from a model we have constructed. This project uses such Hidden Markov Model techniques to find and detect anomalous behaviour using the power grid output data.

[**Part 1: Feature Engineering**](#Part1)

**Part 1.1: PCA analysis**

We conducted three analyses on the data using principal component analysis (PCA). This is essentially a method to determine the relative importance of every feature in the data, such that these features are better suited to be used as training data for a Hidden Markov Model.

The R function prcomp() was applied over 3 years of test data yielding results as seen in Figure 1.1.1:

**Importance of components**

Table 1.1.1:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 |
| Standard deviation | 1.6911 | 0.9992 | 0.9698 | 0.9133 | 0.8777 | 0.68606 | 0.35513 |
| Proportion of Variance | 0.4086 | 0.1426 | 0.1343 | 0.1192 | 0.1101 | 0.06724 | 0.01802 |
| Cumulative Proportion | 0.4086 | 0.5512 | 0.6855 | 0.8047 | 0.9147 | 0.98198 | 1.00000 |

PC1 accounts for 40% of variability in the data, as visualized below

Figure 1.1.1:

Chart, bar chart

Description automatically generated

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 |
| Global\_active\_power | **-0.46872** | 0.134757 | -0.08736 | -0.06854 | 0.261449 | -0.76918 | -0.29968 |
| Global\_reactive\_power | -0.19475 | -0.74423 | 0.166002 | 0.60787 | 0.064466 | -0.0329 | -0.07678 |
| Voltage | 0.330526 | -0.13246 | -0.03506 | -0.13266 | 0.919 | 0.081727 | 0.056028 |
| Global\_intensity | **-0.5596** | 0.019129 | 0.001156 | -0.06409 | 0.138189 | 0.081176 | 0.810364 |
| Sub\_metering\_1 | -0.29884 | -0.12875 | 0.728446 | -0.47839 | 0.042941 | 0.240646 | -0.27363 |
| Sub\_metering\_2 | -0.28379 | -0.41294 | -0.65121 | -0.42742 | -0.05497 | 0.286867 | -0.23846 |
| Sub\_metering\_3 | -0.38747 | 0.472183 | -0.09419 | 0.438797 | 0.242829 | 0.503786 | -0.33575 |

Table 1.1.2:

To verify that values are accurate, the data was ran on a smaller set.

Data for 6 months, JAN 2009 – to July 2009

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | PC1 | PC2 | PC3 | PC4 | PC5 | PC6 | PC7 |
| Global\_active\_power | **0.500356** | -0.15051 | 0.101917 | -0.13568 | -0.0053 | 0.77475 | -0.31294 |
| Global\_reactive\_power | 0.240051 | **0.595666** | -0.20484 | -0.02376 | -0.73251 | -0.05213 | -0.07575 |
| Voltage | -0.26098 | -0.05891 | 0.209997 | -0.92553 | -0.16294 | -0.02391 | 0.024278 |
| Global\_intensity | **0.554497** | -0.04634 | 0.034814 | -0.13284 | 0.059017 | -0.06547 | 0.814714 |
| Sub\_metering\_1 | 0.325075 | 0.178927 | -0.60464 | -0.32577 | 0.481271 | -0.26847 | -0.29479 |
| Sub\_metering\_2 | 0.271993 | 0.432722 | 0.72738 | 0.012405 | 0.280928 | -0.27697 | -0.23218 |
| Sub\_metering\_3 | 0.369848 | -0.63059 | 0.087577 | 0.022243 | -0.35046 | -0.49336 | -0.30196 |

Table 1.1.3:

Since PC1 has the most variability with 40% and because the data is consistent, the highlighted variables will be used to train the Hidden Markov Model:

1. Global Active Power
2. Global intensity

PC2 components hold significant value for global reactive power, however due to the low variance at 17.2% of data, we chose global active power and global intensity as our HMM features.

**Part 1.2: Feature Selection and Observation Time Window.**

The test and train data sets are every Monday from 7am to 10am.

The time window values are consistent in test and train data. We ran a simple moving average on the test and train data and compared this to the anomaly data sets. The values of the anomaly data sets are different and do not fit the expected pattern seen in the test and train data.

Pattern seen in normal, non-anomalous data (Figure 1.2.1 & Figure 1.2.2).

**Figure 1.2.1:**

Chart, line chart

Description automatically generated

**Figure 1.2.2:**

Chart, line chart

Description automatically generated

**Part 2: Hidden Markov Model Engineering**

Data has been divided into two subsets: train data of 2 years from December 2006 to December 2008 and test data from January 2009 to December 2009. The data is further divided to show typical electricity consumption on Monday mornings between 7am to 10am (3 hours).

This gives model data consisting of 106 weeks of training and 50 weeks of testing.

**Part 2.1 HMM TRAINING:**

Data ran for 4-24 states to find best model.

We ran the model 2 times on the same machine, the resultant values were compared and averaged. The following values were obtained:

TRAIN DATA

Table 2.1.1:

|  |  |  |  |
| --- | --- | --- | --- |
| STATES | LOGLIKE | AIC | BIC |
| 4 | -18023.89 | 36109.78 | 36353.33 |
| 5 | -12781.91 | 25651.82 | 25997.51 |
| 6 | -11196.91 | 22511.81 | 22975.34 |
| 7 | -7224.733 | 14601.47 | 15198.55 |
| 8 | -5049.951 | 10289.90 | 11036.26 |
| 9 | -3513.379 | 7258.758 | 8170.1 |
| **10** | **-2975.834** | **6229.667** | **7321.706** |
| 11 | -1152.013 | 2632.026 | 3920.474 |
| 12 | -1056.944 | 2495.889 | 3849.081 |
| 13 | -593.247 | 1626.494 | 3185.145 |
| 14 | 1013.177 | -1524.354 | 253.925 |

TEST DATA

Table 2.1.2:

|  |  |  |  |
| --- | --- | --- | --- |
| STATES | LOGLIKE | AIC | BIC |
| 4 | -7878.144 | 15818.29 | 16037.92 |
| 5 | -5875.928 | 11839.86 | 12151.59 |
| 6 | -3889.521 | 7897.042 | 8315.044 |
| 7 | -2872.481 | 5896.961 | 6435.404 |
| 8 | -2915.289 | 6020.579 | 6693.633 |
| 9 | -2090.329 | 4412.657 | 5234.492 |
| **10** | **-1056.556** | **2391.113** | **3375.897** |
| 11 | -1116.486 | 2560.972 | 3722.876 |
| 12 | -206.6025 | 795.205 | 2295.777 |
| 13 | 365.8019 | -291.6038 | 1436.803 |
| 14 | 1882.959 | -3263.918 | -1291.963 |

The test and train values were starting to diverge between 14 to 24 states (meaning the model is overfitting)

In terms of model selection, state 10 seems to be the one making sense in the test data and training data with the value of loglikelihood and BIC as follows:

Model Log Likelihood (Normalized with N observations): (N is in minutes)

Table 2.1.3:

|  |  |  |
| --- | --- | --- |
| STATE= 10 (Normalized) | TEST | TRAIN |
| LOGLIKE | **0.1559661** | **0.1197909** |
| BIC | **0.3837372** | **0.3827547** |

The normalized values are very close together, meaning that the model is not overfit.

**Part 3: Anomaly Detection**

**Part 3.1: Feeding the model.**

we ran the data into the HMM model and the loglikelihood is way off for the 1st, 2nd, and 3rd Dataset with Anomalies.

After tweaking the parameters and model we decided on a .2 threshold for normalized likelihood and .6 for the normalized BIC.

**Table 3.1.1:**

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset with N Observations | Log Likelihood | AIC | BIC |
| Model | -1086.82 | 2451.639 | 3436.423 |
| Anomaly Dataset 1 | -6733.942 | 13745.88 | 14736.23 |

**Table 3.1.2:**

Normalized value 1:

|  |  |  |
| --- | --- | --- |
| Dataset with Normalized Observation | Log Likelihood | BIC |
| Model | **0.137878** | **0.383246** |
| Anomaly Dataset 1 | **0.733544** | **1.6052538** |

**Table 3.1.3:**

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset with N observations | Log Likelihood | AIC | BIC |
| Model | -1739.688 | 3757.376 | 3436.423 |
| Anomaly Dataset 2 | -6004.436 | 13745.88 | 4742.16 |

**Table 3.1.4:**

Normalized value 2:

|  |  |  |
| --- | --- | --- |
| Dataset with Normalized Observation | Log Likelihood | BIC |
| Model | **0.197165** | **0.5376598** |
| Anomaly Dataset 2 | **0.624077** | **1.446320** |

**Table 3.1.5:**

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset with N observations | Log Likelihood | AIC | BIC |
| Model | -1194.151 | 2666.301 | 3651.085 |
| Anomaly Dataset 3 | -6294.694 | 12867.39 | 13857.73 |

**Table 3.1.6:**

Normalized value 3:

|  |  |  |
| --- | --- | --- |
| Dataset with Normalized Observation | Log Likelihood | BIC |
| Model | **0.13537** | **0.389617** |
| Anomaly Dataset 3 | **0.68569** | **1.509556** |

This means that given the likelihood of these output Data to prop up is 3 times less likely than the model.

**Part 3.2 Anomalous Analysis**

We ran a simple moving average (to not overfit the data to see what is going on.)

Model data from every Monday with ROLLING MEAN OF 10 MINUTES PER OBSERVATION. (Global power)

As we saw on **Figure 1.2.1,** the spike in data should be gradual and not chaotic between these hours, yet the data itself has no patterns.

**Figure 3.1** **(SMA of anomalous data from 7am to 10am)**

Graphical user interface

Description automatically generated

**Part 4: Conclusion**

The model used to train and test the HMM worked well in this case where the anomaly was injected. Proper understanding of feature selection and fitting the right model is crucial to any Hidden Markov Model Anomaly Detection methodologies. Understanding and manipulation of unknown parameters is the utmost importance, this project emphasizes that to use any Machine Learning Anomaly detection technique we first must understand and work with such a model that will not just capture what the system is doing but to help us in better understanding of the future data. This kind of data extrapolation is important in the fight against cyber criminals who are using Artificial Intelligence to produce malicious codes and software that attacks critical infrastructure (such as hospitals, power grids, water supply) that must be protected at all costs.

This project also shows that data analysis is very important in risk mitigation and anomaly detection whether it be the Hidden Markov Model style of anomaly detection or other forms of methodology regarding cybersecurity.

**Last Remarks:**

A big thank you for Dr. Glasser and Amir Yaghoubi Shahir for a great semester. We are excited to see more ways of using cybersecurity concepts and methodologies from the academia side to real world applications in the forthcoming years.

# References

Juan J. Flores, A. A. (n.d.). Evolving Hidden Markov Models For Network.

Minkel, J. (2008, 08 13). *https://www.scientificamerican.com/article/2003-blackout-five-years-later/*.

Oehm, D. (2018). *http://gradientdescending.com/*. Retrieved from Hidden Markov Model example in R with the depmixS4 package: http://gradientdescending.com/hidden-markov-model-example-in-r-with-the-depmixs4-package/

Uwe Glasser, Z. Z. (2016). Hidden Markov Based Anomaly Detection. *2016 IEEE International Conference on Big Data (Big Data)*.