A pseudo Karnaugh mapping approach for datasets imbalance

Zhijian Lu¹, Gang Liu*¹ and Rongwen Liao¹

¹Shanghai University of Electric Power, Automation Engineering, Shanghai 200082, China

Abstract. The problem of dataset imbalance has raised a wide concern in many machine learning areas, but not in non-intrusive load monitoring, or load disaggregation. In this study, a pictorial evaluation method is proposed to represent the imbalance class distribution in datasets. We colored a Karnaugh maps according to the quantities of different variables combination to offer a visual impact to the whole dataset. After utilizing this method on a public dataset and its testing result, a clear imbalanced abundance in the dataset and an exciting performance have been found. A preliminary Python package to realize this mapping method has been uploaded on GitHub.

1 Introduction

In the field of Non-Intrusive Load Monitoring (NILM), most researchers focus on extracting the wave signal of a single appliance from aggregate public datasets, or recognizing the belongings of a single wave signal. There exists a vacancy of disaggregating a private-collected appliance from aggregate datasets.

NILM can easily and quickly identify load and decompose energy and monitor the use of appliances in a room. Ideally, it only needs data from smart meters to decompose the power consumption of appliances. The goal of NILM algorithm is to identify the power consumption of a single appliance based on the measurement data of the integrated power of the whole room. NILM can help the National Grid respond better to demand, store energy and develop and use more renewable energy, which is closely related to smart grid technology.

1.1 Unbalanced classification

As a multi-tag classification problem, NILM aims to determine whether a device is on or off within a given time step. However, in daily life, some electrical appliances are in a state of long-term switching on, and some electrical appliances are in a state of long-term switching off, which will be reflected in the data set and have a certain impact on the training model. There is always an imbalance in the data set in NILM, but appropriate techniques can be used to improve the robustness and effectiveness of the system. Zhang[1] adopts a stochastic sensitivity measure-based noise filtering and oversampling (SSMNFO) training base classifier to form a multi-class set of neural network and perform multi-label classification for NILM. Compared with the famous oversampling method, the synthetic minority oversampling technique (SMOTE) has reached remarkable performance in the actual non-invasive load monitoring application. Fang[2] uses SMOTE for data expansion and for alleviating the severe imbalance of switch events to achieve better performance than the current approach, especially in multi-appliance switching scenario.

Figure 1. Weighted Karnaugh map presented by Hassans [4]

Figure 2. Multi-valued Karnaugh map presented by Rushdi [5]

1.2 Variant Karnaugh Map utilizations

Apart from logistics analysis, a few attempts have utilized weighted Karnaugh Map [4], and multi-valued Karnaugh Map [5] in medical data statistics.

Hassans [4] proposed a meta-data presentation approach that relies on human’s pictorial perception capability. He present a methodological process of converting its equaling binary representation followed by furnishing values on to each cell in a Karnaugh Map instead of Boole. This pictorial graphing as shown in Figure 1, is expected to helped in simplifying the overview of meta-analysis and systemic reviews.

Rushdi[5] advocated and demonstrated a similar mapping approach of a multi-valued Karnaugh map, as Figure 2 shown, in qualitative comparative analysis for analyzing the clinical data in medical studies.

Other contributions like variable-entered Karnaugh map has been proposed in Banzhaf voting problems[6] and
multi-state system analysis[7–9] also proposed a numbered Karnaugh Map to assist their researches. These studies have greatly expanded the utilization of Karnaugh Map, but are still restricted to the quantity of variables included. The occupation of the map will be exponentially enlarged as variables increasing. Furthermore, human are far less sensitive to recognize numbers in compare to pure color.

1.3 Advanced mapping approach

While investigations on the presentation require of class imbalance in NILM dataset is still pending, and latest variant Karnaugh Maps fails encountering multiple variables, the pseudo Karnaugh Map (PKMap), a graph approach based on multi-valued Karnaugh Map, is proposed with the aim of filling this vacuum.

By counting, mapping and coloring the different appliance combination in a NILM dataset, the distribution of classes can be easily told with a single glance the color-disparate cells in the PKMap. And the arrangement of Karnaugh Map will help in pointing out to what appliances a specific cell in the PKMap referring.

Among the growing number of public datasets in NILM, the REFIT[10] offered by D. Murray is selected for demonstration since it has a long enough collecting period of two years, and have a cleaned edition released for easy acquiring. Especially, REFIT has 20 household for enough diversity, and involves 9 sub-metered power data in each house as well as the aggregate power data, which is exactly we need for mapping.

Along with the utility of PKMap in dataset, we also use this tool to mapping the testing consequence of metric, and some interesting discovers about the relationships between testing consequence and appliance combinations have been found. This implies the probability that characters in test dataset may impact the performance evaluation of a model. This may cause prejudice, bias or errors when cross-comparing in multiple models or datasets.

This study is expected to warn about the imbalance inherent to datasets to those who training, testing, or using NILM techniques for other purposes.

2 Assumptions and simplification

Some hypotheses are widely used in NILM researches, and others are arose for adaptive of the Karnaugh Map.

• Power consumers are parallelly connected, which means all of them share almost the same voltage value at any timestamp. This assumption is naturally right as none electrical appliance is connected in series, and is the crucial prerequisite for the establishment of formula (1).

\[
y_t = \sum_{k=1}^{n} x_t^{(k)}
\]  

Where \( y_t \) denotes the aggregate power of the timestamp \( t \), and \( x_t^{(k)} \) denotes the individual power of the \( k \)th appliance among \( n \) appliances. This is also the common way how researchers treat aggregate power data and individual power data, but no one mention its precondition.

• Each appliance is treated as binary-state for simplify, despite the working pattern of appliance in NILM analysis has been categorized[11,12]. In this paper, a constant preset power threshold is applied to transfer valued power data into Boolean. Each appliance is treated as ON state while its power value surpass the threshold and OFF state for the rest situation. Multi-state will be considered in future works.

• The plugging in or out behaviors during the dataset’s collecting period are neglected. There is no guarantee of the complete non-occurrence of such incident, and such occurrence do affect a lot in training and testing process. But we’ll still treat any individual data channel as one identical appliance since this is beyond our discussion.

3 Pseudo Karnaugh Map

An overview of the main operational phases of PKMap generation is shown in Figure 3, and is briefly summarized as follows.

Figure 3. Overview of the work flow of PKMap.

Table 1. A dataset slice and its binarized encodings.

<table>
<thead>
<tr>
<th>Timestamp Unix</th>
<th>power data (watt)</th>
<th>state encodings</th>
</tr>
</thead>
<tbody>
<tr>
<td>1397986205</td>
<td>152 0 3 ... 30</td>
<td>100...1</td>
</tr>
<tr>
<td>1397986211</td>
<td>151 0 3 ... 30</td>
<td>100...1</td>
</tr>
<tr>
<td>1397986220</td>
<td>222 101 3 ... 31</td>
<td>110...1</td>
</tr>
<tr>
<td>1397986221</td>
<td>208 101 2 ... 31</td>
<td>110...1</td>
</tr>
<tr>
<td>1397986226</td>
<td>208 84 2 ... 31</td>
<td>110...1</td>
</tr>
<tr>
<td>1397986236</td>
<td>198 80 3 ... 30</td>
<td>110...1</td>
</tr>
</tbody>
</table>

3.1 Binarization

As been expounded in Section 2, the power value will be converted into a Boolean value as we concern nothing but the ON/OFF states of appliances. Table 1 is a binarization example in which value 1, True in Boole, refers to ‘ON’ state, while value 0, False in Boole, refers to ‘OFF’ state.

A pre-set threshold of 8 watts is taken because a general sleep power of 2 to 3 watts have been detected on some appliance, and the minimal working power value is around 30 watts. A small threshold will help in classifying an appliance into the appropriate statement.

In the 4th column in Table 1, the power value of appliance 3 is lower than 8, so it will be treated as ‘OFF’ state and be encoded as ‘0’.
3.2 Encoding

Each one of the $2^n$ possible cases of $n$ appliances’ state combinations is encoded to a $n$-digit binary number, and the value of $k$th ($k = 1,2,\ldots,n$) position starting from the left, the kth digit, represents the state of appliance $A_k$ at the corresponding timestamp. In another way, the encodings of each timestamp mean the combination of the binarized states of all $n$ appliances.

The last column of Table 1 is the corresponding encodings of the former binarized load states in the same lines.

```
result={
    '11100010': 53,
    '11100110': 627,
    ...
},
```

Figure 4. A counting result from REFIT

3.3 Counting

After encoding, we count the encoded data line by line, and store the statistical result into a dictionary, an embedded Python class. As can be told, a $n$-appliance-involved dataset will be transferred into a one-dimensional data of $n$-digits binary numbers with a length of $2^n$.

For advanced illustration of the PKMap, a fully controlled statistical result of a smaller scale would be welcomed. The Artificial Dataset, AD as being simplified, is manually created for this purpose. As been expanded in Table 2, it’s a counting consequence of a hypothetical dataset with 4 appliances, which means it has $2^4 = 16$ encodings of 4-digits each.

### Table 2. Statistical result of the AD

<table>
<thead>
<tr>
<th>BIN</th>
<th>HEX</th>
<th>DEC</th>
<th>statistical result</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0001</td>
<td>1</td>
<td>01</td>
<td></td>
</tr>
<tr>
<td>0011</td>
<td>3</td>
<td>03</td>
<td></td>
</tr>
<tr>
<td>0010</td>
<td>2</td>
<td>02</td>
<td></td>
</tr>
<tr>
<td>0110</td>
<td>6</td>
<td>06</td>
<td></td>
</tr>
<tr>
<td>0111</td>
<td>7</td>
<td>07</td>
<td></td>
</tr>
<tr>
<td>0101</td>
<td>5</td>
<td>05</td>
<td></td>
</tr>
<tr>
<td>0100</td>
<td>4</td>
<td>04</td>
<td></td>
</tr>
<tr>
<td>1100</td>
<td>C</td>
<td>12</td>
<td>255</td>
</tr>
<tr>
<td>1101</td>
<td>D</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>1111</td>
<td>F</td>
<td>15</td>
<td>604</td>
</tr>
<tr>
<td>1110</td>
<td>E</td>
<td>14</td>
<td>58852</td>
</tr>
<tr>
<td>1010</td>
<td>A</td>
<td>10</td>
<td>2771</td>
</tr>
<tr>
<td>1011</td>
<td>B</td>
<td>11</td>
<td>38</td>
</tr>
<tr>
<td>1001</td>
<td>9</td>
<td>09</td>
<td>40272</td>
</tr>
<tr>
<td>1000</td>
<td>8</td>
<td>08</td>
<td>1282001</td>
</tr>
</tbody>
</table>

The values in AD by each encoding are set according to the following details we found in REFIT datasets:

- The “everything shut-down” state, which is encoded as 0 in decimal (whatever the quantity of appliances is), tends to obtain the highest frequency.
- Some state combinations never show up through the whole collecting period of the dataset.

3.4 Drawing

Inheriting the weighted Karnaugh map[4] and the multi-valued Karnaugh map[5], we map the data with marker of variables on all the four margins. This is a more intuitionistic way than using the table-style headers.

Like the Karnaugh Map, the PKMap comprises $2^n$ cells, where $n$ represent the number of concerned appliances in a dataset. Each cell in PKMap corresponds to a minterm denoted as $m_i$, $i \in \{0,1,2,\ldots,2^n\}$ in Boolean algebra. And it is made up of all the $n$ variables meaning a specific state combination of all the $n$ appliances.

The color map is select because it has multiple colors and is also capable in black and white printing without any loss suffered.

An example of PKMap with AD is shown in Figure 5, where we use a Karnaugh Map structure and fill in a heatmap instead of a textual table. Other visualization techniques of 2D array such as Hinton diagram are also welcomed.

This change is made to support large scale of data with far more than $2^8 = 64$ cells. In fact, the data we explored from REFIT in the latter section has $2^8 = 512$ cells. It's unnecessary and disorientating to expose a human before a textual table of that scale. These data will be observed visually with the help of visualization tools such as heatmap as we used.

From the PKMap presented in Figure 5, we can easily learn the basic distribution of the appliance working pattern and their relationship. All the three hypotheses put forward in Section 2 can be told.

4 Results & Discussion

To get an impression of PKMap to be expected in real datasets, we map the data in a public dataset REFIT [10] using the PKMap toolkit. The dataset is chosen since it has a clean edition released, and sub-metered power data...
included. The REFIT records 20 houses with 1 aggregate power sensor and 9 individual Appliance Monitors (IAMs) in each, and includes active power data in Watts with a 8-second interval [10].

4.1 PKMap utilizations on REFIT dataset

4.1.1 Logical expressions

Two Index Expression are introduced for better acknowledge the PKMap. One is the Peak Index Expression (PIE) representing the bright part of the map, and implies the abundant appliance combination in a dataset which is recommended for researching. The other is the Valley Index Expression (VIE) representing the dark part of the map, and implies the lack of appliance combinations in a dataset which require more attention. Concerning its REFIT we are using, the PIE usually refers to a magnitude of millions (equivalent to the quantity of total number of the dataset), whereas the VIE usually refers to a magnitude lower than hundreds or even zero.

4.1.2 Anaylsis of House4

The PKMap of House4 in REFIT is presented in Figure 6 with cyan-square-highlighted PIE, from which we can get:

\[ \text{VIE} = A_5 \cap (A_6 \cup A_8) \quad (2) \]

The 4th appliance is always ‘OFF’ unless both the 8th appliance and 9th appliance are ‘OFF’. They are mutually exclusive.

In other words, analysis on the \( k \) th (\( k \in \{1,2,3,5,6,7\} \)) appliances will be meaningful, and analysis on the \( k \) th (\( k \in \{4,8,9\} \)) appliances will require more attention.

4.1.3 Analysis of House 7

The PKMap of House7 in REFIT is presented with PIE highlighted in Figure 7a, and with VIE highlighted in Figure 7b. We can get:

\[ \text{PIE} = A_3 \cap A_7 \cap A_8 \quad (3) \]
\[ \text{VIE} = A_3 \cap (A_7 \cup A_8 \cup A_9) \quad (4) \]

This indicates that the \( x \) th (\( x \in \{3,7,8\} \)) appliance seldom work through the collecting period.

As can be noticed, the PIE and the VIE are roughly mutually Boole complementation as:

\[ \text{VIE} \approx \overline{\text{PIE}} \quad (5) \]

4.1.4 Analysis on House12

The PKMap of House12 in REFIT is presented in Figure 8, from where we can get:

\[ \text{PIE} = \overline{A_7} \cap \overline{A_8} \cap \overline{A_9} \quad (6) \]
\[ \text{VIE} = A_7 \cup A_8 \cup A_9 \quad (7) \]

This strongly indicate that the \( x \) th (\( x \in \{7,8,9\} \)) appliance do not appear through the whole dataset at all.

Extreme examples like this is not common. But it also warns us that will there be another one?
4.2 PKMap utilizations on testing result

When mapping the testing result, we bottom up the metric, because in PKMap, the bigger mean better.

4.2.1 Evaluation Metric

The Mean Absolute Error (MAE), a widely used metric in signal processing, as well as NILM, which is defined as:

\[
\text{MAE}^{(i)} = \frac{1}{T} \sum_{t=1}^{T} |x_t^{(i)} - \hat{x}_t^{(i)}| \tag{8}
\]

In which \(x_t^{(i)}\) represents the estimated power consumption of target appliance on timestamp \(t\) belonging to the appliance combination \(i\), while \(\hat{x}_t^{(i)}\) denoting the ground-truth power consumption offered by the original dataset. And \(T\) is the total length of an single appliance combination \(i\).

Generally, metrics are applied to continue wave signals. Whereas, In this case, a test result will crack and be regathered by each appliance combination, and then be passed to evaluate the performance.

4.2.2 on appliance 1 in house7

By comparing Figure 9 and Figure 10, an exciting and counterintuitive discovery that the lack of data do not obstacle the performance of a model has been told. In most cases, the model performance evenly nice regardless the disappearance frequency of the relative combination. But in some classes, the metric become worse.

For further analysis, Figure 11 demonstrating the relationship between the class frequency (counts) and the model performances is brought out. Apart of the discovery above, it seems that those worse performances only happens when the counts is below \(10^2\) magnitude.

5 Conclusion

In this paper, we have introduced the PKMap, a Karnaugh Map arranged heatmap mapping approach, as an indicator of dataset imbalance and a navigator of dataset synthesis.

Benefitting from its Karnaugh Map arrangement, peaks and valleys of a PKMap can be easily pointed out, and be named by a Boolean expression. Then, load combination cases that appear more frequently or less frequently throughout a dataset can be directly told.

According to peaks and valleys in the PKMap, some traps in the dataset have been found while selecting loads to recognize and slicing dataset to training and testing in general NILM analyzing.

In the latest experiment, an exciting discovery has been found that the performance may not concern with the abundance of the input dataset, and the model do perform worse in some combinations. These findings deserve further studies.

Reference