Data Mining and Analysis of Energy-Economics Time Series

Nicholas Chandler
Computer Science Department
Western Washington University
Bellingham, WA, USA
chandln@wwu.edu

Majd Safi
Computer Science Department
Western Washington University
Bellingham, WA, USA
safim@wwu.edu

***Abstract*—This paper discusses the analysis of energy economics time series data. Through the analysis this paper develops methods which can be extended for use on other time series data. The first method examined is the clustering of entire time series as a way of summarizing the data. Thereafter, using clustering, it looks at the yearly trends in each of the time series to analyse seasonality in the data. Finally, the data is split into yearly intervals and discretized as preparation for sequence mining where frequent behaviors of the time series are analysed.**

***Index Terms*—time-series, economics, energy, data-mining, clustering, sequence-mining, discretization, FRED, data-science, machine-learning**

I. Introduction

This project, looks at how historical and current events are represented by our time-series data as well as the existence of time-series trends. This is done by employing a variety of data mining and analysis techniques such as k-means clustering and data indexing. Through this analysis we have found several relationships among the various time series which tie into historical events that could have had influence on the records examined. Regarding the techniques used, this paper discusses the k-means algorithm as a way to reduce the dimensionality of data which prompts further inquiry on its value as a pre-processing technique for other machine learning and data mining methods. In the section after, the paper discusses discretization and sequence mining. Specifically, it looks at how the data can be binned by value and by its change, determined by a time step's value relative to the value of surrounding time steps. Finally, the PrefixSpan algorithm is used to find the top 30 frequent sequences from each of the discretized time series split on years.

II. Data

A. Dataset Selection

The datasets were selected using the Federal Reserve Economic Data (FRED) database since this is a US government source and therefore should have a standard of credibility pertaining to the reliability of the data. The datasets are all available at the link in the references [2]. Time series with the same intervals and endpoints were chosen for simplicity. Specifically, the data starts on January 1 st 2000 and ends on January 1 st 2023. The data is in monthly intervals so each year has 12 data-points where each data-point corresponds to the monthly average value of the attribute in the time series. The use of the average value over the entire interval rather than the value at the end date yields a more informative account of the phenomena through the span of each interval.

B. Dealing With Missing Values

There were two methods to deal with missing values such as the absence of market closing prices on bank holidays. First, an appropriate granularity of the data was chosen. That is, by using data which are recorded at large enough time intervals, they should not yield many missing values. In the case of this paper, months were chosen instead of days to minimize the effective gaps in the data. The cost of this is that the information is slightly less detailed and that the analysis pertains to long term phenomena.

C. Data Preparation Methods

To use and manipulate the data, we employed the python library, pandas [3]. This library allows a user to read in the data from a .csv file and manipulate it in a way similar to a relational database. The metric to test the similarity between the time series in preliminary analysis was Pearson's correlation coefficient [4]. This was done primarily to gain a cursory understanding of which time series were related beyond that of a preliminary visual analysis. The final data preparation technique utilized was indexing of the data, the topic of the next section.

III. Indices and Visualization

This section expands upon the initial transformation of the data and does preliminary analysis through visualization. It first defines and discusses time series indices. Then it examines the use of this method on the data.

A. Indices Overview

Indices in the context of time series are a way of examining a proportional relationship between the data at different points in time, centered around one point in time termed the *index*. The transformation is applied to all data-points in the time series to create a new time series called, *I*. It is defined:

$$I_t = \frac{f(t)}{f(t')} \cdot 100, \forall t \in T \tag{1}$$

The elements above are as such:

- $f(t)$ is the value of the time series at a given point.
- $f(t')$ is the value of the time series at the index time-step.
- I is the new time series.
- T is the set of all time-steps.
- t is the time-step at an arbitrary point in the time series.
- t' is the time-step at the index in the time series.

In the resulting time series the index, denoted by time-step t' has the value $f(t') = 100$. Values greater than 100 denote a higher value in the raw time series than the index. Values less than 100 denote a value lower than that of the index from the raw time series. This allows for a representation of values which are centered around one time-step. Note that the original units of the time series are no longer applicable when the index transformation is made. In short, one could view the index transformation as a form of normalization around one point in time.

B. Our Use of Indices

The analysis of the time series data utilizes the technique defined in the previous section to examine the data. Following the common practice, as defined in the literature, an index with a corresponding value that is near the mean value of all of the data is chosen. Specifically, the date: January 1st 2010 was used as the index. To enable comparisons between each of the time series, the same index date was used for each data-set. The next section will explore some visualizations.

C. Visualization

Visualization is an important first step in working with time series data. This section is concerned with the plotting of both the raw and indexed time series under examination during the project.

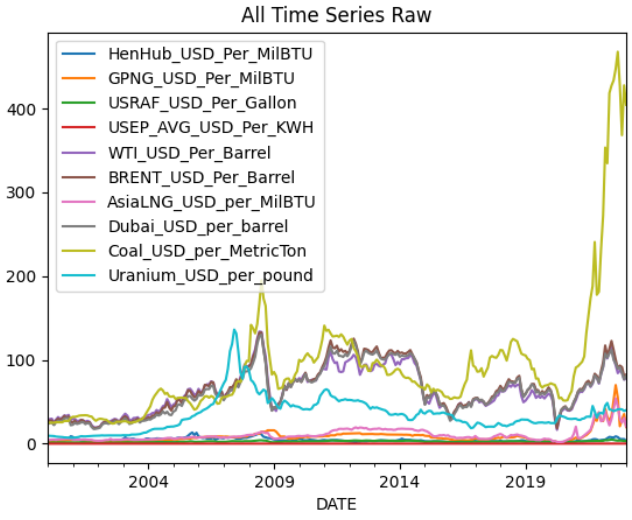

Fig. 1. All time series on one plot

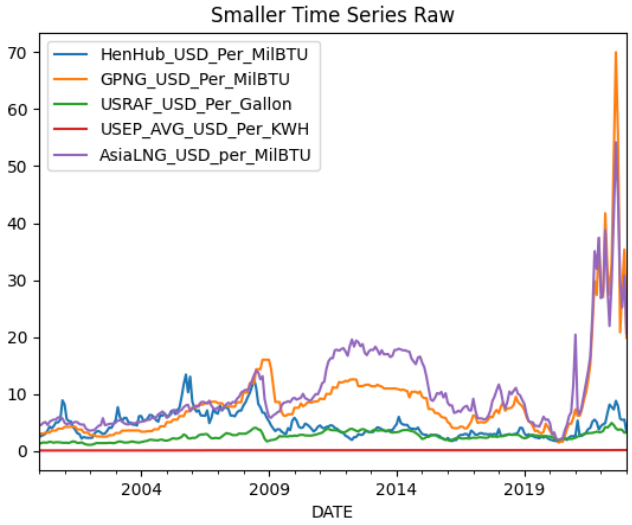

Fig. 2. Larger-valued time series omitted

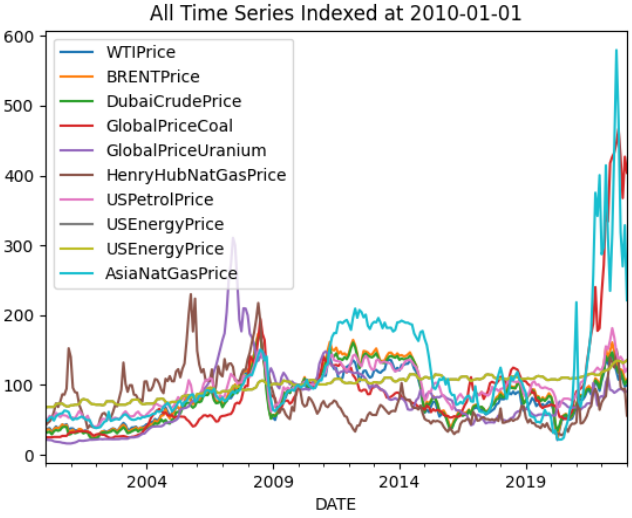

Fig. 3. All plots indexed at 01-01-2012

D. Preliminary Analysis of Plots

To begin the analysis, discussion with an economist took place. They identified several large economic events which took place in the span of the data. These events can be seen in the data visualization.

In the raw data plots specifically, Figures 1 and 2 show some interesting phenomena. The time period starting with 2008 and ending with 2009, has some erratic activity. The large drop in the price of oil here could likely be attributed to the economic recession following the market crash of 2008.

Furthermore, in the raw data plots, the time period starting with 2021 and ending with 2023 also exhibits some erratic activity. This may be attributed to the global COVID-19 pandemic and the consequential, far-reaching effects of changes in collective behavior.

Another point in the raw data plots from 2022 to 2023 displays some more erratic behavior. This may be attributed to the remnants of the COVID-19 pandemic and the beginning of the war in Ukraine.

There are certainly more events shown in the data that have been collected, though the previous paragraphs name a few major ones. The sections concerning data mining techniques will now be presented.

IV. Time Series Clustering

A. K-Means Clustering

K-Means clustering is a clustering algorithm which forms k clusters around k centroids by creating a Voronoi Tessellation of the feature space. That is, it partitions the feature space into k subsets where each point in each subset is closer to its defining centroid than any other centroid by some distance measure.

B. Distance Metrics

The distance metric that is employed is Euclidean distance. Euclidean distance is derived from the Pythagorean theorem and is given by the following formula:

$$d(\vec{x}, \vec{y}) = \sqrt{(x_1 - y_1)^2 + \cdots + (x_n - y_n)^2} \tag{2}$$

This metric provides the *geometric* distance between two vectors x and y. It is commonly employed in K-Means clustering. This is applied in the context of time series by placing the values for the dependent variable (in this case the indexed values or USD) into vectors. The Euclidean distance between two vectors can then be computed as shown in the formula above.

C. Input Data

This section will discuss the additional pre-processing that went into the application of the Time Series clustering algorithm. Specifically, the consideration of the noise in the data will be discussed. It is also worth mentioning that the clustering was done with the indexed data rather than the raw data.

In consideration of the noise in the data, we decided that choosing a large enough granularity would mitigate the ill-effects of this noise. Specifically, month-long intervals were chosen to give an accurate picture of the phenomena being analysed while still mitigating the effects of irreducible noise. In support of this, month-long *averages* of the values that were selected were also used where possible to create a more accurate picture of the activity of the phenomena through each time step of the time series. Therefore, the noise mitigation technique was careful selection and curation of input data.

D. Our Process

Two approaches to time-series clustering were explored. First, all the data was examined at once to see which datasets were deemed similar by the algorithm. This provided insight into which datasets were related. Another technique was also used where one time-series was examined at a time to get more insight into the behavior through the time period. The former will be detailed first.

To apply K-Means clustering to time series data, a 3D array that contained all of the time series was created. This array had the shape (number of time series, number of intervals, dimensionality of time series), in this case for univariate time series this was (10, 276, 1). Next, the tslearn package was used to create an instance of the TimeSeriesKMeans class, which implements K-Means clustering with different distance metrics. The Euclidean distance metric was used because it yielded more favorable results.

In the case where a single time-series was examined, the selected series was first partitioned into equally sized sections. The resulting sub-sections were fed into the time series K-Means algorithm. This provided a summary of the time-series' behavior throughout that interval of time by clustering based on common phenomena. Note that the interval used is one year. The plots of the resulting centroids will be shown after the all time series clusters.

E. All Time Series Clustering

This section will present a few plots of the resulting clusters identified by the algorithm. First, the elbow plots used for all the time series clustering on indexed and raw data will be shown. Then the centroids for each of the clusters found by the K-Means algorithm will be examined. Do note that the centroids themselves look similar to the input time series.

The elbow plot for the indexed time series, Figure 4 shows the optimal number of clusters as K=7. The elbow plot for the raw time series, Figure 5, shows the optimal number of clusters as K=4. Since there is a disagreement between the elbow plots, we show both K=7 and K=4 clusters on both the indexed and raw data.

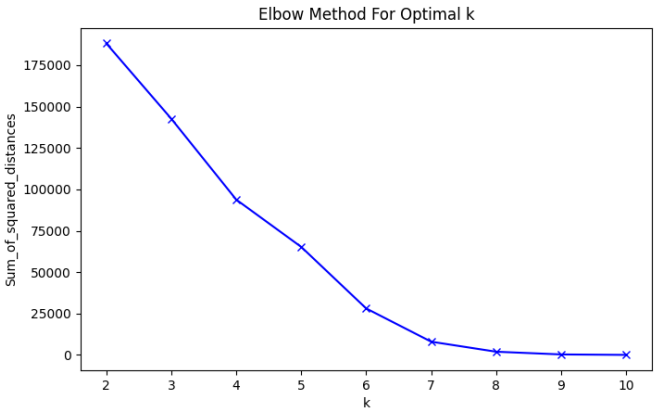

Fig. 4. Indexed time series elbow plot

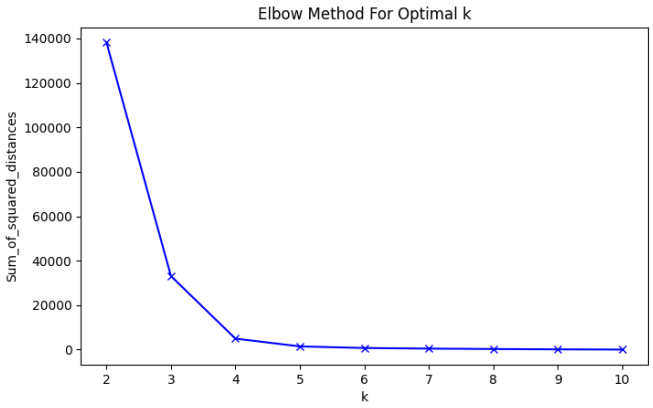

Fig. 5. Raw time series elbow plot

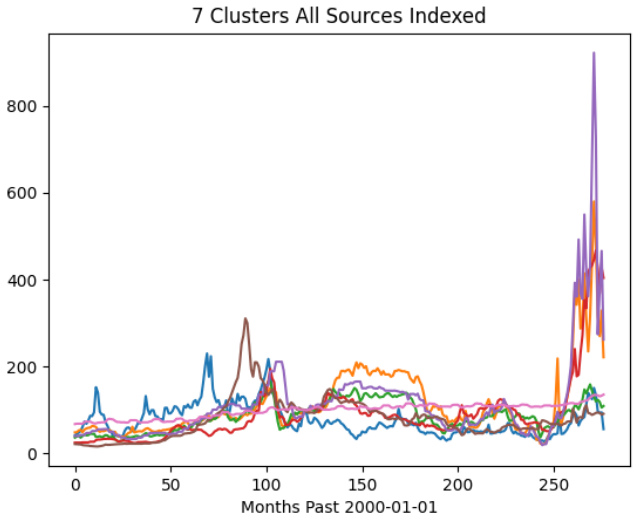

Fig. 6. 7 clusters of indexed time series

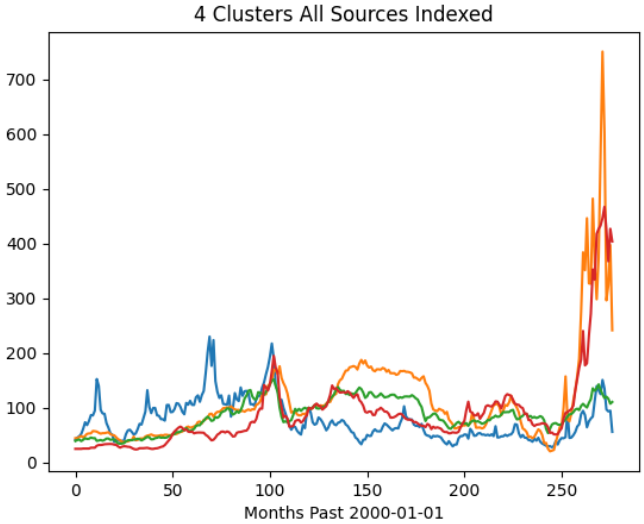

Fig. 7. 4 clusters of indexed time series

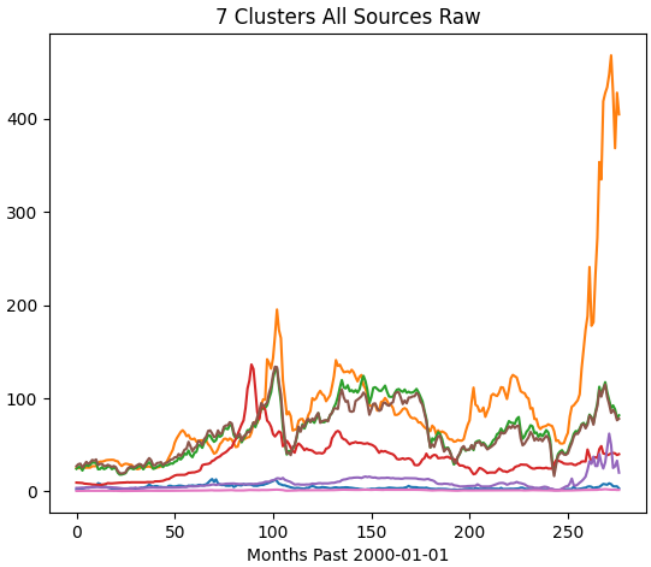

Fig. 8. 7 clusters of raw time series

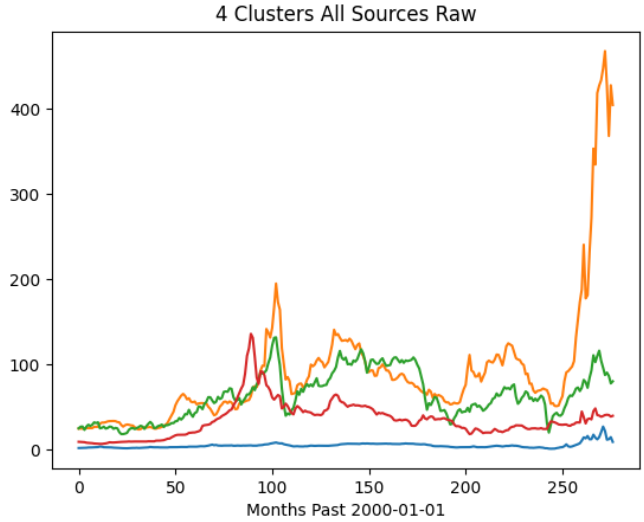

Fig. 9. 4 clusters of raw time series

F. All Time Series Results

This section will analyse the results of this algorithm and how the information gained from it plays a role in the further understanding of the time series data. Specifically, the analysis concerns which time series are more related to each other and why exactly that may be. The analysis will begin by examining the centroids of the clusters of raw data. Because the plot with only four clusters, Figure 9, is more easily interpreted, that one will be focused on for the analysis. Afterwards, the clusters of indexed data will be analysed though the results are similar. Then, the seven clusters will be looked at because the elbow method determined that this was the best value of k for the indexed plots.

It can be seen in Figure 9, the four cluster raw time series,

there is an apparent grouping. Especially when compared to the original raw data as in Figure 1. It appears that the orange centroid (that with the largest value at the time most recent) accounts for the global price of coal. The green centroid (the second largest at the time most recent) seems to account for the BRENT, WTI, and Dubai crude oil prices. The red centroid (the second smallest at the time most recent) seems to account for the global price of uranium and the natural gas prices. Finally, the blue centroid (the smallest at the time most recent) seems to account for the US average energy price and US gasoline price.

From these centroids, a few conclusions can be drawn. First, the clusters show that there is a similarity between oil prices. This is reasonable (and perhaps obvious) because the price of oil must be similar in order for each market to remain competitive in the global market. Next, it is seen that the natural gas are clustered together. Again this is reasonable because the prices must be similar for the suppliers to remain competitive. The anomalous component of this cluster is the global price of uranium being similar to that of natural gas. This could be due to the operation of the K-Means method but could also be worth further research into the connection between these markets. Finally, it is seen that the last cluster contains both the prices of US energy and US gasoline. These could be clustered together because they are both prices for consumers and tend to be lower than the prices of the other variables tracked.

The analysis will now focus on the seven centroids of indexed time series. The first noteworthy observation is that there are a few instances where the centroids overlap. This could be due to the overfitting of the data by the algorithm. A reason for this overfitting could be the similarity induced by the indexing method, making the distance between each of the time series smaller. A area for further inquiry is the potential for summarization of the data with a clustering algorithm. To do this one could substitute a centroid generated by the algorithm for the original data. This is described in more detail in the future work section. Some notable patterns can still be seen. For example, the three highest valued centroids at the time closest to today are likely representative of the global price of coal and the prices in the oil industry, denoted by WTI, BRENT, and Dubai. Finally, it is also seen that the smaller valued centroids at the time closest to today are reflective of the prices for natural gas and consumer-level commodities.

V. Single Time Series Clusters

Here are the resulting clusters for individual time series. Note that the red lines specify the determined centroid and the gray lines denote the sections of the original time series. The authors decided to only include four of the single time series clusters since the others displayed similar trends to those seen in the figures.

VI. Analysis of Single Time Series Clusters

Throughout each of the red centroids determined by the algorithm, one can see the common phenomena over the course of a year. Specifically, we can see that the prices in USD per unit increase during the Summer and tend to be lower during the winter months. Having made this preliminary observation, each graph will now be examined individually.

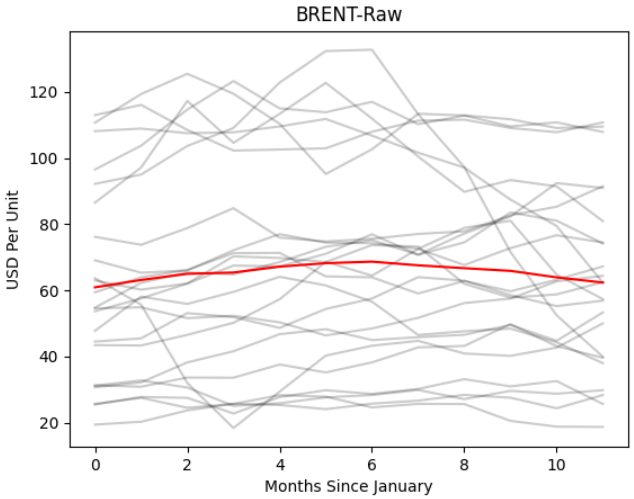

Fig. 10. One year summary of BRENT spot market performance

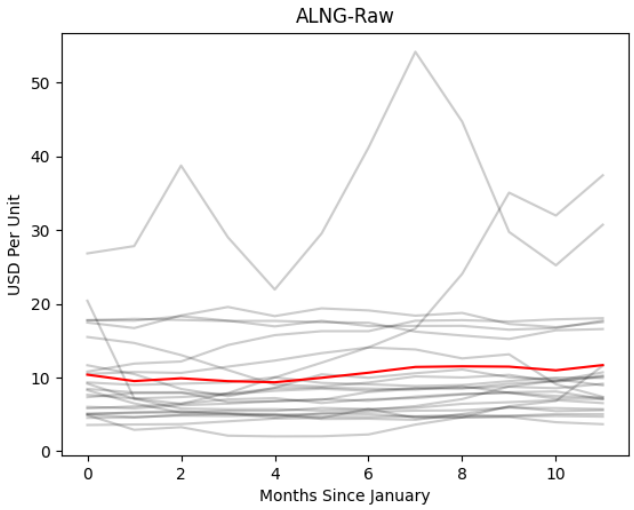

Fig. 11. One year summary of Asia Liquid Natural Gas Price

To begin, examine Figure 10, non-indexed BRENT performance. One can make the observation that the centroid displays a smooth curve, slightly increasing in the Summer months and maintaining lower values in the Winter months. This could be due to the increased production cost of gasoline in the Summer months, thus driving up the price of crude oil.

Now, examine Figure 11, the non-indexed Asia Liquid Natural Gas Price. The centroid here has mild behavior in that it indicates growth but no clear seasonal trend. This growth over the span of a year could be induced by the longer-term growth in prices which occur due to increased inflation of the US Dollar.

Next, examine Figure 12, the non-indexed US Regular All Formations performance. The centroid here displays similar

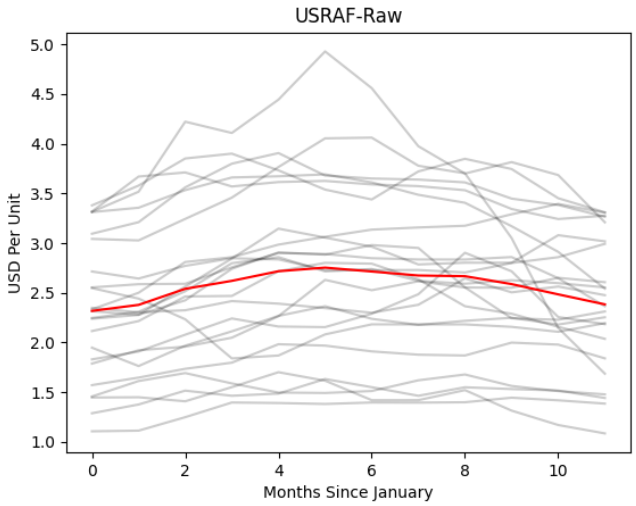

Fig. 12. One year summary of average US Gasoline Price

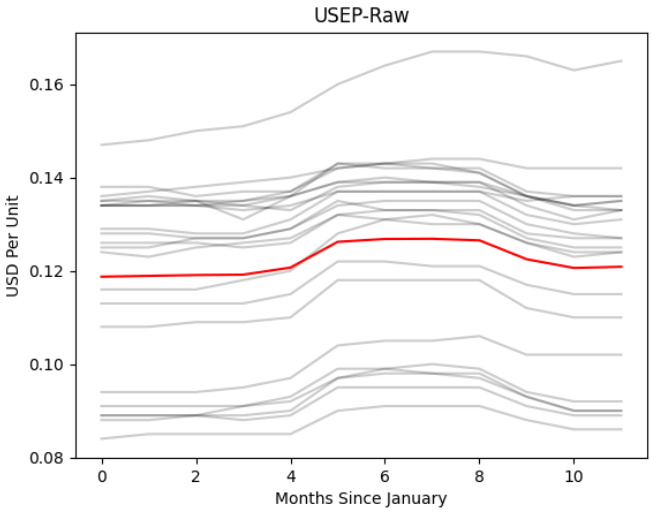

Fig. 13. One year summary of average US Energy Price

behavior to the centroid defined for BRENT. This could be attributed to the strong correlation between oil prices and gasoline prices. Additionally, as mentioned earlier, the cost of gasoline production increases in the Summer due to the requirement by the EPA to produce a more environmentally-friendly seasonal blend [9]. It is also worth noting that the units for BRENT pricing and USRAF pricing are different. This trend is something that may have been missed in the preliminary visualization.

Finally, examine Figure 13, the non-indexed Average US Energy Prices. Here we also see the centroid denoting an increase of price in the Summer months. This could be attributed to the increased demand for electricity to mitigate hot Summer weather in places like the sun-belt of the United States [9].

Through this analysis, we have explored some of the seasonal trends present in the data and speculated why they may occur. The next section discusses approaches to discretization of the data and sequence mining.

VII. Discretization & Sequence Mining

A. Discretization of Variables I

In a first attempt, the discretization function in SciKitLearn library [6] was used. This function gave the option to decide the width of the bins using a uniform, quantile, or K-Means strategy. The K-Means binning strategy was chosen. This is because the K-Means strategy produces boundaries for each category that are learned from the data and tends to identify cut points well. Therefore it should be better for determining accurate boundaries in the discretization of continuous data.

Finally, mining sequences with 5 and 10 bins was tried. It was thought that too many bins could result in sequences that were too complex to interpret which would degrade their value. Similarly, it was thought that too few bins would result in an over-simplification of the data being analysed.

If sequence mining was related to the *market basket* idea from association rule mining, the market basket would be all of the intervals in the sequence.

The following visual consists of all the raw time series, 5 bin discretized, and 10 bin discretized time series. These sizes were chosen because the data was well-represented by them. Still, they left out some of the original detail. There are no labels here as this figure is meant to give an idea of how the transformation operates. With more labels, it would have been too crowded to have any value.

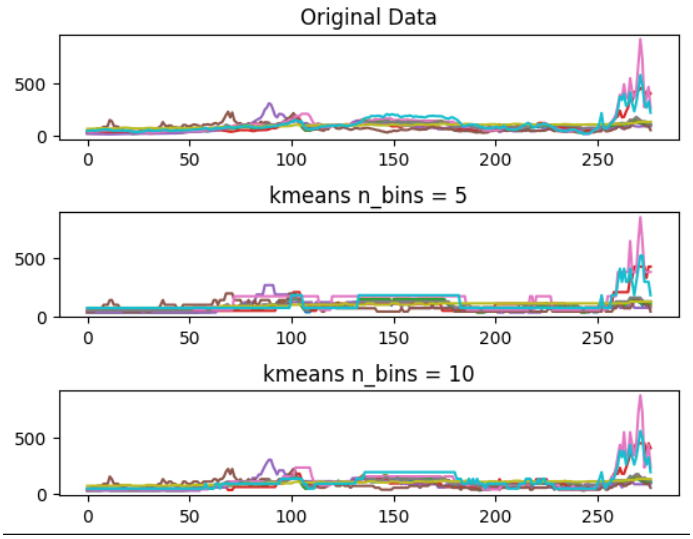

Fig. 14. Indexed data and discretized versions

B. Discretization of Variables II

The other discretization technique used was implemented by the authors. Specifically, it had a 3 month window (3 time steps) with a stride of 1 month (it moved forward in time by one month at a time) and would compare

the values on either side of the middle time step to the middle in order to identify both minima, maxima, sections of decrease, and sections of increase. The resulting array held information regarding the change in values seen in the time series. Essentially, this discretization method went over the data, found where the first order derivative would be positive (increases in the data), negative (decreases in the data), and zero (critical points). It then categorized the critical points as either minima, maxima, or plateaus. The resulting encoding is characterized by the following table:

characteristic	value
equal	5
decrease	4
increase	3
peak	2
valley	1
other	0

More information on the encoding scheme can be found in Appendix section A. Figure 15 displays a representation of one discretized time series with the method described above.

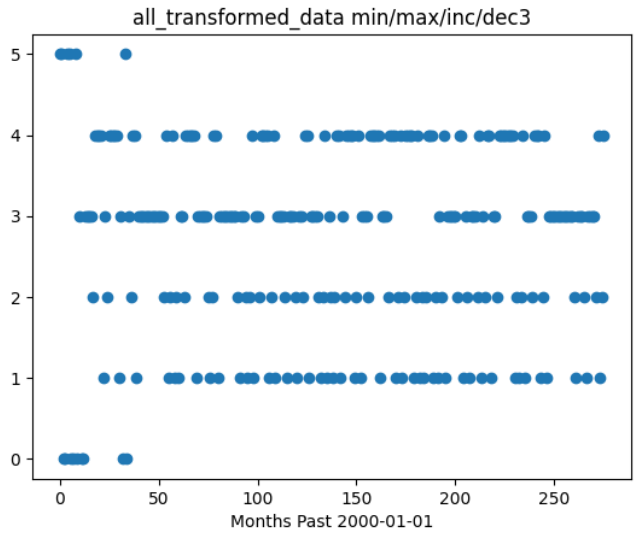

Fig. 15. Example Discretization of Global Price of Coal

C. The PrefixSpan Algorithm

The discretization technique that yielded better results was the one which accounted for change in the value of each time series using a 3-time-step moving window. This allowed for better insight into the behavior of the time series than the simple binning technique did. Specifically, the results obtained from this method were more easily interpreted. The process to obtain these sequences will now be described.

The algorithm that was selected was the PrefixSpan Sequence Mining Algorithm [7]. An implementation [8] online was found which allows use of an API for the PrefixSpan algorithm. This algorithm would take in multiple sequences of data and look for patterns common between the sequences. In the case of this project, each time series was split into 23 yearly sub-intervals, then it was passed into the algorithm. This algorithm did not require the specification of a minimum support like other sequence mining algorithms but rather the number of most-frequent sequences to mine. Specifically, the top 30 for each time series were selected.

D. Analysis I

To put the second method of discretization in perspective, we detail the analysis of the prefix span method applied to the time series transformed by the first discretization method. This approach utilized the same algorithm but had poor results. Specifically, it was common for the first 500 identified sequences to be thirty, 0s and then two events worth of activity. This made interpretation difficult. Furthermore, it was found that the inverse of the discretization was ineffective at restoring the complexity of the data to a degree useful to interpretation. Therefore, after some failed experimentation, it was decided to use the second method of discretization instead.

E. Analysis II

This section analyses sequence mining on the second form of discretization, the transformation which tracks changes in the data. A few common sequences found occurring in 9 of the 10 time series were sequences of length 2. One of which is the sequence, $1-1$, which when visualized appears to be a *double-valley* or double v sequence. To characterize this sequence, it displays a decrease in price followed by an increase, a decrease, and an increase. The conclusion that one can draw from this is that there is often a jagged nature in the change of stock prices. Another is the sequence $3-2$, which is an increase followed by a dip. One can conjecture that, with the prevalence of this one that after some months of increase there may be a dip in stock prices. Finally, a sequence of length 3 which was found in 9 of the 10 time series was $1-2-1$. This is a valley followed by a peak and then another valley. This pattern bears similarity to the initial pattern, $1-1$ though it takes place over a longer period of time. Again one can say that it is common to have increases and decreases following each other successively. Below are the proportion of yearly occurrences of each of the patterns we discussed above. Other common patterns are identified in Appendix section B.

time-series	proportion of years
WTI	0.96
BRENT	0.96
Dubai	0.96
GPC	0.65
GPU	0.7
HHNG	0.91
GPNG	0.43
USRAF	0.83
USEP	N/A
ALNG	0.91

The above figure examines the proportion of years where the sequence $1-2-1$, a valley, an increase, followed by a valley, occurred in each time series.

time-series	proportion of years
WTI	0.96
BRENT	0.96
Dubai	0.96
GPC	0.65
GPU	0.74
HHNG	0.91
GPNG	0.43
USRAF	0.83
USEP	N/A
ALNG	0.91

The above figure examines the proportion of years where the sequence $1-1$, a double valley, occurred in each time series.

time-series	proportion of years
WTI	1.0
BRENT	1.0
Dubai	1.0
GPC	0.7
GPU	0.7
HHNG	0.83
GPNG	0.52
USRAF	1.0
USEP	N/A
ALNG	1.0

The above figure examines the proportion of years where the sequence $3-2$, an increase followed by a dip, occurred in each time series.

In order to expand upon the analysis of the sequences we obtained, we examined the intersection of sequences common to a few sets. These are attached in Appendix C. A notable insight is that the cardinality of each of these intersection sets could be used as a measure of similarity between the yearly behavior of each constituent time series. In our example we see that the behavior of BRENT, WTI, and Dubai Crude is more similar than that of Henry Hub, Asia LNG, and the Global Price of Natural Gas by the simple visualization, Figure 1. Additionally, the cardinality of the intersection of the set of sequences of the oil spot market prices is greater than that of the cardinality of the intersection of the set of sequences pertaining to the natural gas prices.

This metric could be seen as a way to examine the similarity in time series with regard to their behaviors over time, disregarding the values or magnitudes of these time series.

VIII. Conclusion and Future Work

A. Conclusion

This section names several concrete takeaways from the project. They concern both the dataset and the methods discovered along the way. First, often when there is an increase in market prices for three months it will be followed by a dip in prices in the fourth month. This could be due to noise in the data or due to the nature of the market, regardless the sequence appeared in the majority of the examined time series. One may also conclude that the seasonality in energy economics time series indicates higher prices for oil, gasoline, and electricity in the summer. This contrasts with the view the authors held prior to the project of more expenditures in the winter due to increased heating costs.

Additionally, though related more to the architecture of the methods, the time series clustering offers a means of summarizing time series data with fewer points than originally given. This could be a means of dimensionality reduction while working with time series data. One should note that some information will be lost in the process but has the benefit of providing a more general view of the feature space. This is the case with many dimensionality reduction techniques.

Through sequence mining, there were some interesting technical discoveries as well. Specifically, one can discretize time series based on the changes of the series in a window which can allow for an insightful mining of sequences. This could be built upon in future work to allow for the mining of more complex sub-sequences by modifying the discretization technique to look at a larger window with more complex phenomena than simple optima and changes of value.

Finally, one can also conclude that examining the cardinality of the intersection of the set of sequences mined between time series discretized based on their respective rates of change could be used as a metric for determining similarity of the behavior of groups of time series without being affected by the magnitude of their values.

B. Future Work

Going forward, more experiments could be done with regards to dimensionality reduction of data using clustering. This would be especially useful in areas where high dimensional data is unfavorable yet multiple sources of information are available.

A software tool that will be designed and built as a result of this project is one which takes in a time series, a split length, and number of top sequences and outputs the frequent sequences. This could then be developed further (as mentioned above) to mine more complex technical indicators.

Another path forward is one which explores the similarity metric which examines the behavior of various time series based on the cardinality of the intersection of mined sequence sets. One could formalize it and analyse it further to determine the properties and possible applications associated with it.

Acknowledgment

We acknowledge Computer Science Professor James Hearne for constructive criticism and guidance through this assignment. We also acknowledge Economics Professor Reid Dorsey-Palmateer for consultation and assistance in some of our analysis.

REFERENCES

[1] Google Colab https://colab.research.google.com/drive/1c2q1nl5P4sIMcB38wKGeASdAwvPLFs0Z?usp=sharing

[2] Federal Reserve Economic Data (FRED) https://fred.stlouisfed.org/categories/32217?t=gasrt=gasob=pvod=desc

[3] Python Pandas https://pandas.pydata.org/

[4] Pearson, Karl (20 June 1895). "Notes on regression and inheritance in the case of two parents". Proceedings of the Royal Society of London. 58: 240–242. Bibcode:1895RSPS...58..240P

[5] K. Bringmann and M. Künnemann, "Quadratic Conditional Lower Bounds for String Problems and Dynamic Time Warping," 2015 IEEE 56th Annual Symposium on Foundations of Computer Science, Berkeley, CA, USA, 2015, pp. 79-97, doi: 10.1109/FOCS.2015.15.

[6] SciKit-Learn KBinsDiscretizer https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.KBinsDiscretizer.html

[7] Jian Pei et al., "PrefixSpan,: mining sequential patterns efficiently by prefix-projected pattern growth," Proceedings 17th International Conference on Data Engineering, Heidelberg, Germany, 2001, pp. 215-224, doi: 10.1109/ICDE.2001.914830.

[8] Prefixspan Implementation https://github.com/chuanconggao/PrefixSpan-py

[9] EPA Gas Regulations https://www.epa.gov/gasoline-standards/gasoline-reid-vapor-pressure

[10] Summer Electricity Prices https://vistaenergymarketing.com/blog/electricity-summer-increase/

APPENDIX

C. Discretization Method/Key

The values are encoded relative to a moving window that is three indices wide. That is, each of the following values describes the middle value relative to the outer two. Let the window consist of the values w_{t-1}, w_t, w_{t+1}. The following encodings are as shown:

5 - equal - $w_{t-1} = w_t = w_{t+1}$
4 - decreasing - $w_{t-1} > w_t > w_{t+1}$
3 - increasing - $w_{t-1} < w_t < w_{t+1}$
2 - peak - $w_{t-1} < w_t > w_{t+1}$
1 - valley - $w_{t-1} > w_t < w_{t+1}$
0 - other - Undefined

D. Common/Notable Patterns

We found a few patterns worth mentioning and encourage the reader to try drawing them out on a piece of paper based on the rules defined above to aid in visualization and understanding.

4 1 4 – deep valley
3 2 3 – tall peak
1 2 – low lightning bolt
2 1 – high lightning bolt
3 2 – increase then peak
1 1 – double dip
1 2 1 – valley-peak-valley
2 1 2 – peak-valley-peak
1 3 – swoosh
1 3 2 – low zig zag

E. Intersecting Sequences

The intersection of the sets of sequences from Henry Hub Natural Gas, Global Price of Natural Gas, and Asia Liquid Natural Gas: ['1 1 ' '1 1 2 ' '1 2 ' '1 2 1 ' '1 3 ' '1 3 2 ' '2 1 ' '2 1 2 ' '2 2 ' '3 2 ' '3 3 ']

The intersection of the sets of sequences from BRENT, WTI, and Dubai Crude:

['1 1 ' '1 2 ' '1 2 1 ' '1 2 1 3 ' '1 2 3 ' '1 3 ' '1 3 2 ' '1 3 3 ' '2 1 ' '2 1 1 ' '2 1 2 ' '2 1 2 1 ' '2 1 3 ' '2 2 ' '2 2 1 ' '2 3 ' '3 1 ' '3 2 ' '3 2 1 ' '3 3 ' '3 3 1 ' '3 3 2 ' '3 3 2 1 ']