Predicting Bankruptcy with Bayesian Networks

Zach Schwark - 2434346
COMS4062A - Probabilistic Graphical Models
University of Witwaterstrand

***Index Terms*—Bayesian Networks, Bankruptcy Prediction, PGM**

I. Problem Statement

Financial risks involved with business ownership and investments are, and will always be prevelent. There is always some probability that the company at hand could go into bankruptcy. There are many different factors involved that influence the likelihood of a company going bankrupt. For instance, there are factors that are exclusive to the company itself such as the cash flow, profitability and liabilities. There are also external factors, such as the economic state. Another example that was very prevalent was the COVID-19 pandemic and it's influence on companies going into bankruptcy.

There are many reasons for why predicting bankruptcy is useful. The first being that entrepreneurs and outside investors will want to know how likely a company is to succeed or fail to see if the company is a good investment opportunity or not. If a certain company is applying for a commercial loan, a bank will also want to assess the likelihood of the company going bankrupt. This is because if the company has a high probability of going bankrupt, it likely lacks sufficient cash flow and profitability to repay a loan, or it may have excessive liabilities. This therefore increases the probability of the company defaulting on the loan and the bank losing money.

II. Data

A. Overview

The data used in the proposed bankruptcy prediction model comprised of financial ratios of different companies for a 5 year period and the classification of if the company was bankrupt or not after the 5 year period [1]. This data specifically was from Polish companies. The data is originally separated by each year, where each year contains 64 financial ratios for the company and the bankruptcy result after the 5 year period. It is noted that the bankruptcy result in each year's data refers to the same result at the end of the 5th year. The 64 different financial ratios that represent the different features are displayed in the table V in the appendix.

B. Preprocessing

There were various preprocessing techniques used to prepare the data before it can be used to create a model to predict bankruptcy. I first had to take into account the fact that there were rows in the data that had missing values. In order to resolve this I decided to remove the rows which had missing values. I decided to do this since I had a sufficient amount of data that allowed to remove rows and still have enough data.

The data was also normalised and standardised in order to make sure the data was within a uniform scale.

Since all of the features in the dataset were financial ratios, the values were continuous. Continuous data is difficult to work with when working with Probabilistic Graphical Models (PGMs). The process of discretising my data involved splitting my data into 20 bins. The strategy I used to discretise my data was to split the data into quantiles, this results in each each bin having roughly the same amount of data points in them. In order to decide on the number of bins and the strategy used, I used scatter plots to plot each feature against the bankruptcy classification in addition with using different coloured rectangles imposed onto the scatter plots to visualise the different bins. This process resulted in being able to visualise how my data is spread and how the data is divide into the different bins. This helped me decide how many bins and which strategy would best discretise my data. These graphs also helped to assess if the normalisation and standardisation were useful and how they affected my data. These graphs are displayed in the appendix as figure 2 and figure 3.

The data was split into a training and testing partitions, I used 75% of the data for the training data and 25% for the testing data.

C. Feature Selection

The data used contained 64 different features which is a substantial amount, therefore selecting which features to use becomes a very useful task. Different methods were used to select the best features. The first technique used was to make use of domain knowledge. The paper [2] performed an empirical analysis on the most useful financial ratios. This paper also discussed the most important financial ratios for predicting firm failure. Based on these ratios I selected certain features from my data. These features are as follows:

- X23
- X1
- X7
- X49
- X9
- X3
- X2
- X59
- X6
- X4
- X51
- X8

The second method I used is based on findings from the paper [3] that used the same data to predict bankruptcy using Ensemble Boosted Trees. This paper discussed the most popular features used by there Ensemble Boosted Trees model for each year of data. I selected the features which were the most popular in all the years. These features are as follows:

- X52
- X40
- X25
- X5
- X9
- X13
- X58
- X15
- X27
- X36
- X22
- X42
- X31
- X48
- X32
- X57
- X12
- X35
- X6
- X53
- X16
- X11
- X55
- X14
- X29
- X41

Finally I used all the features contained in the dataset to explore if there is any benefit in other features or if all features are needed to predict bankruptcy.

I trained and evaluated my model using all three different feature sets to see which feature set resulted the best performance. All the features were used in the final model as I found that this resulted the best model performance.

D. Ethical Considerations

This dataset contained financial data for various Polish Companies. Financial data can be very sensitive and ethical concerns are relevant. The concerns of data privacy and ownership are relevant [4]. Financial data can include personal or private information about the company or owners of the companies. This data is also owned by the relevant people involved. It is unethical to use data that belongs to someone else with out their consent. It is also unethical to expose a person's personal data either through your machine learning (ML) model's outputs or through data analysis techniques. The dataset used in this project only consisted of financial ratios and the bankruptcy classification, therefore there is no data in the dataset that consists of personal or private data. Therefore no processing technique was needed to remove or handle this data.

Another important ethical consideration that needs to be discussed is the topic of outliers in the data. Outliers can drastically affect how the ML model trains and predicts bankruptcy results. An outlier in the data can create a very strong bias and result in the ML predicting the wrong value. This raises ethical concerns because if your ML model is utilized for financial decision-making, its predictions must be accurate to prevent substantial financial losses.

E. Other Thoughts and Considerations

There were various other facts and techniques that were considering during the preprocessing stage. The first being that the data used was very unbalanced between the amount of bankrupt records and non-bankrupt records. In some cases the bankrupt records only made up less than 10% of the non-bankrupt records. This unbalance in the data creates bias when training the model.
The technique of Principle Component Analysis (PCA) was considered to reduce the number of features in the date set. However, this method was discovered to be not useful as PCA will construct new features with the combined information of the original features. This results in losing the meaning and significance of what the features represent. Since the main goal of using a Bayesian Network and PGMs is to achieve interpretable and explainable AI, losing the meaning of a feature goes against our goal.

III. Methodology

This methodology will outline and discuss the method used to construct, train and evaluate my final model. I tried many different techniques and methods before settling on a final model. These techniques and methods will also be discussed in this section. The final model was trained and initially evaluated using only the 5th years data. The model was also tested and evaluated on each year's data after the final model was selected.

A. Model Creation

1) Structure Learning: I tried many different methods for learning the structure of the Bayesian Network and used the method that resulted in less complexity while sill providing better performance. I used a score-based approach to structure learning. The final model was constructed using the Hill Climbing Algorithm with the BIC score.

2) Parameter Estimation: Parameter estimation was done using the Maximum Likelihood Estimator.

3) Other techniques tried: There were many other techniques and methods tried before deciding on the final model. Since most of the data is continuous the use of Gaussian Networks was explored. Gaussian Networks are ideal if the data is continuous, this is because instead of discrete categories as each node, there is a Gaussian

Distribution function. In theory this could result in more accurate predictions since there is less loss of information compared to discretising the data. However, the attempted implementation of a Gaussian Network was not successful and this is due to a few factors. The first being that there are very limited tools and Python libraries available to construct and implement a fully continuous Bayesian Gaussian Network. The implementation that was attempted in this project used the CausalNex library and used a score-based structure learning algorithm for continuous data called NOTEARS. However, the implementation for parameter estimation and inference had to be discretised. The resulted Bayesian Network was very complex and close to a fully connected graph. This resulted in the parameter estimation and inference being too computationally expensive.

Various different structure learning algorithms were tried. These mainly consisted of the score-based approach using the Hill Climbing algorithm. The various scores tested were the K2 score, the BDeu and BIC score.

The different parameter estimation methods tried consisted of the Maximum Likelihood Estimator and the Bayesian Estimator. There were various prior types tried for the Bayesian Estimator and these consisted of the dirichlet and BDeu priors. For the dirichlet prior different values were tried for the pseudo counts. For the BDeu prior different values for the equivalent sample size was also tried.

4) Model Complexity: Since the data set contained 64 different features model complexity is an important consideration. In order to decrease the complexity of the model the BIC score was used when performing the Hill Climbing algorithm during structure learning. The BIC score penalises complex models with a large amount of edges, hence resulting in a less complex model. This also results in certain ”sub-graphs” being independent and not being in the active trail of the target class, and can therefore be removed.

It is also noted the the complexity of the attempted Gaussian Network was significantly more complex than the discrete Bayesian Network produced by the BIC Score and Hill Climbing. This is mainly due to the NOTEARS algorithm used to learn the structure for the Gaussian Network.

5) Selection of Model: In order to select the final model the method of ”trail and error” and experimentation was used. I experimented with the various methods above, with different Bayesian Estimation prior types, different hyper parameter values and various amounts of bins when discretising. I evaluated the performance of the model with different performance metrics and selected the hyper parameters/values that resulted in the best performance.

Another consideration that was used in selecting the model was the model complexity. The BIC score with Hill Climbing resulted in the model that had minimal complexity but still had the best performance.

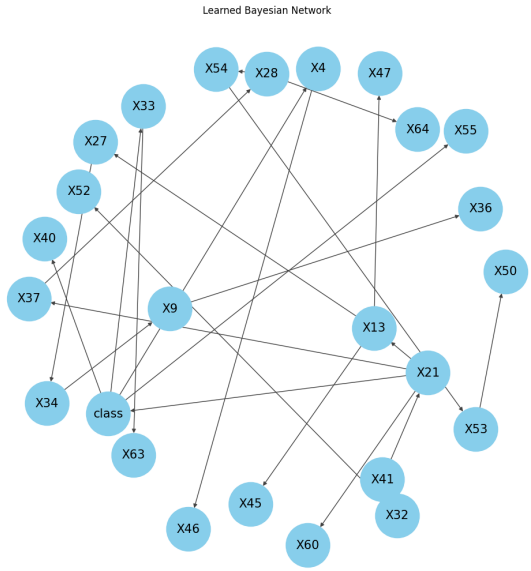

Fig. 1. Final Discrete Bayesian Model Structure

B. Inference and Evaluation

1) Inference: Exact Inference, specifically the process of Variable Elimination was used to find the Conditional Probability Distributions (CPDs) for the nodes in the final model. To obtain the elimination ordering for the variable elimination, the cost function of ”MinFill” was used for the elimination cost, this method uses the number of edges that need to be added to that node in the cost function. The evidence I used in my inference was the values for all the financial ratios for each company in my testing data. I used a MAP query to obtain the classification prediction from my ”class” node.

2) Evaluation: To evaluate the initial model, the model was trained and evaluated once on one set of training and testing data, specifically the 5th year’s data. After the hyper parameters were fine tuned and the final model was decided, 10-fold cross validation was performed. 10-Fold cross validation was performed on each year’s data. This is done, firstly to further evaluate the model on different data. Secondly, to evaluate weather the predictive accuracy increases as the data gets closer to the event of bankruptcy, therefore assessing the impact of data age on bankruptcy prediction.

The performance metrics used were:

- Accuracy Score
- F1 Score
- Precision Score

- Recall Score
- ROC AUC Score
- Balanced Accuracy Score

The macro and weighted average versions of the F1, precision and recall scores were also used. The F1 score was used as it gives a better depiction of your models performance when your data in unbalanced. I used the balanced accuracy score since this also helps assess the model when the data is unbalanced. The weighted average also takes data imbalance into account. [5]

IV. Results

TABLE I
K-fold Cross Validation results for all 5 years.

Scores	***Year 1***	***Year 2***	***Year 3***	***Year 4***	***Year 5***
Accuracy[a]	0.9932	0.9809	0.9781	0.9727	0.9624
Balanced Accuracy[a]	0.0	0.2217	0.0	0.4057	0.7545
F1[a]	0.0	0.2176	0.0	0.4265	0.5819
Precision[a]	0.0	0.2193	0.0	0.5096	0.4762
Recall[a]	0.0	0.2271	0.0	0.4184	0.7863
Weighted F1[a]	0.9898	0.9767	0.9672	0.9718	0.9676
Weighted Precision[a]	0.9865	0.9729	0.9566	0.9729	0.9761
Weighted Recall[a]	0.9932	0.9809	0.9781	0.9727	0.9624

[a]Mean of the scores produced for all folds during the cross validation

A. Analysis of results

Based on the results from table I, the model has a good accuracy. Although we need to remember the imbalance nature of the data set. To do this, we look at the "Balanced Accuracy" score, F1 score, precision and recall. The values for these scores are lower than the normal accuracy score. This shows the imbalanced nature of the dataset, and implies that the model is less accurate when we consider the fact that the data was unbalanced. Therefore, the normal accuracy score value gives a false sense of correctness of the model. However, to further evaluate the model with taking class imbalance into account we look at the weighted F1, precision and recall scores. These scores will give us a more accurate depiction of the correctness of the model since they take into account class imbalance. The values for these scores are high, indicating that even though the data was unbalanced, the model still performs well.

The above table also shows the performance using the different years' data. Looking at the normal F1, precision and recall scores, we can see that the data that is closer to the event of bankruptcy is more accurate at predicting bankruptcy. This implies that it is difficult to predict the event of bankruptcy years in advance when only using the current data available.

Using k-fold cross validation in addition with evaluating the model on different years data aids in performing a sensitivity analysis and assessing the robustness of the model. The accuracy and weighted scores are consistent across the different datasets. The use of the weighted average scores also helps to evaluate the sensitivity of the model to the imbalanced data.

V. Discussion and Conclusion

Throughout the process of developing a PGM for bankruptcy prediction, there were a lot of challenges that were encountered. The first being the continuous nature of the financial data. This was resolved in discretising the data and using a discrete Bayesian Network. However, I firmly believe that it would be better to keep the data continuous as less information in the data would be lost and could potentially result in a model that is a more accurate depiction of the data. The use of Gaussian Bayesian Networks to model the continuous data was discussed previously in this report. There were several limitations that were apparent when Gaussian Networks were attempted. These include the complexity of the model and the fact that there are very limited tools available to develop a fully continuous Gaussian Network. Exploring the development of a Gaussian Network for predicting bankruptcy using solely continuous data could prove to be a valuable area for future research.

Another challenge that was encountered was the presence of outliers in the data. Some of the outliers created very strong biases and affected the structure and nature of my Bayesian Network greatly. I attempted to remove the outliers, but this proved to cause more issues than not removing the outliers. Removing the outliers resulted worse accuracy and worse model performance. It should be noted that the removal of outliers could be very beneficial and result in a better model. However, due to various factors I could not produce a better model with the outliers removed. These factors include the nature of the data, as the data contained outliers that could be responsible for the bankruptcy prediction, and the fact that the data was very unbalanced. Therefore without the outliers, since the data is very unbalanced it was difficult to find patterns in the data to predict bankruptcy.

It can be concluded through this report it was discussed that a PGM, specifically a discrete Bayesian Network can be used to predict bankruptcy of a company using financial data with a fairly high accuracy. The report also showed that the closeness of the financial data to the event of bankruptcy affects the accuracy of the bankruptcy prediction. This indicates that the task of predicting the event of bankruptcy well into the future can be difficult.

References

[1] S. Tomczak, "Polish Companies Bankruptcy," UCI Machine Learning Repository, 2016, DOI: https://doi.org/10.24432/C5F600.

[2] K. H. Chen and T. A. Shimerda, "An empirical analysis of useful financial ratios," *Financial management*, pp. 51–60, 1981.

[3] M. Ziba, S. K. Tomczak, and J. M. Tomczak, "Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction," *Expert Syst. Appl.*, vol. 58, pp. 93–101, 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:40512567

[4] "5 principles of data ethics for business," https://online.hbs.edu/blog/post/data-ethics, (Accessed on 05/13/2024).

[5] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

APPENDIX

X1: Net profit / Total assets

X2: Total liabilities / Total assets

X3: Working capital / Total assets

X4: Current assets / ST liabilities

X5: $\left(\frac{(Cash+STsecurities+Receivables-STliabilities)}{(Operatingexpenses-Depreciation)}\right) \times 365$

X6: Retained earnings / Total assets

X7: EBIT / Total assets

X8: Book value of equity / Total liabilities

X9: Sales / Total assets

X10: Equity / Total assets

X11: $\left(\frac{(Grossprofit+Extraordinaryitems+Financialexpenses)}{Totalassets}\right)$

X12: Gross profit / Short-term liabilities

X13: $\left(\frac{(Grossprofit+Depreciation)}{Sales}\right)$

X14: $\left(\frac{(Grossprofit+Interest)}{Totalassets}\right)$

X15: $\left(\frac{(Totalliabilities\times 365)}{(Grossprofit+Depreciation)}\right)$

X16: $\left(\frac{(Grossprofit+Depreciation)}{Totalliabilities}\right)$

X17: Total assets / Total liabilities

X18: Gross profit / Total assets

X19: Gross profit / Sales

X20: $\left(\frac{(Inventory\times 365)}{Sales}\right)$

X21: $\left(\frac{Sales(n)}{Sales(n-1)}\right)$

X22: $\left(\frac{Profitonoperatingactivities}{Totalassets}\right)$

X23: Net profit / Sales

X24: $\left(\frac{Grossprofit(\text{in 3 years})}{Totalassets}\right)$

X25: $\left(\frac{(Equity-Sharecapital)}{Totalassets}\right)$

X26: $\left(\frac{(Netprofit+Depreciation)}{Totalliabilities}\right)$

X27: $\left(\frac{Profitonoperatingactivities}{Financialexpenses}\right)$

X28: $\left(\frac{Workingcapital}{Fixedassets}\right)$

X29: Logarithm of Total assets

X30: $\left(\frac{(Totalliabilities-Cash)}{Sales}\right)$

X31: $\left(\frac{(Grossprofit+Interest)}{Sales}\right)$

X32: $\left(\frac{(Currentliabilities\times 365)}{Costofproductssold}\right)$

X33: $\left(\frac{Operatingexpenses}{STliabilities}\right)$

X34: $\left(\frac{Operatingexpenses}{Totalliabilities}\right)$

X35: $\left(\frac{Profitonsales}{Totalassets}\right)$

X36: $\left(\frac{Totalsales}{Totalassets}\right)$

X37: $\left(\frac{(Currentassets-Inventories)}{Long-termliabilities}\right)$

X38: $\left(\frac{Constantcapital}{Totalassets}\right)$

X39: $\left(\frac{Profitonsales}{Sales}\right)$

X40: $\left(\frac{(Currentassets-Inventory-Receivables)}{STliabilities}\right)$

X41: $\left(\frac{Totalliabilities}{(Profitonoperatingactivities+Depreciation)\times\frac{12}{365}}\right)$

X42: $\left(\frac{Profitonoperatingactivities}{Sales}\right)$

X43: Rotation receivables + Inventory turnover in days

X44: $\left(\frac{Receivables\times 365}{Sales}\right)$

X45: $\left(\frac{Netprofit}{Inventory}\right)$

X46: $\left(\frac{(Currentassets-Inventory)}{STliabilities}\right)$

X47: $\left(\frac{Inventory\times 365}{Costofproductssold}\right)$

X48: $\left(\frac{(EBITDA(Profitonoperatingactivities-Depreciation))}{Totalassets}\right)$

X49: $\left(\frac{(EBITDA(Profitonoperatingactivities-Depreciation))}{Sales}\right)$

X50: $\left(\frac{Currentassets}{Totalliabilities}\right)$

X51: $\left(\frac{STliabilities}{Totalassets}\right)$

X52: $\left(\frac{(STliabilities\times 365)}{Costofproductssold}\right)$

X53: $\left(\frac{Equity}{Fixedassets}\right)$

X54: $\left(\frac{Constantcapital}{Fixedassets}\right)$

X55: Working capital

X56: $\left(\frac{(Sales-Costofproductssold)}{Sales}\right)$

X57: $\left(\frac{(Currentassets-Inventory-STliabilities)}{(Sales-Grossprofit-Depreciation)}\right)$

X58: $\left(\frac{Totalcosts}{Totalsales}\right)$

X59: $\left(\frac{Long-termliabilities}{Equity}\right)$

X60: $\left(\frac{Sales}{Inventory}\right)$

X61: $\left(\frac{Sales}{Receivables}\right)$

X62: $\left(\frac{(STliabilities\times 365)}{Sales}\right)$

X63: $\left(\frac{Sales}{STliabilities}\right)$

X64: $\left(\frac{Sales}{Fixedassets}\right)$

[1] ST = short-term

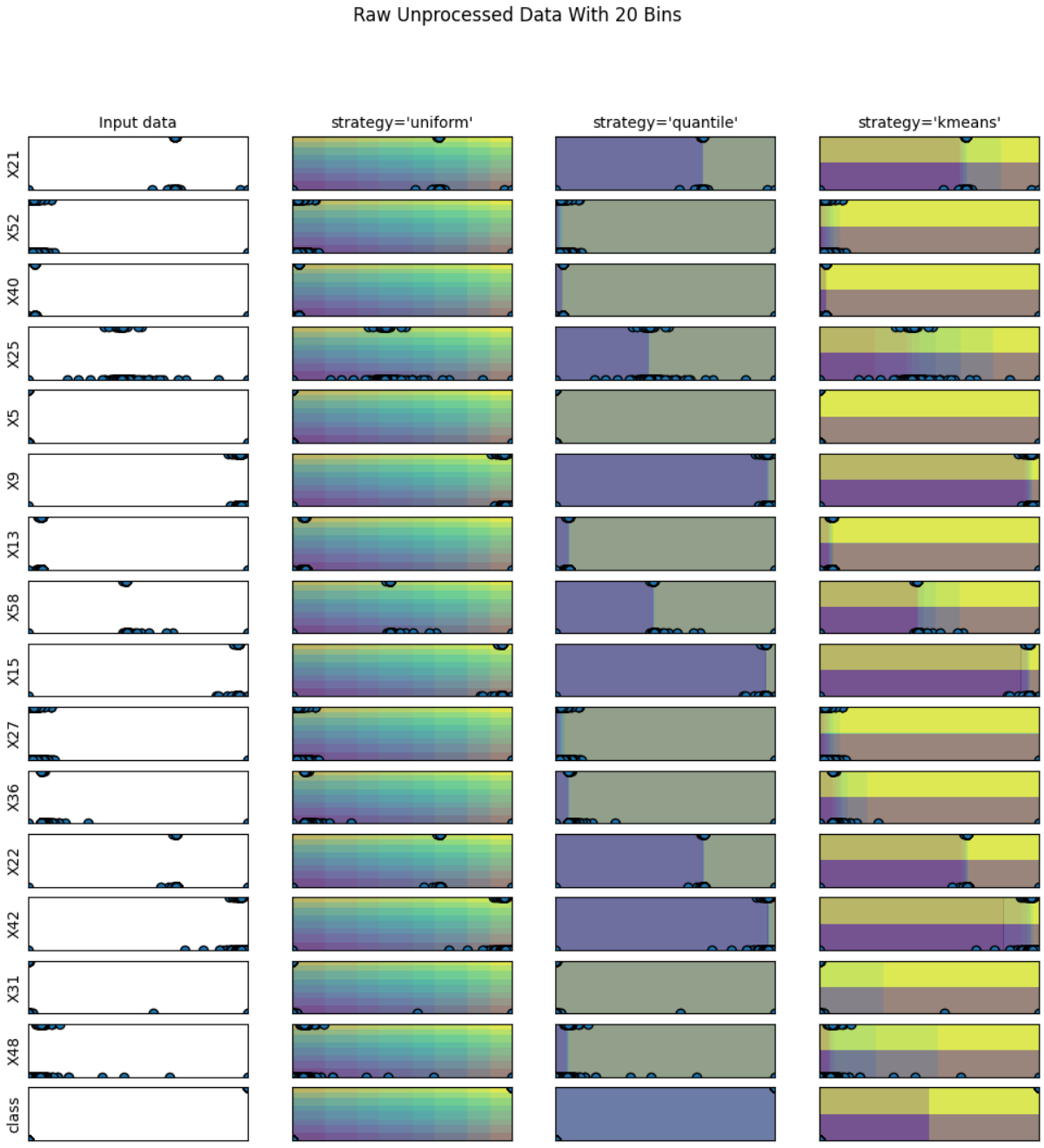

Fig. 2. Raw Unprocessed Data with 20 bins

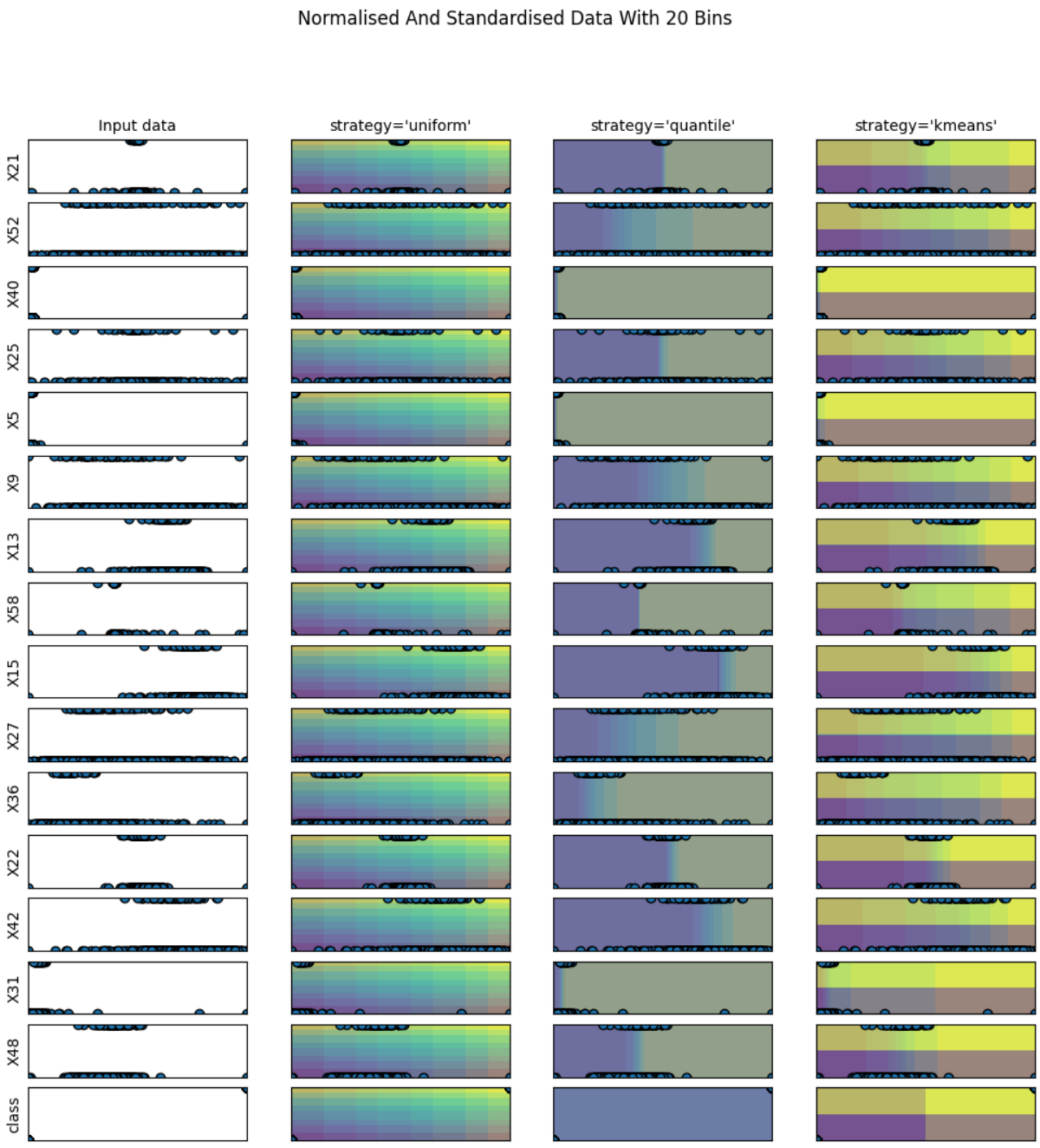

Fig. 3. Normalised And Standardised with 20 bins