Responsible Data Science Project Report

Kevin Fu, Jack Tinker

Background

The life insurance application process is outdated in comparison to the one-click technology of other e-commerce sites, discouraging people from getting life insurance. The purpose of this ADS is, given information about an applicant, to predict a level of risk for life insurance applicants (optimized with respect to the quadratic weighted kappa.) This will streamline the process for consumers and increase sales. Finding a model that accurate classifies risk would also improve public perception of the life insurance industry, encouraging more people to get insurance. The ADS analyzed in this paper was the second-place winner by Bohdan Zhurakovskyi in Prudential's Life Insurance Risk Assessment competition on Kaggle.

Input and Output

The training dataset contains 59381 observations of individuals, over 100 features for each (detailed below), and their corresponding insurance risk score. The test set contains 19765 observations with the same columns, with the exception of the risk score, which is not included. The dataset was provided by the life insurance company Prudential, and is presumably composed of real data from real applications to the company and human-determined risk scores, although Prudential never explicitly states their data collection process.

To preserve privacy, Prudential has anonymized features like "Insurance_History" or "Medical_History"- they are simply numeric values with no indication of what they specifically represent.

The output of the ADS predicts the "response" variable. It is an ordinal measure of the insurance risk of an individual: the highest risk value is "1", and the lowest risk value is "8".

See Table I for detailed feature descriptions.

Figure 1 shows a heatmap of correlation between all features. In general, most features appear to be uncorrelated or weakly correlated. Figure 2 zooms in on the exceptions to this rule. Specifically, we can see that there are strong correlations between the "Insurance_History" features and between some of the "Medical_History" features. Indeed, some very strong negative correlations suggest that some of these features may be mutally-exclusive. It is challenging to make more concrete claims on precisely how these variables are related given the anonymized nature of the data.

Figure 3 shows the distribution of risk across the dataset. It can be seen that there are a strong proportion of high-risk "1"s

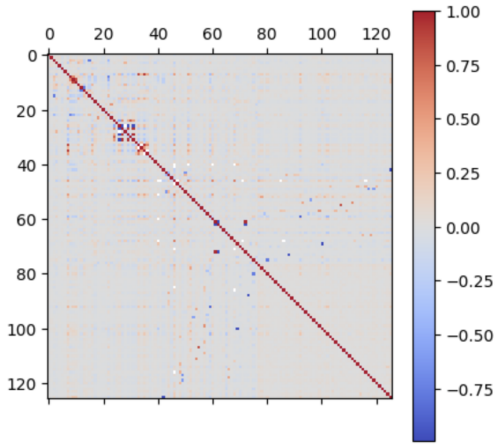

Fig. 1. Heatmap of Feature Correlations

in the dataset, and a downward trend in the number of every risk score after it. Figures 4, 5, 6, and 7 show the distributions of weight, height, BMI, and age, respectively. Each of these metrics has been normalized onto $[0, 1]$ via an undisclosed algorithm. With that in mind, nothing stands out as particularly unusual about these distributions. They appear to capture a standard, real-world distribution of weights and heights. The distributions of Insurance History, Family History, Medical History, etc. were not particularly informative and thus have not been included.

Implementation and Validation

The preprocessing applied to the data is rather minimal.

- **Product_Info_2** is a string consisting of a letter and a number. It is split into two new columns: **Product_Info_2_char** and **Product_Info_2_num**. These are then encoded as integers.
- **BMI** and **age** are multiplied row-wise to make a new column: **BMI_Age**.
- **Med_Keywords_Count** is created as the sum of the values of medical keywords 1-41.
- The number of missing values per row are stored as **count_na**.
- Finally, all NaN values are replaced with -1.

The implementation then uses this data to train a logistic regression, a random forest, thirteen binary XGBoost models (each predicting the probability of risk score equaling 1, equaling 2, being greater than 3, etc.), and a multisoft XGBoost model. These models output risk probability scores for the training and test set.

TABLE I
FEATURE DESCRIPTIONS

Feature Name	Description	Datatype	No. Missing Values
Id	Unique identifier for an application	Integer	0
Product_Info_1-7	Seven normalized variables relating to the product applied for	1, 3, 5-7: integers; 2: string; 4: float	0
Ins_Age	Normalized age of applicant	Float (0.0–1.0)	0
Ht	Normalized height of applicant	Float (0.0–1.0)	0
Wt	Normalized weight of applicant	Float (0.0–1.0)	0
BMI	Normalized BMI of applicant	Float (0.0–1.0)	0
Employment_Info_1-6	Employment history of the applicant	1, 4, 6: float; 2, 3, 5: integer	4: 2137; 6: 3787; Else: 0
InsuredInfo_1-6	Applicant info	Integer	0
Insurance_History_1-9	Insurance history of the applicant	5: float; Else: integer	5: 8105; Else: 0
Family_Hist_1-5	Family history	1: integer; Else: float	1: 0; 2: 9880; 3: 11064; 4: 6677; 5: 13624
Medical_History_1-41	Medical history variables	Integers and floats	1: 2972; 10: 19564; 15: 14864; 24: 18585; 32: 19414; Else: 0
Medical_Keyword_1-48	Binary medical keywords	Binary integer	0
Response	Target variable (final application decision)	Integer (1–8)	0

The train data, *including these probability scores from the four models*, is then used to train a linear regression model to predict risk score. The continuous output of the model is then transformed into the 8 discrete risk scores by threshold cutoffs determined by optimizing on the quadratic weighted kappa via Powell's method. Powell's method is a derivative-free optimization approach, which is necessary in this case since QWK isn't differentiable.

The submissions for the competition were scored based on the quadratic weighted kappa, which was on a scale of 0 to 1, denoting the agreement between the actual ratings and the predicted ratings, with 0 meaning no agreement and 1 meaning complete agreement. Since these are ordinal rankings, QWK is more appropriate than other "all-or-nothing" validation metrics for classification, such as accuracy. QWK essentially gives the model "partial credit" for achieving a risk score that is close to the correct one, and punishes it more heavily for predictions that are far off. A metric like accuracy would treat all incorrect predictions with equal weight, which is not appropriate for ordinal classes.

As mentioned above, the quadratic weighted kappa is the indicator for how well the ADS did. The QWK calculation involves three $N \times N$ matrices: matrix O which captures the frequency of pairs of each actual rating and predicted rating, matrix w which captures the difference in actual rating and predicted rating, and normalized matrix E which captures expected ratings. The formula

$$1 - \frac{\sum_{i,j} w_{i,j} O_{i,j}}{\sum_{i,j} w_{i,j} E_{i,j}})$$

then yields the kappa. This model's QWK score of 0.67921 (on the test set) achieved second place in the competition, indicating that it matches the true labels rather well.

OUTCOMES

Since the test set contains no true labels, we settled for using the training set (which contains true labels) to compute accuracy and fairness metrics. This, admittedly, will produce metrics that are less representative than if they were computed on the test set, but true labels are required to compute metrics like quadratic weighted kappa and accuracy, so we decided to make this compromise for the sake of a more detailed analysis.

As mentioned previously, the dataset used in this competition is heavily anonymized, so it is impossible for us to group the data into "traditional" subpopulations by race, gender, etc. To overcome this, we applied a k-means

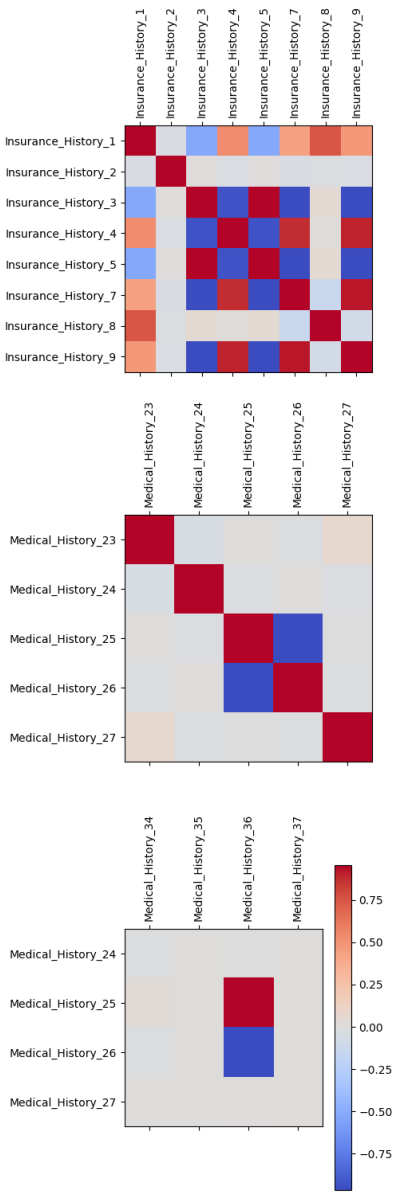

Fig. 2. "Zoomed-In" Feature Correlations

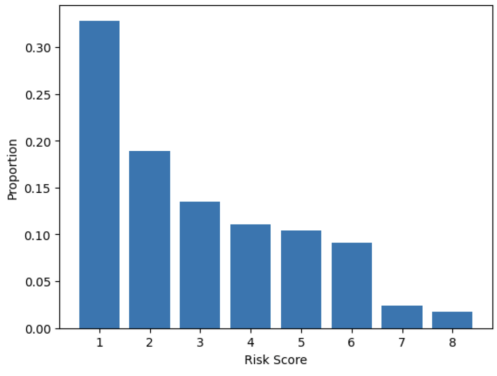

Fig. 3. Distribution of Risk Scores (Train Set)

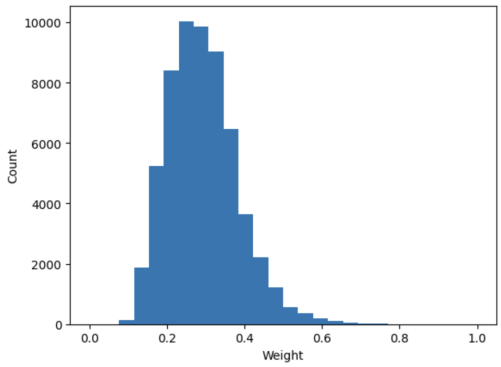

Fig. 4. Distribution of Weight (Train Set)

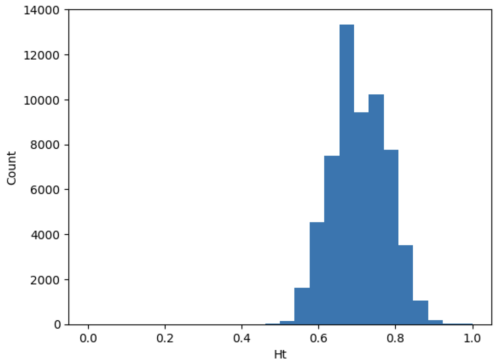

Fig. 5. Distribution of Height (Train Set)

clustering algorithm to extract latent groups in the data without relying on explicitly provided demographics. Using the elbow method, we determined that three clusters was the optimal number in this dataset. Figure 8 shows the inertia for various values of k. We applied our accuracy and fairness metrics to each of the three clusters we found, seen in Figure 14.

To evaluate model accuracy, we used two metrics. We computed a "binary" accuracy by binning the risk labels into two categories: "high risk" for scores below 3, and "low risk"

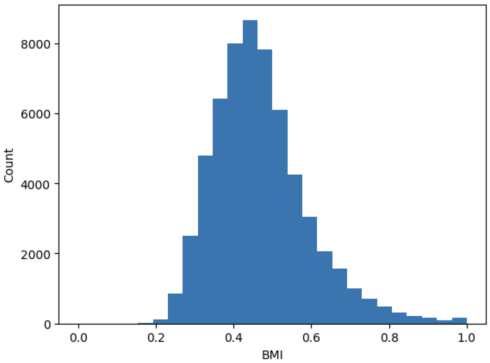

Fig. 6. Distribution of BMI (Train Set)

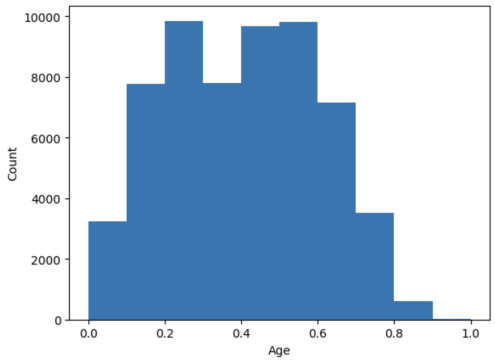

Fig. 7. Distribution of Age (Train Set)

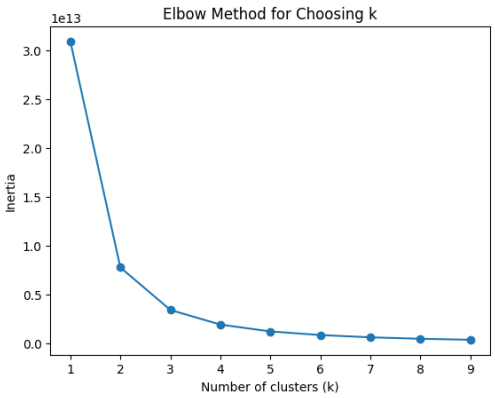

Fig. 8. Interia by value of k

for scores of 3 or greater. We also computed QWK per group. Binning the data in this way to compute accuracy makes it a much more crude measure of performance than QWK, but it is simpler and easier to interpret, so we included it as an option. Using both QWK and "binary" accuracy, it appears that the ADS achieves very similar accuracy across clusters.

To evaluate fairness, we used selection rate, FPR, and FNR. Selection rate, admittedly, is not the best choice of fairness metric here, since differences in selection rate could be attributed to real, objective differences in risk across subpopulations. With this in mind, we included it anyway to provide a more detailed view of the ADS's behavior. FPR and FNR are more useful fairness metrics in this context- FPR measures how likely a group is to be *incorrectly* labelled high risk (in the case of insurance, this is not desirable to the subject being evaluated) and FNR measures how likely a group is to be *incorrectly* labeled low-risk (similarly, in this context, this is desirable to the subject.) Like the accuracy metrics, these fairness metrics appear to be close to equal across clusters. This evaluation shows that both accuracy and fairness are similar across clusters. This is certainly a good sign in terms of fairness for the ADS- but given the anonymized nature of the data, it is impossible to know what these clusters actually represent (if they represent anything meaningful at all), so we hesitate to conclude the ADS is "fair" based on this analysis alone.

We also computed these same metrics binning the groups on two of the few tangible features in the dataset- age and BMI (seen in Figures 15 and 16). Note that age and BMI are both scaled onto $[0, 1]$ via an undisclosed algorithm, and we binned them arbitrarily into three equally sized, fixed width intervals. These bins contain varying numbers of observations and are not based on domain-specific thresholds.

This time, we saw some noticeable differences across groups, both in terms of accuracy and fairness. Like before, binary accuracy and QWK showed similar relative performance, but this time it was not equal across groups. Low age and BMI had the best accuracy, medium age and BMI was just below that, and high age and BMI was a step below that. We see a similar trend in terms of FNR, and an opposite trend in terms of FPR (high BMI/age had the highest FPR, and low BMI/age had the lowest FPR). Both age and BMI are not protected characteristics in the context of health risk assessments, so these disparities are not cause for legal concern. They are, however, ethically troubling, particularly in the case of age, a completely uncontrollable feature. The selection rate being higher for older persons is reasonable—older persons do have an objective heightened risk of health conditions, but it *is* concerning that older persons are far more likely to incorrectly receive an unfavorable outcome than younger people, even if it is legally allowed. It also makes us worry that the algorithm is unfair along other protected characteristics we cannot evaluate due to their exclusion from the data.

Note that we have excluded calibration from this analysis. Calibration would tell us how well the algorithm's probability predictions match the true likelihood of outcomes. While this would be a fantastic metric to analyze fairness across groups and model performance with, this linear regression model outputs onto the entire real number space, rather than probability scores, making calibration a hard metric to capture. Indeed, the lack of probability scores associated with predictions is a mark against the interpretability of this model, which we will explore further in the final section.

Next, we wanted to analyze what features were most important to this model's predictions. At first glance, this model seems very interpretable- it is a linear regression, after all. Of the 142 features included in the regression, the top ten in terms of absolute impact are included in Table II. Of those top 10, 8 are the outputs of the binary XGBoost models, and the other two are vague "insurance history" categories. It appears that the final linear regression is *mostly* ensembling the uninterpretable XGBoost models, essentially nullifying linear regression's major strength in interpretability. Since these XGBoost models are so important to our model's predictions, we performed a SHAP analysis on these much

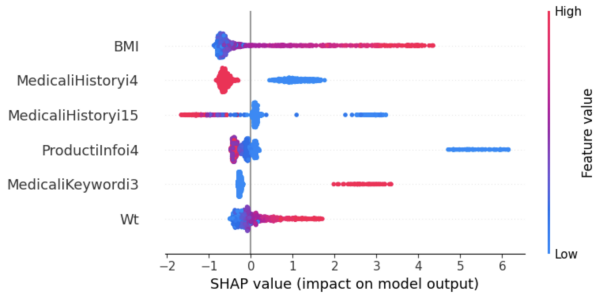

Fig. 9. xgb13 SHAP Values

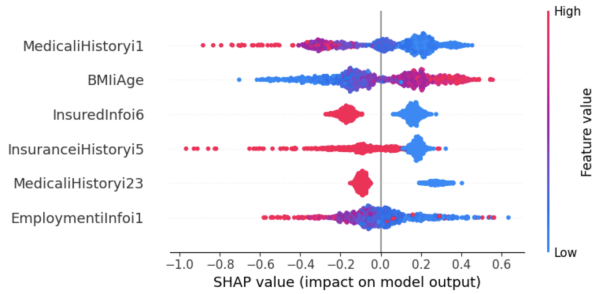

Fig. 10. xgb9 SHAP Values

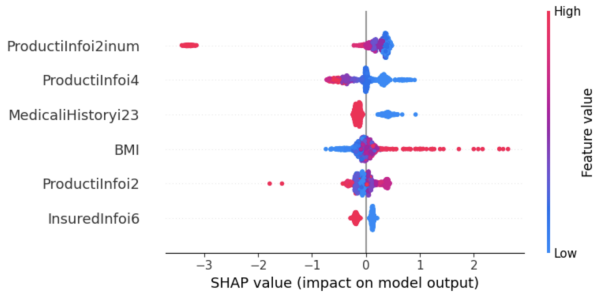

Fig. 11. xgb11 SHAP Values

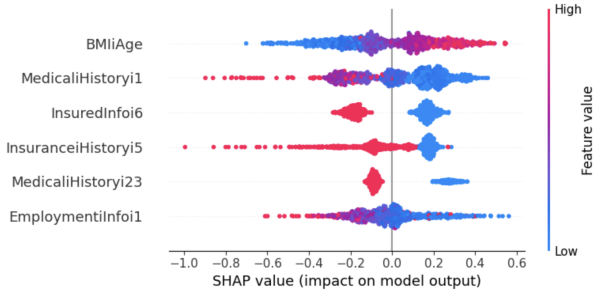

Fig. 12. xgb8 SHAP Values

less interpretable models to see what features were really contributing to the linear regression's predictions.

TABLE II
TOP FEATURES BY ABSOLUTE IMPACT

Name	Coefficient	Absolute Impact
xgb13	-1.610276	1.610276
xgb9	-1.442160	1.442160
InsuranceHistory5	1.368951	1.368951
xgb11	-1.296556	1.296556
xgb8	0.886459	0.886459
xgb1	-0.661225	0.661225
xgb3	-0.657380	0.657380
InsuranceHistory3	-0.609396	0.609396
xgb6	0.475180	0.475180
xgb10	-0.343134	0.343134

Figures 9, 10, 11, and 12 show the plots of SHAP values for the top 4 most relevant XGB models (plots for the next 4 are not included, as they show similar results to these 4). We can see that BMI (or some variant of it, like weight, or BMI·Age) is a very significant contributor to all the models. Interestingly, it is *not* one of the most significant contributors to the linear regression. Apart from BMI, we see ProductInfo, MedicalHistory, InsuredInfo, and MedicalHistory/Keyword features make an appearance in most of the models. When the same one appears in multiple SHAP explainers (such as InsuredInfo6 and EmploymentInfo1 in xgb9 and xgb8) they appear to be contributing in similar ways (that is, a high value of the feature has a positive SHAP value in both models, or vice versa), which raises the question as to if these individual XGBoost models are actually capturing unique insights into the data. To examine this, we checked the correlation between these features. Plotting a correlation matrix of the 13 XGBoost columns reveals that many of these columns are highly correlated, as seen in Figure 13. This level of multicollinearity raises a concern that there may be high variance in the linear regression model. As has been the theme in this report, it is challenging to say anything conclusive about the model's "thought process" despite the SHAP analysis since the feature names are so abstracted.

SUMMARY

The dataset used to create this model seems appropriately large at nearly 60,000 observations, and it has many features to work with including details about the applicant and their

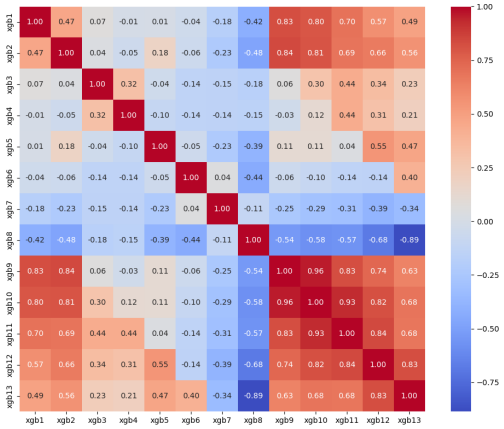

Fig. 13. XGB Column Correlations

medical and employment history. However, the complete lack of metadata surrounding the dataset is a major concern for us. There is no information about how this data was collected, over how long, and from whom. Without this information, there is no way to know if there has been data drift since the collection of this data, or if the data was drawn from a specific subpopulation (i.e. only white people, etc.) We are also a bit concerned about the relevance of the features provided— with vague names like "medical history 4", it is impossible to know what these features actually represent, and if they are truly relevant to the model. Without further information on the data collection method, we cannot claim this dataset was appropriate for this ADS.

Further, based on the distribution of outcomes in the test set, we are concerned that this model may not be robust in real-world deployment. As seen in Figure 3, the highest risk score occupies an overwhelming plurality in the training set. This may be useful if the model is designed to to be very good at detecting high-risk individuals, but, at the admission of Prudential in the challenge description, this model's goal is to classify risk and streamline the application process for *all* life insurance customers, not just very high risk ones, so when the model is deployed and used on real customers who aren't all high risk, it may not perform as well as it does here. We think that a more uniform distribution of risk scores would be more appropriate to allow the model to learn them all better.

Despite this, the implementation does appear to be accurate. It achieved a second place score in the competition using quadratic weighted kappa, a very appropriate metric for this setting, showing that it does match the test data very well. To Prudential, the life insurance provider, this high QWK is very desirable, as it means they will get a good sense of the risk they take on with new clients and can charge them appropriately. In terms of fairness metrics, the model leaves a lot to be desired. The model does have (very near) equal FPR, FNR, and QWK across the latent clusters found by k-means, but the same cannot be said for the groups created on age and BMI. This discrepancy in fairness is concerning, especially considering that we cannot evaluate fairness for other groupings (like race or gender) since the dataset does not contain these labels. To the customer— the subject of the ADS— these imbalances would be very concerning, particularly if you were in one of the disadvantaged groups, like the old-age group. Ultimately, there is reason to believe this model is unfair, and further inspection with the original, deanonymized data is required.

As a consequence of this model's implementation as a linear regression, it does not provide probability scores or any other measure of uncertainty with its predictions. Also, because the model uses a stacked, ensemble approach, explaining any given prediction is a very complicated task, even with tools like SHAP. With these challenges to model interpretability, we worry that the model would not be useful in real-world workflows involving human-AI collaboration, and could potentially *decrease* performance than if it was evaluated by a human alone. Before this model was deployed for use, we would want to perform a study evaluating how useful these predictions actually are in real workflows given the limited explanatory tools.

Considering both the fairness and interpretability concerns we have with this model, we would **not** be comfortable deploying it for use, at least with the current information available to us. To improve this model, we would recommend providing robust metadata on the data collection process, providing demographic information on the subjects to analyze fairness (admittedly, this does raise a data privacy concern, and Prudential could have this information themselves, and simply did not release it for this competition) We also would recommend using a model with more interpretable predictions and features. Ensemble models are powerful, but the XGBoost models used are difficult to interpret, and their extreme levels of multicollinearity make us concerned that this model may not be very robust to new predictions. Switching to a more interpretable model that doesn't include these XGBoost models would foster better human-AI collaboration in real-world workflows and likely improve real-world accuracy.

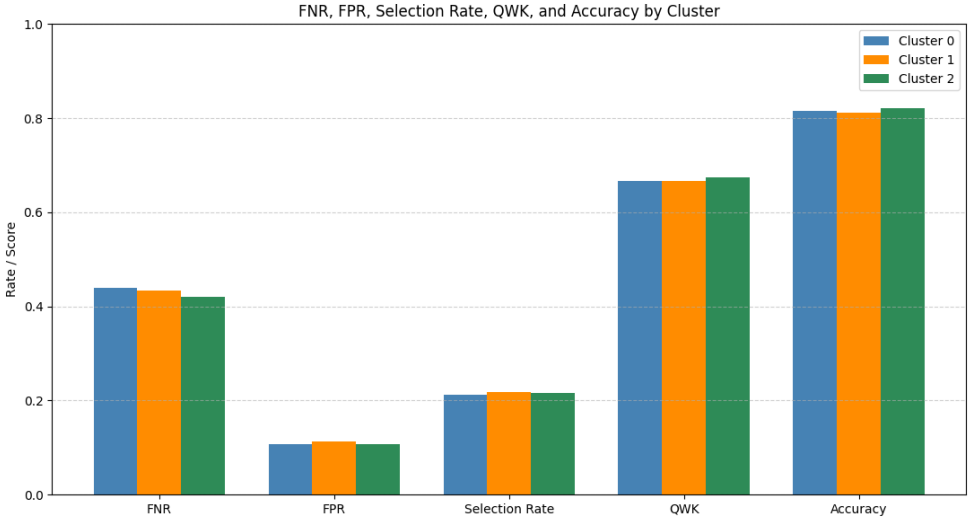

Fig. 14. Metrics by Cluster

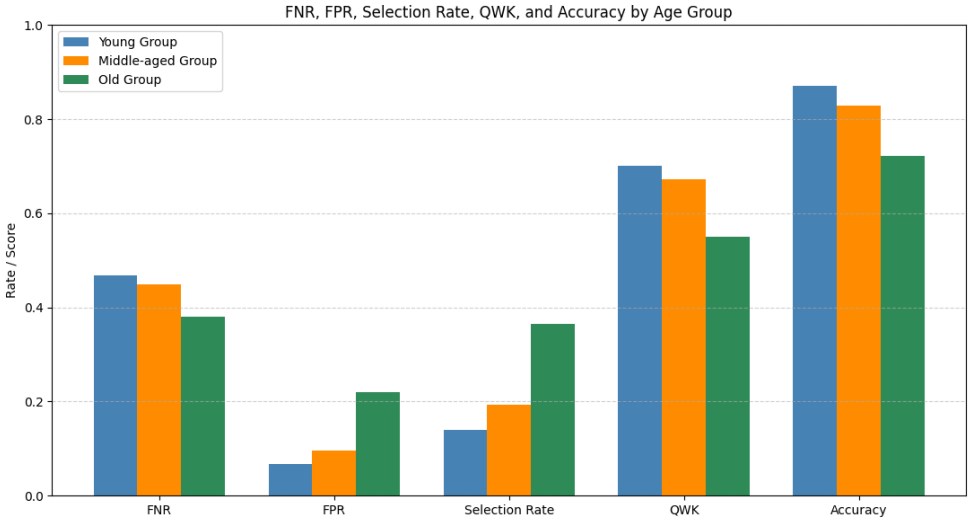

Fig. 15. Metrics by Age

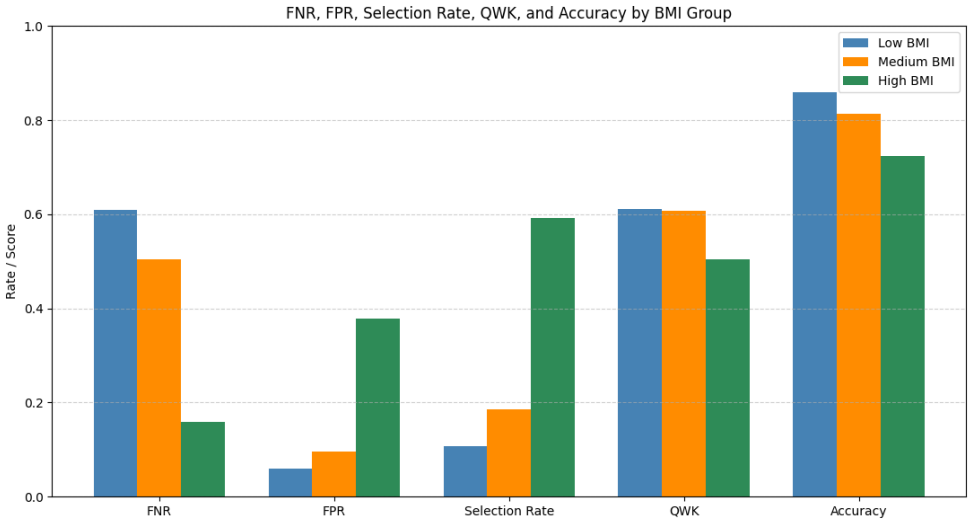

Fig. 16. Metrics by BMI