Maastricht University

Novel Osteoarthritis KLG Classification Using Conformal Prediction

CAPSTONE 3000

Ranjan Mishra, i6210605

Capstone Coordinator
Christof Seiler, Serena Bonaretti

Contents

Novel Osteoarthritis KLG Classification Using Conformal Prediction

9 June 2022

1 Introduction

Over the past 25 years, machine learning techniques have been applied in numerous fields such as medicine, finance, computer science, psychology etc [1]. While these applications have been widely successful, the concept of uncertainty and confidence intervals is still nonexistent among the most popular machine learning libraries even though the notion of uncertainty is fundamental to this field [2]. Machine learning relies on historical data to infer predictions, so these predictions are uncertain at their core [1]. Firstly, the data is often noisy and prone to errors in precision or labelling [1]. Secondly, since the obtained data cannot completely encompass the true nature of an event in the real world, the uncertainties associated with it will be reflected in the resulting model, which will then be propagated to uncertain predictions [1].

Reflecting the uncertainty of a machine learning algorithm is especially important in sensitive fields such as medicine, risk management and autonomous driving [1]. It becomes an integral aspect when these models are used in real-time production, where the impact of a prediction can make the difference between life and death [3]. In medicine, for example, a doctor needs to know about the uncertainty of a prediction beforehand to use it to provide patients with fast, reliable and the most effective disease diagnosis and treatment [3]. Similarly, in risk management, knowing uncertainty can help the end-user understand the best/worst case rewards and make more informed decisions [4]. Moreover, reflecting uncertainty is also a regulatory requirement for high-risk systems such as autonomous driving [1]. Thus, if a machine learning prediction is to have reliability and acceptability among the public, it is of vital importance for it to reflect its uncertainties.

As discussed above, since no top machine learning libraries today quantify the uncertainties of their prediction, the field requires a framework that can help achieve that. Conformal prediction is such an algorithmic framework. It is built on top of traditional algorithms and provides interval estimates of prediction instead of a point/class estimate [4]. The upper and lower bounds of the interval are associated with a significance level which conveys the model's reliability [1]. Moreover, since it is a framework, it can be applied to the classification as well as regression tasks [5].

Therefore, this capstone focuses on applying conformal prediction in the field of biomarker classification for Osteoarthritis. The research question it proposes is: To what extent can the use of conformal prediction quantify the uncertainty in biomarker classification for Osteoarthritis? The capstone first builds on the introduction of conformal prediction. It then breaks down the literature in significant detail, identifying potential research gaps and how this paper contributes to filling that gap. It then introduces the dataset used for analysis. Next, it describes the statistical and machine learning models used to answer the research question. It then presents the results from analysing the dataset and discusses their findings. Finally, the capstone concludes by providing limitations and a roadmap of further research on this topic.

2 Literature Overview

Conformal prediction is an algorithmic framework used on top of traditional machine learning algorithms to provide interval estimates of the predicted variable. There are many advantages of using conformal prediction. First, the framework is consistent, well defined and mathematically robust. Secondly, understanding the interval estimate concept is straightforward and intuitive. For example, if an algorithm is built with a 90% prediction interval, it means that we want to be 90% sure that the true value of the observation lies within our defined interval. It also means that the algorithm will make an error of at most 10%, meaning that there is just a 10% chance of our true value falling outside the obtained interval. As we can see, such an explanation greatly helps in the understanding the uncertainties associated with the algorithm. Finally, the framework is flexible meaning that we can adjust it to fit our specific use cases. For example, we can vary the value of the significance level to extend or narrow the interval [6].

Conformal prediction has been used in medical diagnostics and other subdomains of medicine for a long time [7]. Especially after the advent of Electronic Health Records (EHR), a rich amount of medical data has become digitally available. These data have been used to build models to provide patients with a reliable and effective disease diagnosis. Breast cancer and chronic heart diseases are two such fields where conformal prediction has been used to provide effective diagnosis [3][8]. For these subdomains, conformal predictions have been built on top of traditional algorithms such as Convolutional Neural Networks (CNN), Support Vector Machines (SVM), Random Forest (RF), MultiNomial Naive Bayes (NB), k-Nearest Neighbors (k-NN) etc [9]. The results of these algorithms combined with conformal prediction demonstrate the reliability and usefulness of the obtained confidence measures in disease risk assessment [8].

Finally, in the field of novel osteoarthritis biomarker prediction, a significant amount of academic research has been done by Du et al. (2018) [10]. They have analysed the knee MRI scans and built Convolutional Neural Networks (CNN), Support Vector Machines (SVM), Random Forest (RF) and MultiNomial Naive Bayes (NB) to predict the progression of osteoarthritis. For an accuracy measure, they have used the AUC score [10]. However, it has been considerably difficult to perform a wide scale analysis in the field of osteoarthritis biomarker prediction. This is because data available are very

sparse and collected from hospitals or research institutes. Often these datasets have analysed too few subjects to produce a generalisable machine learning model. Only recently, a significantly large dataset has been made available for Osteoarthritis through the collaboration of National Institute of Health, Zuse Institue of Berlin and the Open Arthritis Initiative [11]. Tack et al. (2021) have performed a significant analysis on this dataset that contains the knee images. Their work assesses the informative value of quantitative features derived from segmentations. It then evaluates the potential of these features as an alternative or extension to CNN-based approaches regarding multiple aspects of osteoarthritis [11]. However, even Tack's work does not utilize the conformal prediction framework for the purpose of osteoarthritis biomarker prediction. So, in this capstone, we aim to contribute to this research gap by analysing the NIH dataset further and quantifying uncertainty in Osteoarthritis biomarker classification.

3 Dataset

The original raw dataset that Tack et al. (2021) use for their analysis contains 46996 knee images collected from 4503 unique subjects [11]. However, for this capstone, we use the derived dataset that contains the quantitative values for the different knee features. The rationale for using the derived dataset was mostly due to the lack of technical knowledge regarding image analysis and effectively analyse the quantitative information contained in them. The values in the derived dataset have been obtained through the processing and analysis of the original knee images in the NIH dataset. Regarding the features of the dataset, it contains 17 columns representing different variables for which the data was collected. Each datapoint is associated with an ID variable, which is a unique identification number for a subject. A subject can have multiple datapoints as they visit at different time intervals. The TIMEPOINT variable represents the different visits and follow-ups that the subject went to. It includes a baseline visit (v00), 1 year follow-up (v12), 2 year follow-up (v24), 3 year follow-up (v36), 4 year follow-up (v48), 6 year follow-up (v72), and 8 year follow-up (v96). Moreover, The LATERALITY variable denotes whether the image was of left or the right knee. The variable of prime importance is the Kellgren-Lawrence grade (KLG), which is also the variable that we aim to predict an interval/class estimate of based on all the other variables. The KLG variable classifies the severity of osteoarthritis. It has five possible values 0(none), 1(doubtful), 2(minimal), 3(moderate) and 4(severe).

Other variables that are of medical importance are LATCOV (Coverage of lateral meniscus on lateral tibia), MEDCOV (Coverage of medial meniscus on medial tibia), LATEXTR (Extrusion of the lateral meniscus), MEDEXTR (Extrusion of the medial meniscus), MTCVOL (Medial tibial cartilage volume), LTCVOL (Lateral tibial cartilage volume), MM_AREA (Medial meniscus area), RATIO_MM (Ratio between surface area and volume for the medial meniscus), LM_AREA (Lateral meniscus area), RATIO_LM (Ratio between surface area and volume for the lateral meniscus), FC_VOLUME (Femoral cartilage volume), TC_VOLUME (Tibial cartilage volume), MM_VOL (Medial meniscus volume) and LM_VOL (Lateral meniscus volume).

4 Methods

To achieve the goals set for our capstone and answer our research question, we employ the methods as discussed in the subsections below.

4.1 Exploratory Data Analysis

First of all, we will perform an extensive exploratory data analysis (EDA) on this relatively new derived dataset. This will be vital in understanding the dimensions of the data and extracting meaningful insights from it [12]. In this part, we will also make use of visual representations to present some of the features explored in the dataset. One such representation will be to build a Markov chain model of disease progression. This will aid in understanding how the osteoarthritis in a subject progresses over different timepoints, e.g. it improves, deteriorates or stays the same. We will also generate subject groups where subjects having the same progression of osteoarthritis are grouped together. Finally, since the dimensions of our dataset is quite large (15), we will also perform principal component analysis (PCA) to understand how the different principal components vary in terms of variance explained [13].

4.2 Machine Learning Algorithms

After completing the EDA, we will train our dataset on the traditional classification algorithms in machine learning. These include Logistic Regression, Gaussian Naive Bayes, K-Nearest Neighbors (kNN), Decision Trees, Support Vector Machines (SVM), Random Forest Classifier and eXtreme Gradient Boosting (XGBoost) [9]. Since our derived dataset is relatively under-investigated, we have selected a wide range of algorithms in order to perform an initial exploration and obtain a best performing baseline. For classification, we use the respective scikit learn packages of the algorithms. For evaluation, we use the classification report demonstrating the precision, recall, F1 score, cross validation and accuracy scores [14].

Moreover, when building our algorithms, we acknowledge that the KLG values are ordinal not nominal. This means that the different KLG values indicate an order, a 4 is greater than a 0, they are not just numbered labels. Furthermore, We will use the results obtained from our algorithms and compare them to the similar work of Tack et al. (2021) and asses whether our work corroborates with their findings [11].

4.3 Conformal Prediction

After building our initial classification algorithms, we will use the conformal prediction framework to obtain the interval estimates of KLG. To build the interval estimates, we will use MAPIE (Model Agnostic Prediction Interval Estimator), a library in Python that estimates prediction intervals [5]. In practise, the MAPIE package works as a wrapper function around our already trained algorithms and we specify an α level. This α determines the significance level of our prediction. Figure 1 below demonstrates the inner working of MAPIE on a classification task.

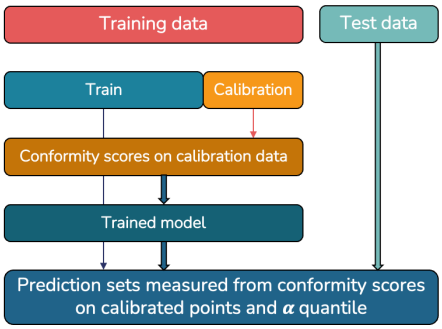

Figure 1: How does MAPIE work on classification? Obtained from `https://mapie.readthedocs.io/en/latest/index.html`

We evaluate the prediction intervals by calculating the effective coverage scores and the mean width of the intervals. The effective coverage score ranges from 0 to 1 and estimates the fraction of true intervals that lie within the prediction intervals [5]. The mean width demonstrates the average of the interval ranges across the entire set [1]. Finally, we present the comparison between the coverage score and mean width interval based on different levels of the α. This will help us understand which value of α to use to get the best combination of a lower mean width and a higher effective coverage.

5 Results

In this section, we show and describe the outcomes of the different experiments that we performed in the methods section.

5.1 Exploratory Data Analysis

5.1.1 Initial Explorations

We started our EDA by exploring the dimensions and the demographics of the dataset. Figure 2 below shows a comprehensive summary of the different demographics within the image dataset analysed by Tack et al. (2021) [11]. Upon comparison, this demographic distribution has remained the same in our derived dataset as well. In their analysis, Tack et al. (2021) did not consider datapoints which had the value NA for the KLG variable. We also do the same in our analysis.

Visit	# Images	Side (left, right)	Sex (male, female)	Age [years]	BMI [kg/m^2]	KLG (0, 1, 2, 3, 4, NA)
v00	9345	4639, 4706	3847, 5498	61.09 ± 9.18	28.59 ± 4.83	3404, 1565, 2329, 1203, 284, 560
v12	8025	4006, 4019	3347, 4678	62.13 ± 9.13	28.44 ± 4.79	2954, 1371, 2103, 1130, 328, 139
v24	7338	3660, 3678	3114, 4224	62.93 ± 9.09	28.39 ± 4.84	2681, 1246, 1914, 1054, 331, 112
v36	5500	2033, 3467	2356, 3144	63.64 ± 9.06	28.38 ± 4.80	1973, 910, 1429, 835, 260, 93
v48	6616	3276, 3340	2819, 3797	64.68 ± 9.06	28.45 ± 4.88	2357, 1072, 1690, 924, 332, 241
v72	5413	2692, 2721	2317, 3096	66.07 ± 8.82	28.25 ± 4.94	1692, 886, 421, 176, 23, 2215
v96	4759	2305, 2454	2055, 2704	67.56 ± 8.64	28.38 ± 5.02	1595, 807, 407, 208, 38, 1704

NA: 'Not available'; no measurement was performed within the OAI study for these knees.

Figure 2: Summary of the demographics of the dataset analysed. Obtained from `https://doi.org/10.1371/journal.pone.0258855.t001`

Based on our EDA, we found that there were 4503 unique subjects on which the data was collected for different variables over the period of a decade. The number of subjects is a considerable improvement in datasets from hospitals relatively few subjects are analysed, making it difficult to generalise those results. Figure 2 also gives us some other interesting insights. For example, the data about different visits decrease significantly as we move further down the timepoints. This might be because of two reasons. Our educated guess is that some subjects discontinued from the study and did not complete entire cycle from v00 to v96. It might also be possible that some subjects passed away during the cycle of the study. The latter might also be possible because the average age of the subjects is around 62.

One of the interesting insights from our EDA was the simultaneous occurrence of duplicate datapoints for the same subject at the same timepoint of the same knee. For these occurrences, the value of the KLG variable was the same, however the measurement of medical variables were different. In total, there were 150 such duplicated instances in our entire dataset. We assumed that this might be because two back to back images of the same knee were taken. So, we kept both the occurrences. Figure 3 below shows an example of a duplicated instance.

```
klg4[klg4["ID"] == 9401202]
```

Python

	ID	TIMEPOINT	LATERALITY	LATCOV	MEDCOV	LATEXTR	MEDEXTR	MTCVOL	LTCVOL	MM_AREA	RATIO_MM	LM_AREA	RATIO_LM	FC_VC
1721	9401202	v00	RIGHT	0.566214	0.255299	-0.018433	1.84618	1867.50	2256.24	2129.77	0.9	2074.70	0.7	17
1722	9401202	v00	RIGHT	0.584801	0.252184	-0.010712	1.48969	1872.06	2252.52	2153.09	0.9	2081.22	0.7	17

Figure 3: Duplicated instance of a data point

5.1.2 Correlation Map

As a next step, we also plotted a correlation heatmap showing the relation of different variables with each other. From our heatmap, we can observe that a majority of the variables seem to have a weak correlation with our predicted variable KLG. Only two variables MEDCOV (-0.41) and MEDEXTR(0.34) have a correlation value greater than 0.2 with KLG. For all the other variables, the correlation scores lie between 0 and 0.2.

Figure 4 below shows the heatmap of the correlation.

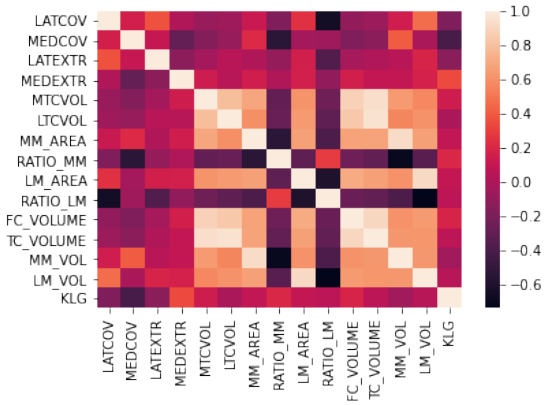

Figure 4: Correlation between different variables in the dataset

5.1.3 Markov Chain Model

Next, to understand our dataset even better, we built a Markov chain model of KLG progression. To define, a Markov chain is a stochastic model in which the probabilities of occurrence of various future states depend only on the present state of the system or on the immediately preceding state and not on the path by which the present state was achieved [15]. To build the model, we grouped the dataset by ID so that entire data for a subject can be gathered at one place. Then from the first occurrence of KLG for an ID, we compared the next occurrence and captured it in a different column. If the KLG has increased, the column value would be positive, negative if decreased, otherwise zero. After that, we made a table showing the previous KLG and the change. This can be seen in Figure 5 below. Finally, we constructed a transition matrix and calculated the probabilities to obtain the Markov chain model fit as seen in Figure 6 below.

CHANGE	-3.0	-2.0	-1.0	0.0	1.0	2.0	3.0	4.0
KLG								
0.0	0	19	65	13141	0	0	0	0
1.0	1	0	63	5845	364	0	0	0
2.0	0	0	41	7334	406	157	0	0
3.0	0	0	44	3671	447	94	45	0
4.0	0	0	0	953	326	16	5	4

Figure 5: Previous KLG level and Offset at current timepoint

Figure 5 can be read as follows. Given a KLG level of 0 at one timepoint, there were no changes at the next timepoint in 13141 observations. In 65 instances, the KLG decreased from 1 to 0 at the next timepoint. Finally, in 19 of the observations, the KLG was previously 2 and became 0 at the next timepoint.

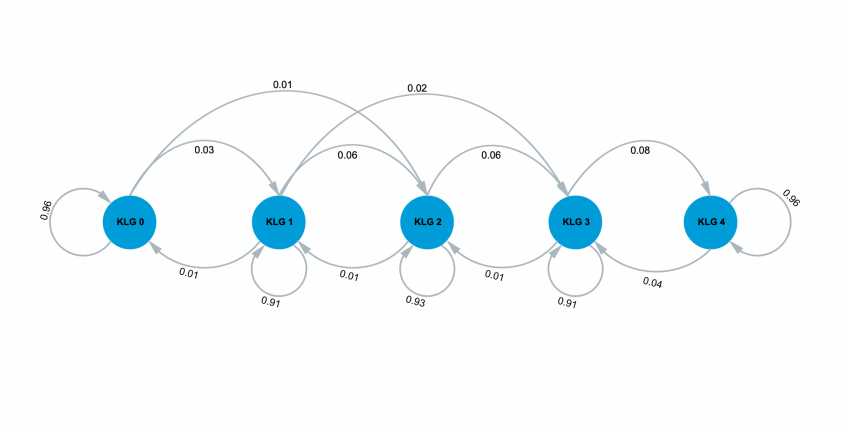

Figure 6: Markov Chain Model Fit depicting the progression of KLG from one timepoint to another

Similarly, Figure 6 can be read as follows. Given a KLG level of 0 at one timepoint, there is 96% probability of it remaining the same at the next timepoint. Similarly, the probabilities of KLG progressing to 1 or 2 at the next timepoint are 3% and 1% respectively.

5.1.4 Subject Groups Generation

As a next step in our EDA, we also explored the possibilities of generating subject groups which have the same progression of KLG over the timepoints where their data were collected. As we can recall from our initial exploration, there were 4503 unique subjects examined in this dataset. So, we made a groupby function that brings together the subjects having the same KLG progression. Figure 7 below shows the plot of the top 20 groups generated ranked in ascending order.

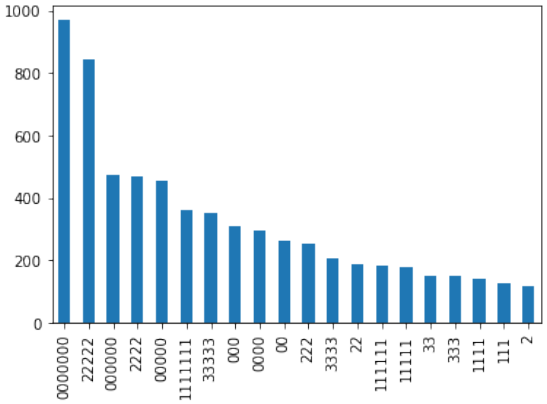

Figure 7: Subject Groups Generation of subjects having the same progression of KLG across all timepoints

Figure 7 can be read as follows. The most popular group is 00000000 which has close to 1000 subjects. Here, the KLG levels remained the same over all the timepoints where data were collected. Second comes the group 22222 where data was only collected for the five timepoints and the KLG levels of all the subjects in this group remained 2 over those timepoints.

5.1.5 Principal Component Analysis

As a final step in our EDA, we performed the Principal Component Analysis (PCA). PCA helps in dimensionality reduction and finds principal components that explain a significant portion of the variance in the dataset [13]. This is very important since the dimension of our dataset is quite large (n=14). So, it becomes important to consider reducing these dimensions and find principal components that have high explanability.

For PCA, we first scaled the dataset using a StandardScaler function from sklearn package. This fits and transforms the dataset into vectors which can then be used for PCA. After performing PCA, we plotted a screeplot that demonstrates the different principal components and the variance explained by them. Figure 8 below shows the screeplot.

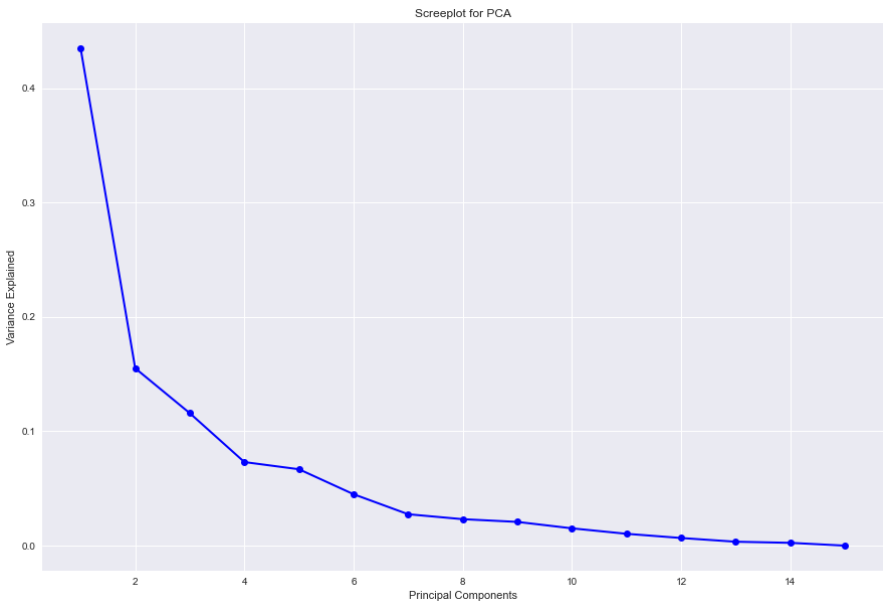

Figure 8: Screeplot showing variance explained by different principal components

As we can see in Figure 8, the first principal component explains about 43% of the total variance, however there is a drastic decrease in the variance explained by the second component at 17%. The trend in decreasing variance explained continues for the remaining principal components as well.

5.2 Machine Learning Algorithms

After completing the EDA, we had gained significant insights into the dataset. So, the next logical step was to use these insights and build our classification algorithms. As discussed in the methods section, we used Logistic Regression, Gaussian Naive Bayes, K-Nearest Neighbors (kNN), Decision Trees, Support Vector Machines (SVM), Random Forest Classifier and eXtreme Gradient Boosting (XGBoost) as our KLG classification algorithms [9]. The motivation was to explore the potential of the most popular classification algorithms and observe how they perform on our dataset.

To start with, we standardized our dataset using the StandardScaler from PCA which limits the data values between 0 and 1. Next, we dropped all the non-numerical values and converted binary variables into 0s and 1s. After that, we split the dataset into training-calibration set and test set with a 2:1 split. Furthermore, we split the training-calibration set into training and calibration using the same 2:1 split. The motivation behind choosing this split was to properly train the model on sufficient instances while still preserving a considerable amount of data for testing [16]. This will help give a better picture of the model in evaluation [14].

Finally, for the evaluation measures, the metrics employed are described as follows. Training accuracy gives a percentage of correct classifications it has done in the training set, whereas the test accuracy accounts for the percentage of correct classifications that the model has done on unseen data [14]. Similarly, precision refers to the proportions of predicted labels that were actually true [14]. Recall refers to the proportion of true labels predicted correctly by the model [14]. F1-score is the harmonic mean of the precision and recall scores [14]. Finally the cross validation score range determines the performance of the model on specific independent test set [14]. Now that we have defined all the metrics and explained our training strategy, the next step is to explain how we used each classification algorithms. All of these classification algorithms have been implemented using their respective scikit-learn packages.

We started with Logistic Regression. To train our Logistic Regression classifier, we trained by setting the following parameters of the model: (penalty: L1, multi_class: ovr, class_weight: balanced, solver: liblinear). Using these values of the parameters helped the model understand that we were performing ordinal classification. Moreover, these parameters were also obtained by continued testing of the different combinations. The evaluation of the resulting classification can be seen in Figure 9 below.

Next, we trained our k-NN algorithm using a maximum of 4 neighbours and our Decision Trees using a max-depth of 5 as parameter values. These values were obtained after performing tuning the parameters and considering the bias-variance tradeoff. In machine learning, the bias-variance tradeoff is the conflict of reducing variance in a model by increasing the bias in the parameters [14].

Moreover, we trained our SVM using a polynomial kernel since we were dealing with a multi-class classification problem. Finally, we trained our XGBoost, Random Forest Classifier and Gaussian Naive Bayes using the default parameters. Here, Gaussian is a good fit from Multinomial Naive Bayes because multinomial NB requires the datapoints to be positive whereas for some of our variables, the values are negative. Finally, The

evaluation of all of these algorithms can also be seen in Figure 9.

Algorithm	Training Accuracy	Test Accuracy	Weighted Precision	Weighted Recall	Weighted F1-Score	5-Fold Cross Validation
Logistic Regression	47.8%	48.3%	0.44	0.48	0.43	43.9% - 49.3%
Gaussian Naive Bayes	42.5%	42.7%	0.38	0.43	0.37	41.9% - 42.2%
K-Nearest Neighbors	68.2%	49.1%	0.48	0.49	0.47	47.0% - 60.2%
Decision Trees	49.3%	48.1%	0.38	0.48	0.40	47.3% - 48.8%
Support Vector Machines	41.4%	42.2%	0.34	0.42	0.29	41.8% - 42.1%
Random Forest Classifier	100%	62.1%	0.64	0.62	0.59	60.1% - 72.5%
eXtreme Gradient Boosting	57.5%	52.4%	0.50	0.52	0.46	52.1% - 53.8%

Figure 9: Evaluation of Different Classification Algorithms

Among all of the classification algorithms, The Random Forest Classifier has performed the best followed by XGBoost and Decision Trees. Moreover, our results did also corroborate with the findings from Tack et al.(2021), although we had a slightly lower accuracy score [11]. This is because Tack et al. (2021) had applied sophisticated bootstrapping methods and parameters tuning which were beyond the scope of our capstone [11].

5.3 Conformal Prediction

After training the classification algorithms, we had achieved a suitable baseline to perform conformal prediction. As discussed in the methods, we trained our MAPIE conformal predictor on calibration data [5]. This gave us the lower and upper bounds of the prediction interval. Then we evaluated the MAPIE predictor on our test dataset and calculated the metrics.

We started by using MAPIE on our best performing classification algorithm, RandomForest. Our initial explorations suggested that for lower values of α our predictor gave very conservative estimates. This can be explained as follows. As we can recall from the definition of conformal prediction, if we use an alpha value of 5%, this translates to having a 95% probability of the true label falling inside our prediction interval [1]. For our RandomForest, this probability was 100%. Even at 5% significance level, all of the true labels were inside our predicted intervals. Normally, we expect at most 5% errors, in our case it is zero, thus this estimate being conservative. Our guess is that this might be because of the distribution of our data and the way in which these algorithms are trained.

So, to get a better estimate, we plotted the coverage score and mean width of the intervals on varying levels of α for our RandomForest classifier. Figure 10 below depicts three plots: coverage scores compared to α levels, mean width compared to α levels and mean width and coverage scores compared to each other.

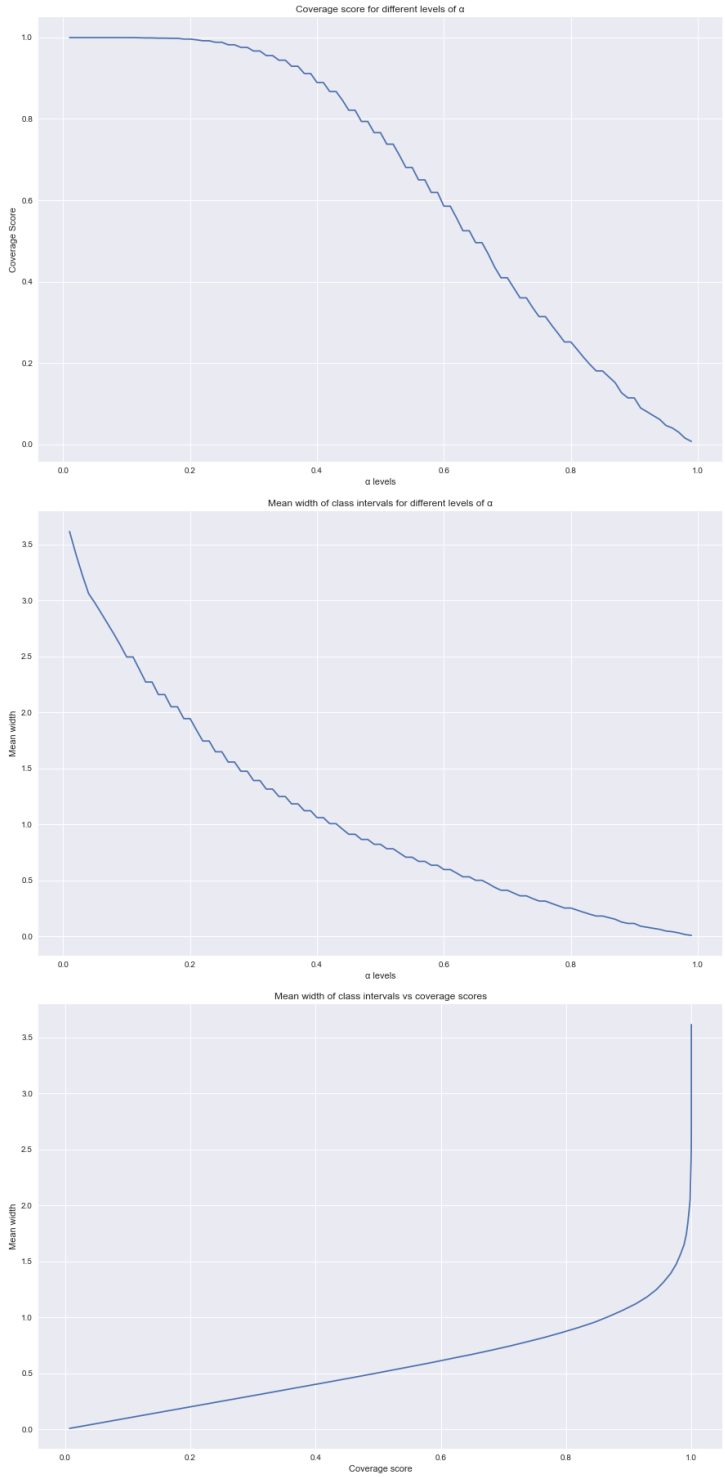

Figure 10: i) Coverage score at different α levels depicting the fraction of true values that lie within the predicted interval ii) Mean Width of intervals across the entire dataset at different α levels iii) Coverage Score and Mean Width Comparison at different levels of α

As we can see in Figure 10 above, up until $\alpha = 0.2$, the effective coverage score is mostly unchanged, which allows us to choose a higher alpha level without compromising on coverage. Secondly, the mean width decreases non linearly as we increase the significance level. Combining both of these, we can obtain a higher coverage score and lower mean width at the same time, as can be seen in the third part of the figure.

So, keeping this initial exploration in mind, for all of the other classification algorithms, we consider three different levels of α and evaluate the coverage scores and mean width of intervals for them. We have arbitrarily chosen α values of 0.1, 0.25 and 0.4. However, the choice of α can be different from these, these are just to explore the progression of the evaluation metrics. Figure 11 below shows the metrics for all the classification algorithms at different levels of α.

Algorithm	α = 0.1		α = 0.25		α = 0.4	
	Effective Coverage	Mean Width of Intervals	Effective Coverage	Mean Width of Intervals	Effective Coverage	Mean Width of Intervals
Logistic Regression	1.000	2.88	0.998	2.12	0.961	1.43
Gaussian Naive Bayes	1.000	3.16	1.000	2.42	0.999	1.74
K-Nearest Neighbors	1.000	5.00	0.999	2.03	0.982	1.70
Decision Trees	0.998	2.90	0.993	1.79	0.988	1.42
Support Vector Machines	0.961	3.12	0.931	2.37	0.888	1.60
Random Forest Classifier	0.999	2.58	0.988	1.64	0.895	1.07
eXtreme Gradient Boosting	0.999	2.71	0.999	1.87	0.978	1.21

Figure 11: Evaluation of MAPIE predictors at different levels of α

From Figure 11 above, we can see that Random Forest Classifier still performs the best overall in terms of achieving a higher coverage while keeping the mean width of intervals low. This is followed by XGBoost and Decision Trees. This corroborates with our point based algorithms as these three were also the best performing among point KLG classification. Here, it is important to notice that since we have 5 classes for our predicted variable KLG, we want to achieve a mean width that is the furthest from value 5 as that will encompass all the classes and it is 100% certain for the true label to lie within one of the five classes.

6 Discussion

6.1 Initial Explorations

In this capstone, we have employed an elaborate set of methods to answer our research question and depicted their findings in the results section. Now, we put the results into context and provide relevant interpretation for them. We start our discussion by acknowledging the complexity of the dataset. As we can see in our correlation heatmap, 12 out of the 14 predictor variables have a very low correlation with our predictor variable. This signified the importance of considering all the variables while building our models, which we did in our analysis.

Similarly our Markov chain model fit suggests that for most subjects, the KLG levels

remains the same when they come for follow up at the next timepoint. This is depicted in the self-probabilities for all the KLG levels being greater than 90%. This effect is also visibly seen in the subject groups generation where all the top 20 groups plotted had the same KLG level over the entire timepoints where their data were collected.

Finally, the complexity of our dataset is also reflected in the PCA screeplot. As we can observe, none of the principal components have an explained variance greater than 50%. The first component has an explained variance of 43%, but it decreases drastically for the second one and so on. So, even through standard scaling, we cannot successfully reduce the dimensions of this dataset without losing a significant proportion of the variance [13]. This further corroborates our hypothesis regarding the complexity of this dataset.

6.2 Machine Learning Algorithms

When we started building our classification algorithms, we first employed a naive approach of only using the highly correlated variables and build our classification algorithms. However, this performed very poorly leading to test accuracy of less than 30% for most of the algorithms. We also explored with the combination of different variables based on the work of Tack et al. (2021), but they did not provide any improvement over the algorithm considering all the variables [11]. So, keeping this exploration in mind, we used the entire set of variables in all of the analyses afterwards.

As we can see in our results, Decision Trees, Random Forest and XGBoost have significantly outperformed other classifiers on this difficult dataset. For Decision Trees, our guess is that it performed well since the dataset had enough training instances to build a reliable rule set and use it to classify KLG [17]. This might also have been aided by the presence of different subject groups, which had same KLG levels and similar measurement of the medical variables. However, why did RandomForest and XGBoost performed even better than Decision Trees? Is there a similarity between these three classifiers? There is. Random Forest and XGBoost both use decision trees as their base learners for the classification, which is why they have improved evaluation metrics over Decision Trees. In particular, RandomForest performed better since it combines different trees (hence called a forest), and uses randomness to enhance its accuracy and prevent overfitting [18]. Similarly, XGBoost also uses decision trees albeit a different version of it, the CART trees (Classification and Regression trees). Instead of containing a single decision in each node, these trees contain the information of whether the particular instance belongs to group or not [19]. It then uses that score and converts them into categories upon reaching the maximum depth of the tree [19]. Utilising this learning technique helps give the XGBoost improved results over the baseline decision trees [19].

6.3 Conformal Prediction

While using conformal prediction also, the top three algorithms from traditional classifiers performed the best in terms of maintaining a good balance between coverage score and mean width. This is because, as explained in the methods, MAPIE is just a wrapper function around the traditional algorithms, it does not fundamentally change the

behavior of the algorithm [1]. So, the accuracy of the algorithm will be propagated when performing conformal prediction as well.

Next, when interpreting the results of the MAPIE, we can see that most algorithms have a very high coverage score at $\alpha = 0.1$. However, these come at a tradeoff of having higher mean width of the intervals, with most algorithms having a mean width between 2 and 3 and for some even reaching 5. However, this is not optimal as we only have five classes and having a mean class width of 5 means that we are considering the entire interval. In that case, there is 100% guarantee of our true label falling in the interval. Moreover, a mean width interval between 2 and 3 is also not optimal as we are still utilising half of our classes for prediction. So, optimally we should aim for a mean width score between 1 and 2. Having a mean width score closer to 1 would mean that we are considering just one class in our interval on average, which can be compared to a point estimate from traditional algorithms [1]. And for our best performing algorithms, this is achieved at the α value of 0.4, where our mean width are 1.07, 1.21 and 1.42 respectively, while still regaining around 90% of the effective coverage. If we increase our α value further, our mean width will further decrease, but this will come at a tradeoff of lower coverage [5]. So, while utilising the MAPIE model, these are the two metrics we should keep in mind when deciding on what value of α to use.

Furthermore, while these estimates for most algorithms are quite conservative, our best guess suggests this might be because of the nature of the dataset and not a fault in employing the methods correctly. This is because we consulted the official MAPIE documentation when applying the Classifier function and followed each and every method in detail [1]. Although our estimates are conservative, there are highly reliable, having coverage score of 90% or better. They are also a significant improvement over the traditional point based classifiers. This reassures the importance of using these classifiers in medical applications.

7 Limitations and Further Research

The goal of this capstone was to perform an extensive exploratory data analysis as well as implement different machine learning algorithms on this derived dataset. We achieved significant success in reaching this goal. However, we were mostly limited by two factors: time and knowledge.

Although four months is an elaborate period of time, combining the capstone with other courses meant that we had to make compromises on the depth of exploration. For example, we could have done a more extensive EDA on the dataset, garnering even more insights, which might be helpful in building algorithms. Furthermore, we could also have considered using the state-of-the-art machine learning techniques like Neural Networks and not be limited to the most popular ones.

Moreover, we were also limited by our knowledge of the subject and different machine learning techniques. For example, I personally did not have a significant background in Biology. This meant that we were analysing the dataset mostly from a data science perspective by following the entire data science pipeline. Had I had more knowledge of

Biology or Osteoarthritis, I could have used that to bring about a different perspective in the analysis. Furthermore, since I had limited knowledge of Image Processing, we could not totally build our models based on the original paper of Tack et al. (2021) [11].

Thus, despite of the limitations above, this capstone presented a genuine attempt to explore the dataset. It is not a standalone work. Future researchers can expand this capstone by conducting an intensive EDA using the knowledge of Osteoarthritis. They can also consider other machine learning techniques not trained in this analysis and examine whether the best performing algorithm still propagates to conformal prediction or not. Moreover, they can also utilise other metrics of evaluation than the ones used in this capstone. This way, combining the work of different researchers through collaboration and building upon them will help fill the research gap in this field and contribute positively to the academic community.

8 Conclusion

To conclude this capstone, we come back to our research question: To what extent can the use of conformal prediction quantify the uncertainty in biomarker classification for Osteoarthritis? From our analysis, we can state that using conformal prediction has brought about significant improvement in quantifying the uncertainty in KLG classification compared to traditional machine learning algorithms. Our best performing traditional algorithm has a test accuracy of 62.1% meaning it will only classify 62.1% of the unseen instances correctly. While this might be a good baseline, our conformal predictors have constantly obtained near perfect coverage scores at lower levels of α. Even at higher levels of α, the mean width was significantly lower while still maintaining a good effective coverage. Finally, using conformal prediction not only quantifies uncertainties in KLG classification, it also aids in the explainability of the algorithm [1]. Since medicine is a sensitive domain where reliability is of paramount importance, using conformal prediction helps decision makers to be certain about their uncertainties [1].

References

[1] Taquet, V. *With MAPIE, uncertainties are back in machine learning*. https://towardsdatascience.com/with-mapie-uncertainties-are-back-in-machine-learning-882d5c17fdc3. 2021.

[2] Alvarsson, J., Arvidsson McShane, S., Norinder, U., and Spjuth, O. "Predicting With Confidence: Using Conformal Prediction in Drug Discovery". In: *Journal of Pharmaceutical Sciences* 110.1 (2021), pp. 42–49. ISSN: 0022-3549. URL: https://www.sciencedirect.com/science/article/pii/S002235492030589X.

[3] Lambrou, A., Papadopoulos, H., and Gammerman, A. "Evolutionary Conformal Prediction for Breast Cancer Diagnosis". In: *2009 9th International Conference on Information Technology and Applications in Biomedicine*. 2009, pp. 1–4. DOI: 10.1109/ITAB.2009.5394447.

[4] Li, S. *An Introduction to Conformal Prediction*. https://towardsdatascience.com/conformal-prediction-4775e78b47b6. 2021.

[5] Taquet, V., Martinon, G., Brunel, N., Ibnouhsein, I., and Deheeger, F. *MAPIE - Model Agnostic Prediction Interval Estimator*. https://github.com/scikit-learn-contrib/MAPIE. 2021.

[6] Norinder, U., Carlsson, L., Boyer, S., and Eklund, M. "Introducing Conformal Prediction in Predictive Modeling. A Transparent and Flexible Alternative to Applicability Domain Determination". In: *Journal of Chemical Information and Modeling* 54.6 (2014). PMID: 24797111, pp. 1596–1603. DOI: 10.1021/ci5001168. eprint: https://doi.org/10.1021/ci5001168. URL: https://doi.org/10.1021/ci5001168.

[7] Vazquez, J. and Facelli, J. C. "Conformal Prediction in Clinical Medical Sciences". In: *Journal of Healthcare Informatics Research* (2022), pp. 1–12.

[8] Lambrou, A., Papadopoulos, H., Kyriacou, E., Pattichis, C. S., Pattichis, M. S., Gammerman, A., and Nicolaides, A. "Assessment of Stroke Risk Based on Morphological Ultrasound Image Analysis with Conformal Prediction". In: *Artificial Intelligence Applications and Innovations*. Ed. by H. Papadopoulos, A. S. Andreou, and M. Bramer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 146–153. ISBN: 978-3-642-16239-8.

[9] Mitchell, T. M. *The discipline of machine learning*. Vol. 9. Carnegie Mellon University, School of Computer Science, Machine Learning ..., 2006.

[10] Du, Y., Almajalid, R., Shan, J., and Zhang, M. "A Novel Method to Predict Knee Osteoarthritis Progression on MRI Using Machine Learning Methods". In: *IEEE Transactions on NanoBioscience* 17.3 (2018), pp. 228–236. DOI: 10.1109/TNB.2018.2840082.

[11] Tack, A., Ambellan, F., and Zachow, S. "Towards novel osteoarthritis biomarkers: Multi-criteria evaluation of 46,996 segmented knee MRI data from the Osteoarthritis Initiative". In: *PLOS ONE* 16.10 (Oct. 2021), pp. 1–17. DOI: 10.1371/journal.pone.0258855. URL: https://doi.org/10.1371/journal.pone.0258855.

[12] Patil, P. *What is Exploratory Data Analysis?* https://towardsdatascience.com/exploratory-data-analysis-8fc1cb20fd15. 2018.

[13] Koech, K. E. *Principal Component Analysis*. https://towardsdatascience.com/principal-component-analysis-ac90b73f68f5. 2022.

[14] Mishra, A. *Metrics to Evaluate your Machine Learning Algorithm*. https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234. 2018.

[15] Merriam-Webster. *Markov chain*. In: *Merriam-Webster.com dictionary*. URL: https://www.merriam-webster.com/dictionary/Markov\%20chain (visited on 06/10/2022).

[16] Galarnyk, M. *Understanding Train Test Split (Scikit-Learn + Python)*. https://towardsdatascience.com/understanding-train-test-split-scikit-learn-python-ea676d5e3d1. 2022.

[17] Chen, L. *Decision Tree Classifier, Explained*. https://medium.com/bite-sized-machine-learning/decision-tree-classifier-explained-9543dd952746. 2018.

[18] Molina, E. *A Practical Guide to Implementing a Random Forest Classifier in Python*. https://towardsdatascience.com/a-practical-guide-to-implementing-a-random-forest-classifier-in-python-979988d8a263. 2021.

[19] T, B. *Beginner's Guide to XGBoost for Classification Problems*. 2021. URL: https://towardsdatascience.com/beginners-guide-to-xgboost-for-classification-problems-50f75aac5390.