AI539 Machine Learning Challenges
Final report and lessons learned

Alena Makarova, ID 934-453-634

March 20, 2023

1 Problem to be solved [1 paragraph]

1.1 What is the prediction problem to be solved?

The activity of ice tongues plays an essential role in the stability of marine-terminating glaciers. However, their behavior is complex and poorly understood, and climate change is likely affecting their strength. The loss of ice tongues can accelerate ice flow into the ocean, leading to sea level rise. In this project, I aim to classify the activity of the glacier's tongue using a 1-digit code. The code includes the following categories from Table 1 below:

Code	Activity
1	Marked retreat
2	Slight retreat
3	Stationary
4	Slight advance
5	Marked advance
6	Possible surge
7	Known surge
8	Oscillating

Table 1: Classes of the *tongue_activity*

Description of each code:
1) Marked retreat: This category refers to a significant and observable retreat of the glacier's tongue over a period of time.
2) Slight retreat: This category refers to a small and observable retreat of the glacier's tongue over a period of time.
3) Stationary: This category refers to the glacier's tongue maintaining a constant position without any significant advance or retreat over a period of time.
4) Slight advance: This category refers to a small and observable advance of the glacier's tongue over a period of time.
5) Marked advance: This category refers to a significant and observable advance of the glacier's tongue over a period of time.
6) Possible surge: This category refers to a situation where the glacier's tongue exhibits unstable behavior that could potentially lead to a surge.
7) Known surge: This category refers to a situation where the glacier's tongue is in a surge phase.

8) Oscillating: This category refers to the glacier's tongue exhibiting a regular or irregular oscillation in its position over a period of time.

To make the predictions, I will analyze factors such as longitude, latitude, mean elevation, and phase of the glacier activity.

1.2 Who are the real (or hypothetical) users or beneficiaries of a solution?

The accuracy of these predictions is crucial for understanding the impacts of climate change on local and global scales, as well as for planning and decision-making related to water resources, sea level rise, and other issues.

2 Data set properties

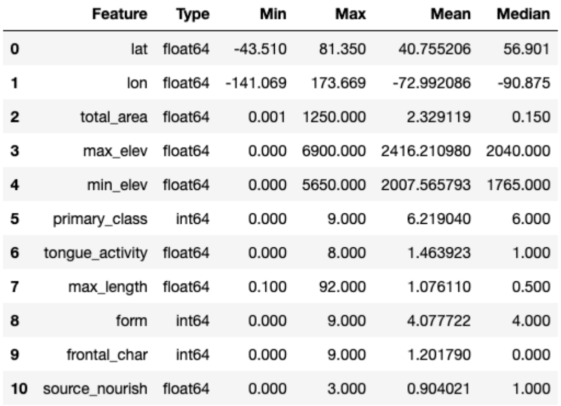
2.1 What is the source of your data set? Include a citation that specifies at least the author(s) and URL. [1 short paragraph]

I have used the World Glacier Inventory (WGI) data set, which is available on the National Snow and Ice Data Center (NSIDC) website. WGMS (World Glacier Monitoring Service). 2012. World Glacier Inventory. Compiled and made available by the World Glacier Monitoring Service, Zurich, Switzerland, and the National Snow and Ice Data Center, Boulder CO, USA. `https://nsidc.org/data/glacier_inventory/`.

2.2 Data set profile: number of items, class distribution, type of features, min/max/mean/mean or distribution for each feature, etc. [length depends on the number of features in your data set]

The data set contains 30725 items, 14 features, and one target feature.

Below we can observe the main parameters of the features:



	Feature	Type	Min	Max	Mean	Median
0	lat	float64	-43.510	81.350	40.755206	56.901
1	lon	float64	-141.069	173.669	-72.992086	-90.875
2	total_area	float64	0.001	1250.000	2.329119	0.150
3	max_elev	float64	0.000	6900.000	2416.210980	2040.000
4	min_elev	float64	0.000	5650.000	2007.565793	1765.000
5	primary_class	int64	0.000	9.000	6.219040	6.000
6	tongue_activity	float64	0.000	8.000	1.463923	1.000
7	max_length	float64	0.100	92.000	1.076110	0.500
8	form	int64	0.000	9.000	4.077722	4.000
9	frontal_char	int64	0.000	9.000	1.201790	0.000
10	source_nourish	float64	0.000	3.000	0.904021	1.000

Figure 1: Feature description

Let's see which features we have to analyze:

lat - the latitude of the glacier in decimal degrees North or South; up to 7 digits. Positive values indicate the Northern Hemisphere, and negative values indicate the Southern Hemisphere. Latitude is given to a maximum precision of 4 decimal places."The point on the glacier whose coordinates given should be in the upper part of the ablation area, in the mainstream and sufficiently high so as not to be lost if the glacier retreats" (Müller et al. 1977).

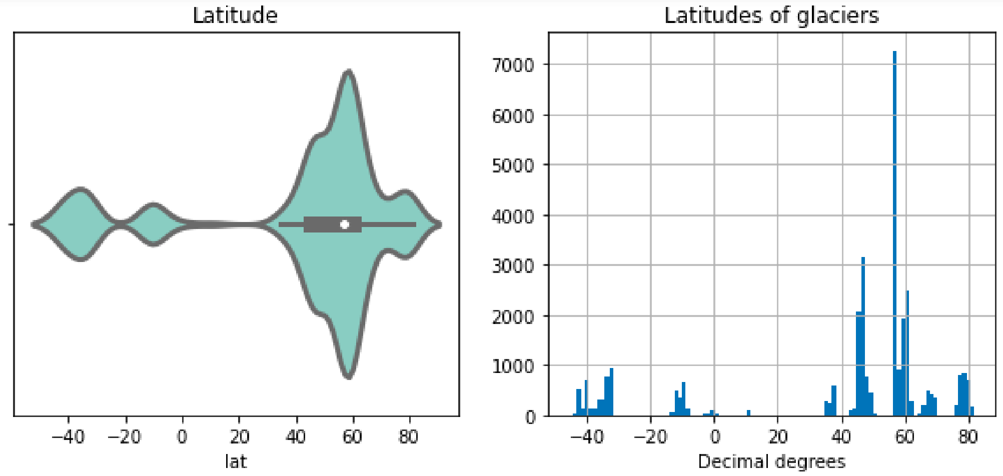


Figure 2: Latitude

lon - the longitude of the glacier in decimal degrees East or West; up to 7 digits. Positive values indicate east of the zero meridians, and negative values indicate west of the zero meridians. Longitude is given to a maximum precision of 4 decimal places.

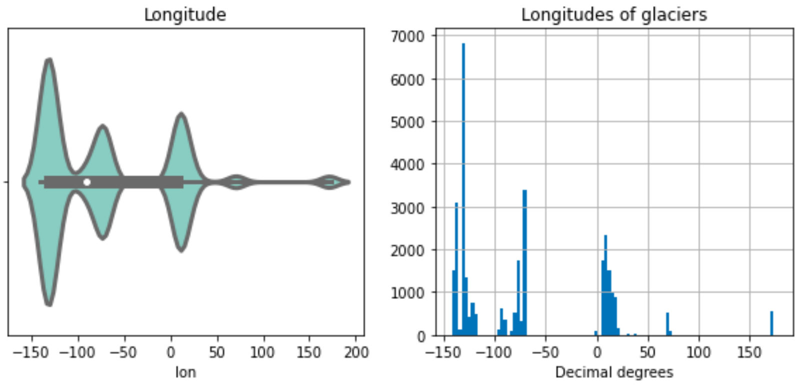


Figure 3: Longitude

total_area - The total area of the glacier in a horizontal projection in square kilometers, up to 6 digits.

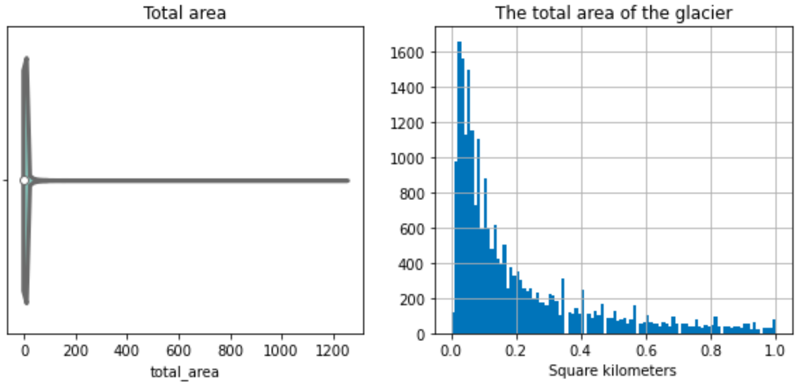


Figure 4: Total area

max_elev - maximum elevation of the highest point of the glacier in meters above sea level, up to 4 digits.

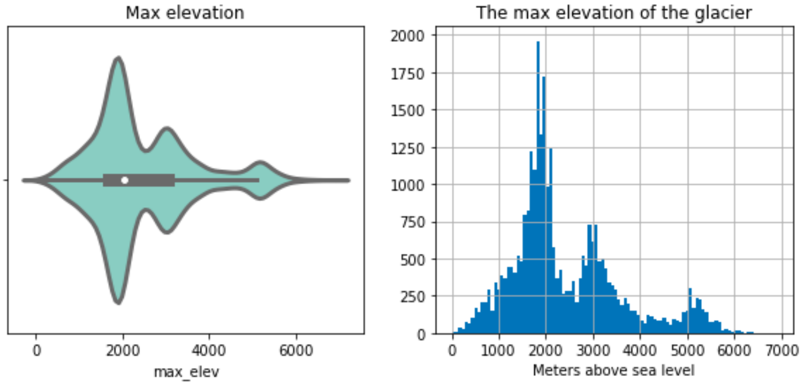


Figure 5: Maximum elevation

min_elev - the minimum elevation of the lowest point of the glacier in meters above sea level, up to 4 digits.

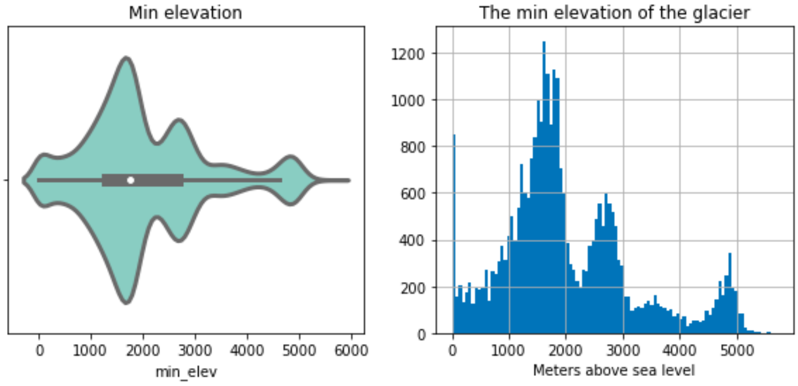
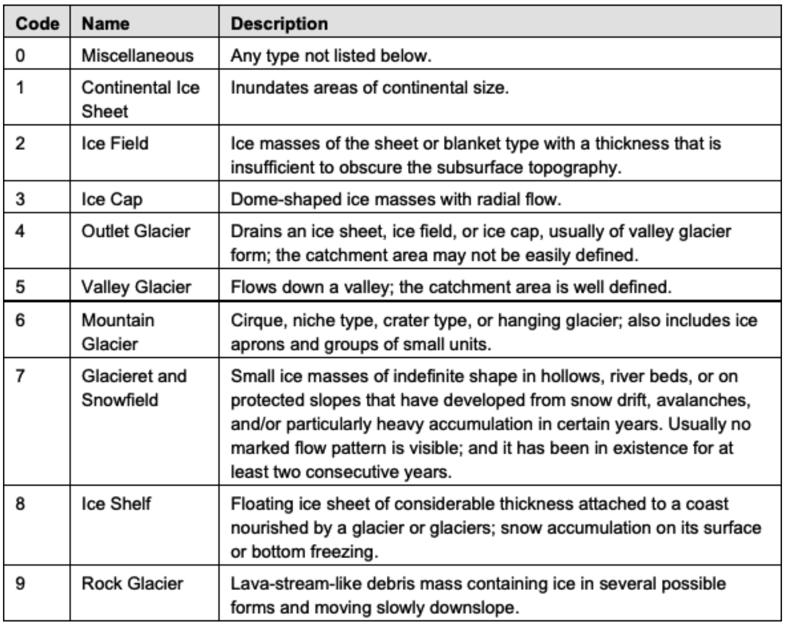


Figure 6: Minimum elevation

primary_class - a 1-digit code that describes the primary classification of the glacier. The codes are described in Figure 7.



Code	Name	Description
0	Miscellaneous	Any type not listed below.
1	Continental Ice Sheet	Inundates areas of continental size.
2	Ice Field	Ice masses of the sheet or blanket type with a thickness that is insufficient to obscure the subsurface topography.
3	Ice Cap	Dome-shaped ice masses with radial flow.
4	Outlet Glacier	Drains an ice sheet, ice field, or ice cap, usually of valley glacier form; the catchment area may not be easily defined.
5	Valley Glacier	Flows down a valley; the catchment area is well defined.
6	Mountain Glacier	Cirque, niche type, crater type, or hanging glacier; also includes ice aprons and groups of small units.
7	Glacieret and Snowfield	Small ice masses of indefinite shape in hollows, river beds, or on protected slopes that have developed from snow drift, avalanches, and/or particularly heavy accumulation in certain years. Usually no marked flow pattern is visible; and it has been in existence for at least two consecutive years.
8	Ice Shelf	Floating ice sheet of considerable thickness attached to a coast nourished by a glacier or glaciers; snow accumulation on its surface or bottom freezing.
9	Rock Glacier	Lava-stream-like debris mass containing ice in several possible forms and moving slowly downslope.

Figure 7: Primary class codes
World Glaciers Inventory, Version 1
$https : //nsidc.org/sites/default/files/g01130 - v001 - userguide\_1\_0.pdf$

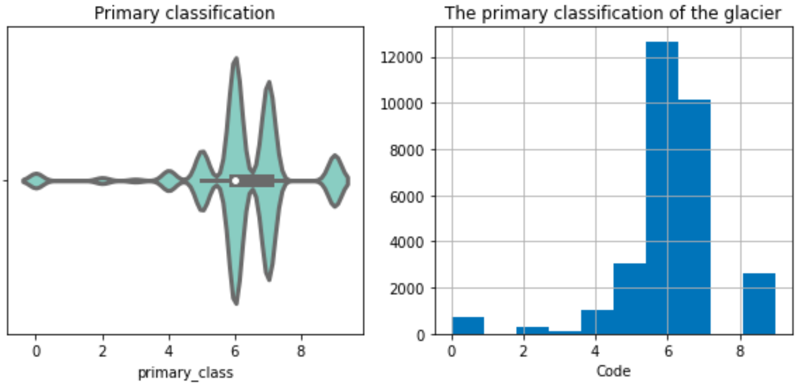


Figure 8: Primary class codes distribution

tongue_activity - a 1-digit code that describes the activity of the tongue of the glacier. Table 2 lists the tongue activity codes.

Code	Activity
0	Uncertain
1	Marked retreat
2	Slight retreat
3	Stationary
4	Slight advance
5	Marked advance
6	Possible surge
7	Known surge
8	Oscillating

Table 2: Classes of the *tongue_activity*

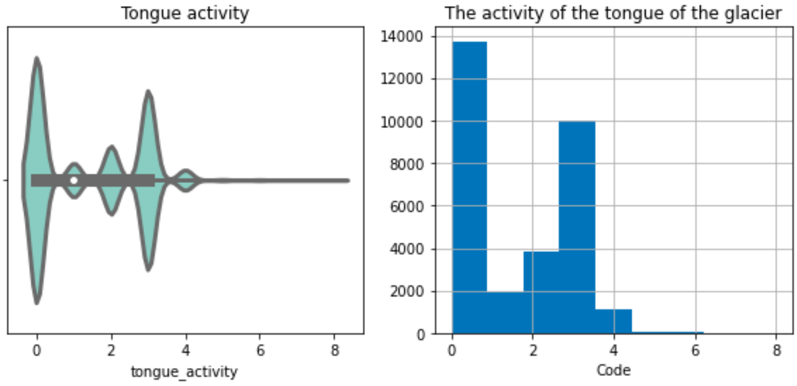


Figure 9: Tongue activity distribution

Having the longitude and latitude, I could plot the classes of the glaciers tongue activity (Figure 10) to explore the different regions and see if there are any anomalies.

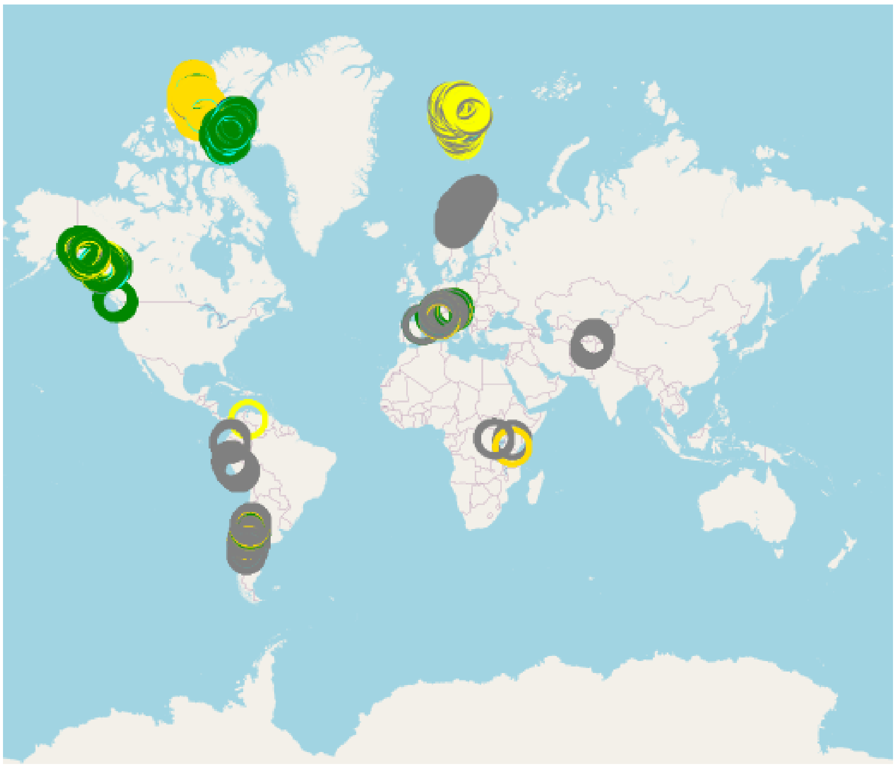
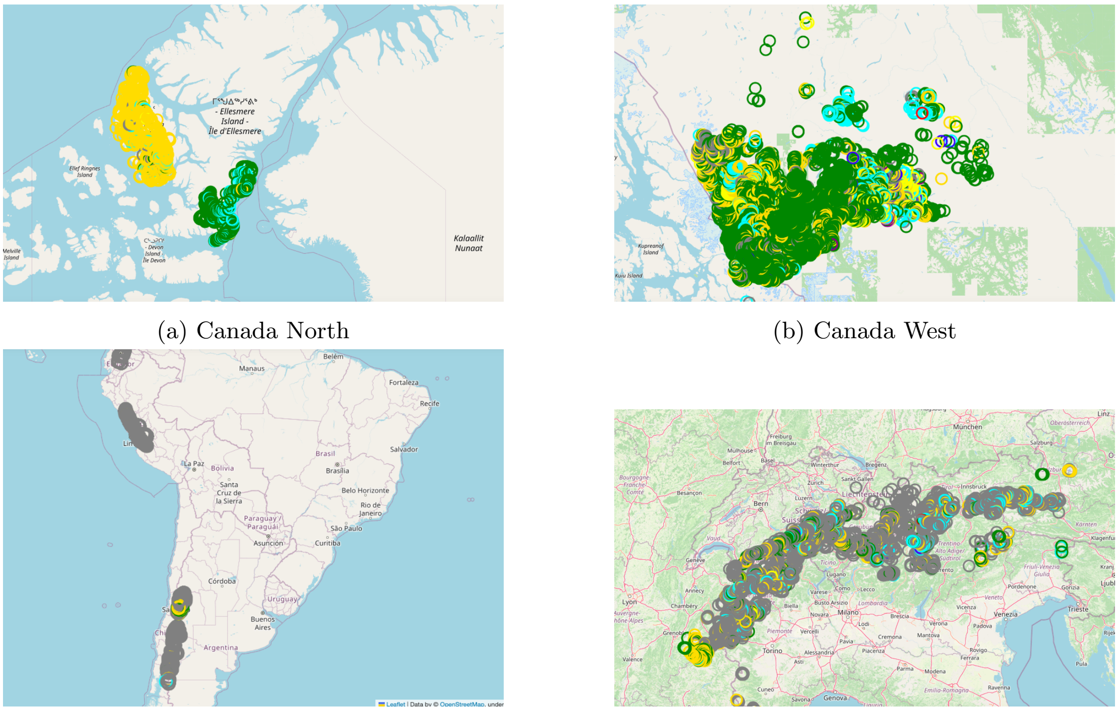


Figure 10: Classes of the tongue activity

I can observe the evident class imbalance as I have 9 classes with corresponding colors:
0 - Uncertain - gray;
1 - Marked retreat - yellow;
2 - Slight retreat - gold;
3 - Stationary - green;
4 - Slight advance - cyan;
5 - Marked advance - blue;
6 - Possible surge - purple;
7 - Known surge - orchid;
8 - Oscillating - red;

Three classes - uncertain, retreat and stationary represent the major part of the glaciers on the map, while others are minor. Let's take a look on other regions to see if there is anything else.



(a) Canada North

(b) Canada West

(c) Latin America

(d) Central Europe

Figure 11: Classes of the glacier's tongue activity

If to increase the scale, we can observe that Canadian samples are mostly Marked retreat, Stationary, and Slight advance with a small amount of Marked advance and Oscillating glaciers. At the same time, Latin America and Central Europe region has more uncertain values. Probably, the reason could be that the World Glaciers Inventory doesn't have any information about other areas besides Canada. Or maybe, the scientists couldn't agree how to classify the glacier, so they decided to mark it as uncertain.

max_length - the maximum length of the glacier in kilometers measured along the most important flowline in a horizontal projection, up to 4 digits.

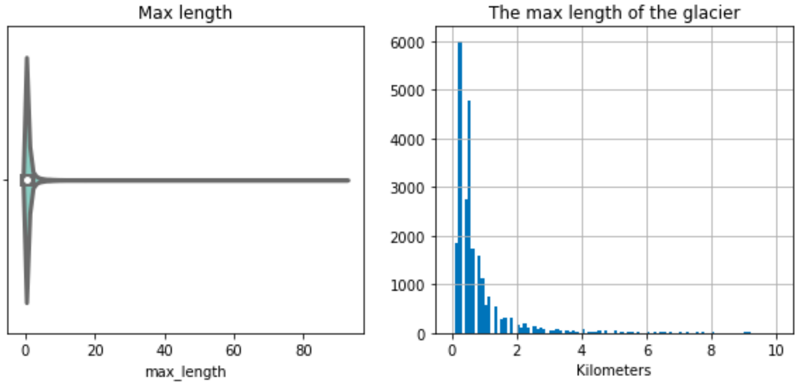


Figure 12: Max length

form - a 1-digit code that describes the form of the glacier. Figure 14 describes the glacier form codes.

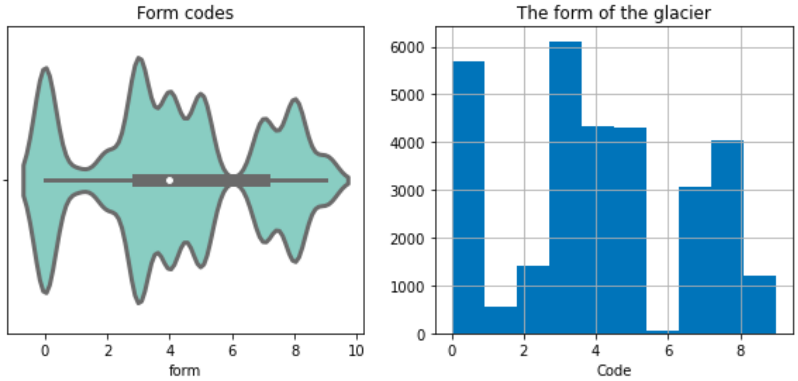
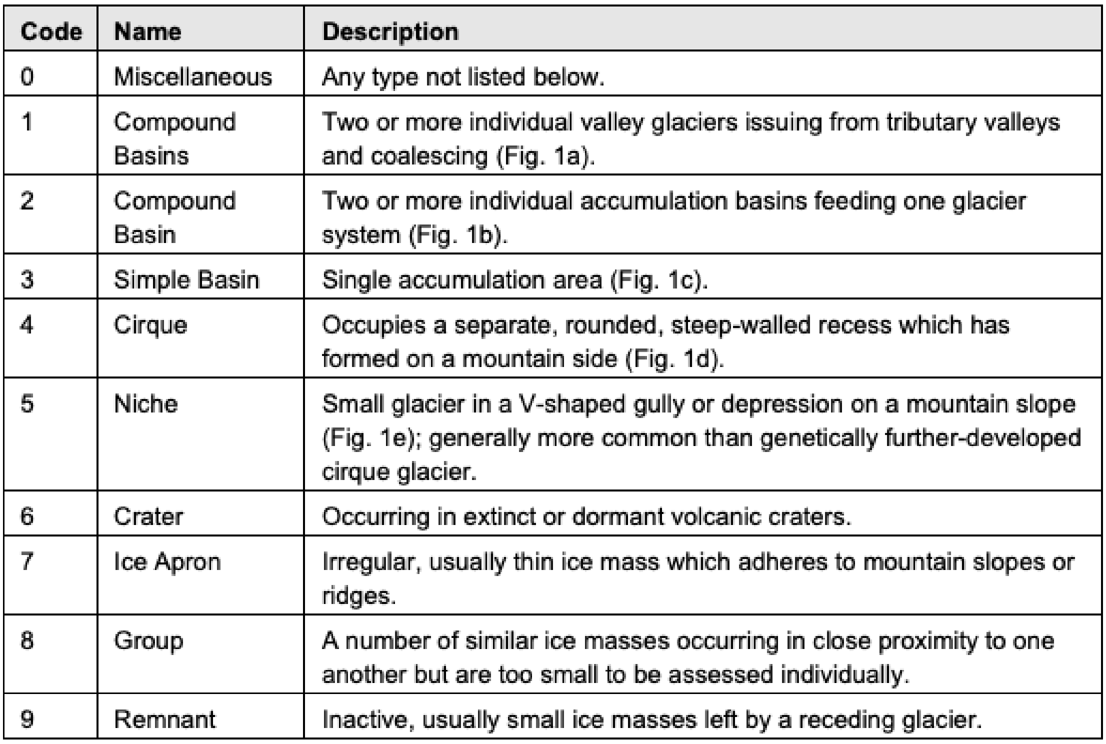


Figure 13: Form



Code	Name	Description
0	Miscellaneous	Any type not listed below.
1	Compound Basins	Two or more individual valley glaciers issuing from tributary valleys and coalescing (Fig. 1a).
2	Compound Basin	Two or more individual accumulation basins feeding one glacier system (Fig. 1b).
3	Simple Basin	Single accumulation area (Fig. 1c).
4	Cirque	Occupies a separate, rounded, steep-walled recess which has formed on a mountain side (Fig. 1d).
5	Niche	Small glacier in a V-shaped gully or depression on a mountain slope (Fig. 1e); generally more common than genetically further-developed cirque glacier.
6	Crater	Occurring in extinct or dormant volcanic craters.
7	Ice Apron	Irregular, usually thin ice mass which adheres to mountain slopes or ridges.
8	Group	A number of similar ice masses occurring in close proximity to one another but are too small to be assessed individually.
9	Remnant	Inactive, usually small ice masses left by a receding glacier.

Figure 14: Form codes
World Glaciers Inventory, Version 1
https : //nsidc.org/sites/default/files/g01130 − v001 − userguide_1_0.pdf

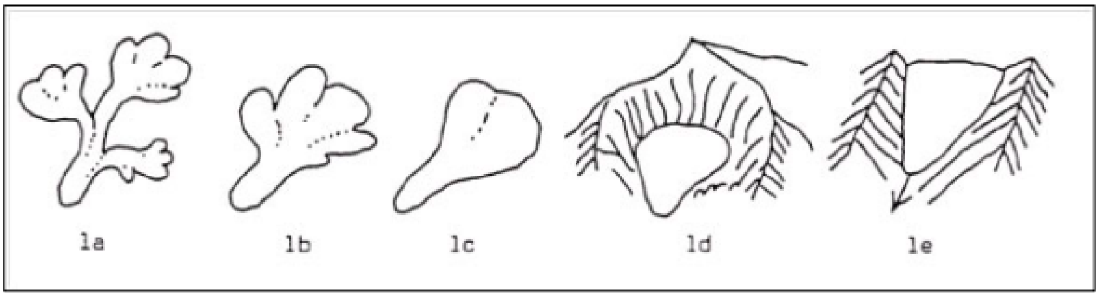


Figure 15: Glacier forms
World Glaciers Inventory, Version 1
https : //nsidc.org/sites/default/files/g01130 − v001 − userguide_1_0.pdf

frontal_char - a 1-digit code that describes the frontal characteristics of the glacier. Table 5 lists the frontal characteristic codes. Figure 14 describes the frontal glacier characteristics codes.

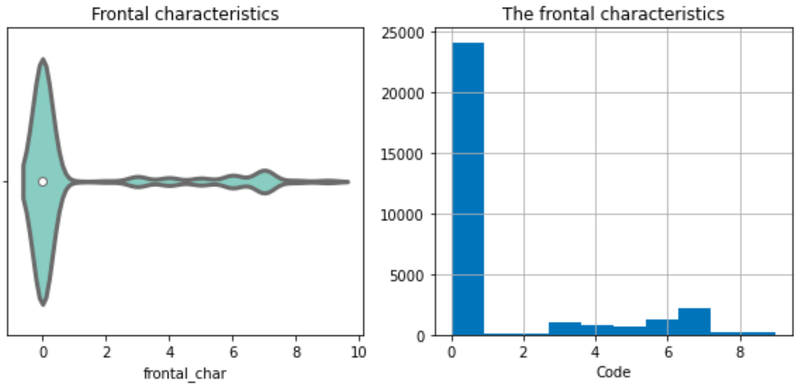
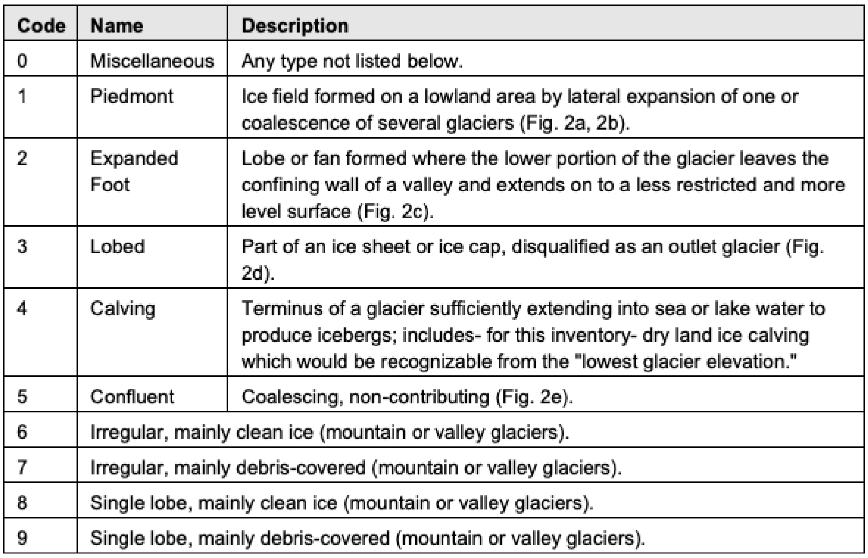


Figure 16: Frontal characteristics



Code	Name	Description
0	Miscellaneous	Any type not listed below.
1	Piedmont	Ice field formed on a lowland area by lateral expansion of one or coalescence of several glaciers (Fig. 2a, 2b).
2	Expanded Foot	Lobe or fan formed where the lower portion of the glacier leaves the confining wall of a valley and extends on to a less restricted and more level surface (Fig. 2c).
3	Lobed	Part of an ice sheet or ice cap, disqualified as an outlet glacier (Fig. 2d).
4	Calving	Terminus of a glacier sufficiently extending into sea or lake water to produce icebergs; includes- for this inventory- dry land ice calving which would be recognizable from the "lowest glacier elevation."
5	Confluent	Coalescing, non-contributing (Fig. 2e).
6	Irregular, mainly clean ice (mountain or valley glaciers).	
7	Irregular, mainly debris-covered (mountain or valley glaciers).	
8	Single lobe, mainly clean ice (mountain or valley glaciers).	
9	Single lobe, mainly debris-covered (mountain or valley glaciers).	

Figure 17: Frontal Codes
World Glaciers Inventory, Version 1
https : //nsidc.org/sites/default/files/g01130 − v001 − userguide_1_0.pdf

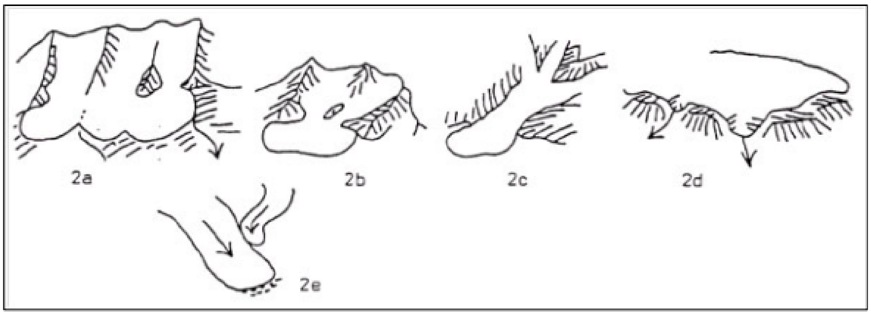


Figure 18: Glacier Frontal Characteristics
World Glaciers Inventory, Version 1
https : //nsidc.org/sites/default/files/g01130 − v001 − userguide_1_0.pdf

source_nourish - a 1-digit code that describes the source of nourishment for the glacier. Figure 20 lists the source nourishment codes.

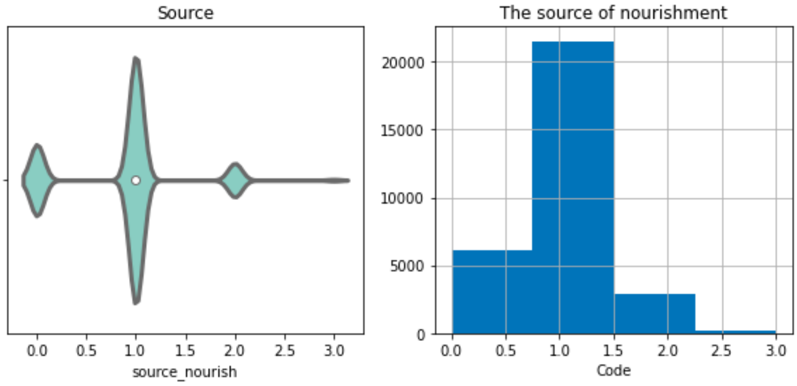
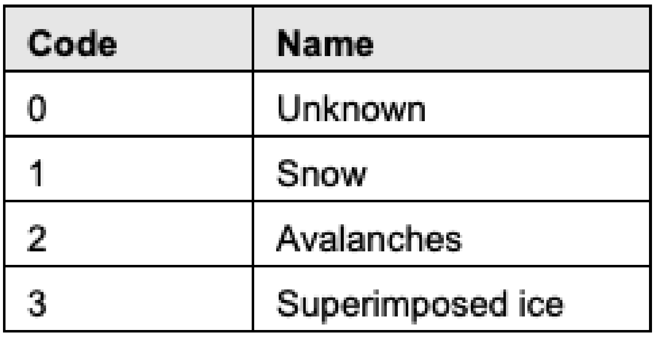


Figure 19: Source nourishment



Code	Name
0	Unknown
1	Snow
2	Avalanches
3	Superimposed ice

Figure 20: Source Codes
World Glaciers Inventory, Version 1
https : //nsidc.org/sites/default/files/g01130 − v001 − userguide_1_0.pdf

Machine learning model(s) [1 paragraph]

2.3 What type of model did you use? Why did you choose this model type?

Since I have a classification problem, I used three classifiers: DecisionTreeClassifier, RandomForestClassifier, and 9-KNeighborsClassifier. Also, I used DummyClassifier for a baseline with two strategies: 'stratified' and *'most_frequent'*. I chose these models because:
- K-Nearest Neighbors is a simple and effective algorithm that can be used for classification problems with multiple classes.
- Decision Tree Classifier can handle both binary and multi-class classification problems. Since I have multiple classes, this model is the best decision for me as it is easy to understand, and I can see how the essential features contribute to the classification.
- Random Forest Classifier combines multiple decision trees so that I can improve the accuracy and reduce overfitting. It can also handle both binary and multi-class classification problems, and I also have quite a big data set so this model can be a good choice.

2.4 What are some strengths and weaknesses of this model type?

Decision Tree Classifier is easy to interpret and explains the model's decision-making process, and it can also handle both categorical and numerical data, but small changes in the data set can lead to a different decision tree, and it is quite sensitive to a class imbalance in the data.
Random Forest Classifier can handle large and complex data sets, and it is less prone to overfitting compared to decision tree classifiers. On the other hand, it can be slow to train and make predictions on large data sets, and it is not as easy to interpret as decision tree classifiers.
K-Nearest Neighbors (KNN) Classifier has the same strengths as a Decision Tree Classifier, but it can be sensitive to the choice of neighbors and is sensitive to a class imbalance in the data.

2.5 What hyperparameters must be specified, what values did you choose, and how did you choose them, and/or what range of values did you explore for each hyperparameter?

In the Random Forest Classifier, I specified the number of trees in the forest, the maximum depth of each tree, the function to measure the quality of a split, and the random state.
RandomForestClassifier - $random\_state =' 12345'$, max_depth in the range $\{1, 25\}$, $n\_estimators$ in the range $\{1, 25\}$, $criterion =' gini'$.
I have chosen this range as I have determined that increasing the number of trees or depth beyond a certain point (25) doesn't improve the accuracy a lot but increases the computational cost significantly, so it makes sense to choose a lower value for these hyperparameters.

For the Decision Tree Classifier, I specified the maximum depth of the tree, the function to measure the quality of a split and the random state.
DecisionTreeClassifier - $random\_state =' 12345'$, max_depth in the range $\{1, 20\}$, $criterion =' gini'$.
I based on the trade-off between computational cost and overfitting when choosing a range for the DecisionTreeCLassifier. I have observed that the model begins to overfit significantly after 20 trees and the computational cost increases with more trees, so it makes sense to limit the number of trees to a lower value of 20.

In the 9-Nearest Neighbors (KNN) model, the main hyperparameter to specify is the number of neighbors to consider (k). I have tried different values for k, but with k=9 my model showed the best performance on thaining data set. KNeighborsClassifier - $n\_neighbors = 9$

For the selection method to choose the best value, I decided to choose a model with the best score after each challenge, and then use the pre-processed data set in further challenges. After getting the results each challenge and obtaining the score of the models, I tested it on the held out data set to get the final results for each challenge and to choose the best strategy and the best model for the second challenge.

Evaluation [1 paragraph]

2.6 What metrics did you use to measure performance?

For all challenges I used accuracy and $f1\_score$. As we can see from Figure 27, the data set is quite imbalanced, so I decided to use f1 score metric as it considers both precision and recall, which are important aspects that need to be balanced.

2.7 What experimental methodology did you employ to construct training, testing, and validation (if used) data sets?

For the training and validation data set I used items from 1900 to 1975 ('*train_test_split*' method), and I used a held-out data set from 1976 to 1996.

2.8 What baseline approach(es) did you compare your model to?

I have chosen to use DummyClassifieras a baseline approach. With the $'most\_frequent'$ strategy I can see the accuracy that a model can achieve by predicting the majority class. With the 'stratified' strategy, I can evaluate the performance of my classifiers compared to a random guessing strategy. I chose this approach because the Dummy Classifier is very easy to understand and implement, and it is useful for identifying class imbalance issues in the data .

Challenges [e.g., for each challenge, 1 paragraph per challenge to describe it (in your data set) and a paragraph for each of the three strategies]

2.9 Describe at least three challenges you've found with this data set (outliers, missing values, range/scale/units for features, sampling bias, correlations that make the data not i.i.d., etc.) and/or the application area (deployment, maintenance, etc.). Explain how the challenge manifests in your data set (e.g., for missing values, what fraction of values are missing and for which features? For class imbalance, what is the class distribution?)]

2.9.1 First challenge - missing values

The problem of missing values in the data set refers to instances without data for a particular measurement or observation. This can occur for various reasons, such as instrument failure, data not being recorded, or data loss or damage. This missing data can significantly impact the accuracy and reliability of analysis and modeling performed on the data set.

The number of missing values for each feature is below:

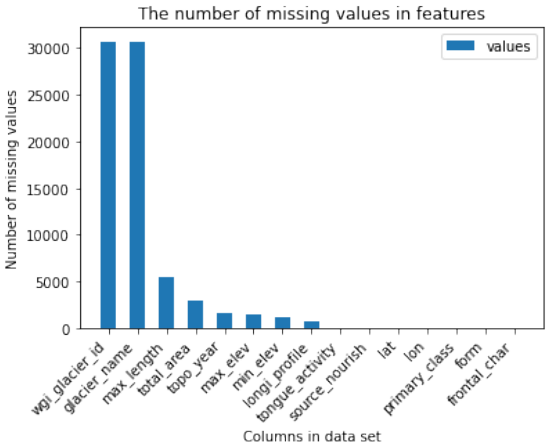


Figure 21: Missing values in the data set

Since the features, *wgi_glacier_id* and *glacier_name* contain the id and the name of the glaciers, respectively. Both features contain an essential number of missing values of the data set; we can abstain from these features as they don't convey any vital information for the training. So that we can show the missing values in the features below in Figure 22:

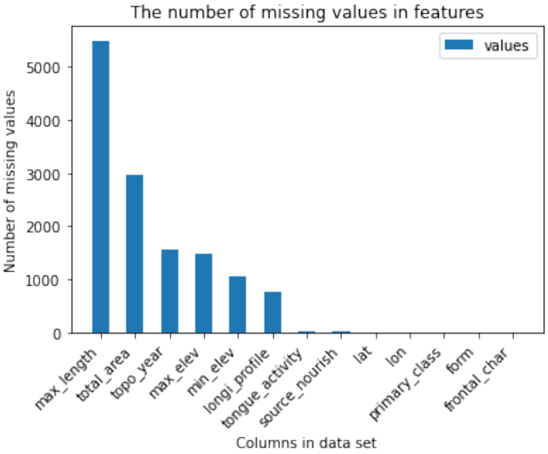


Figure 22: Missing values in the data set

Let's look at the plot of missing values:

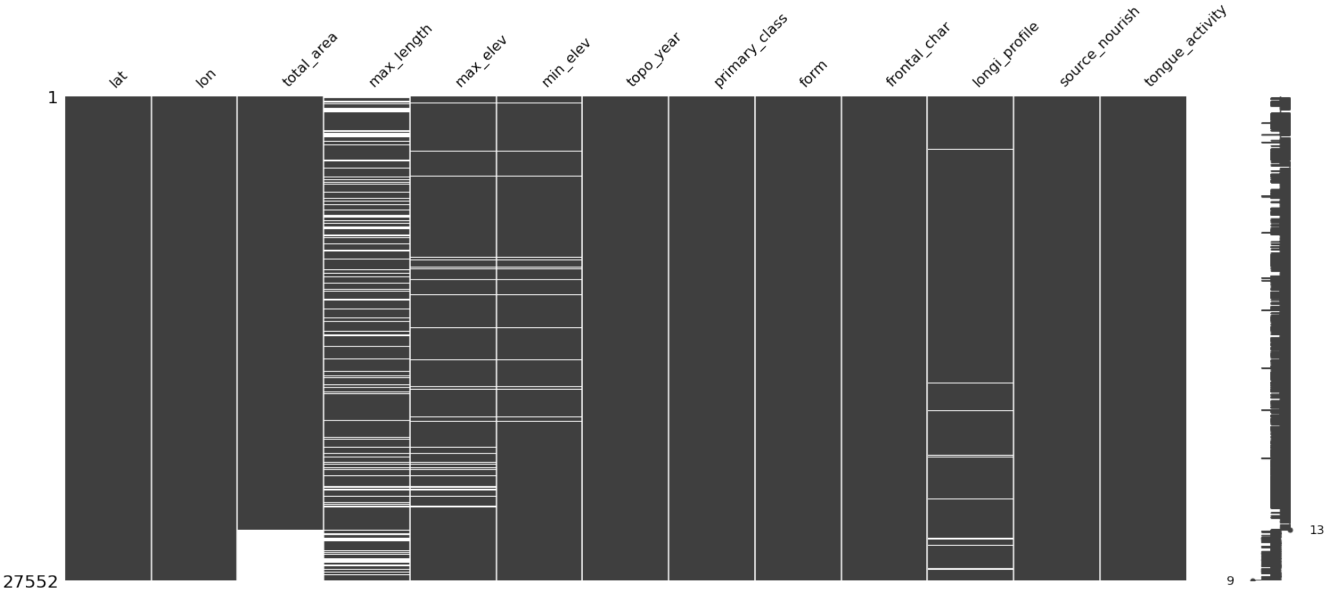


Figure 23: Missing values in the data set

I have sorted the data set on the column of *total_area*.

The *longi_profile* has very few missing values and does not seem to be correlated with any other column,hence, the missingness in column can be attributed as Missing Completely at Random. The *max_elev* and *min_elev* at the same time have a lot of missing values, and this could be a case of MAR as we cannot directly observe the reason for missingness of data in these columns. Let's look at the heatmap:

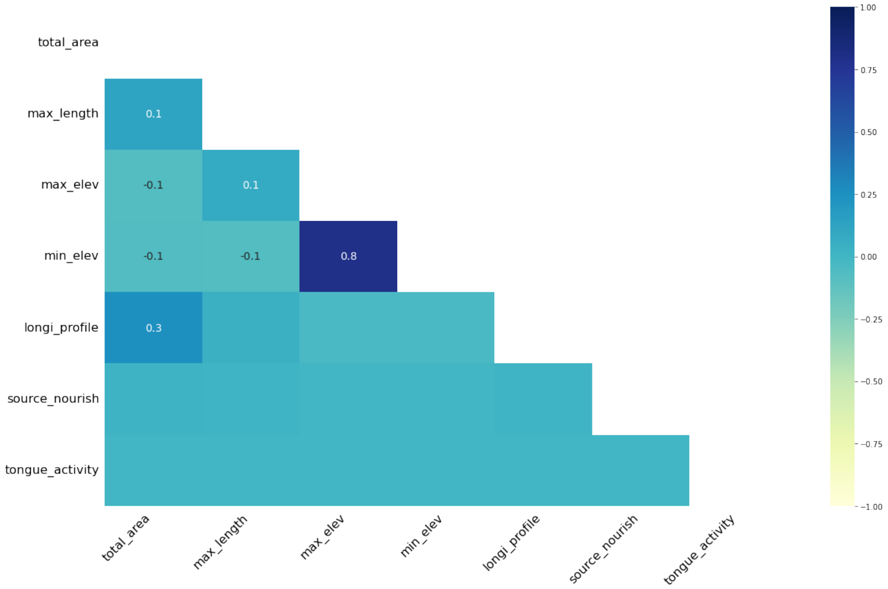


Figure 24: Heatmap of the missing values in the data set

The heatmap function shows that there is a pretty strong correlations between missing values of *max_elev* and *min_elev* features. It may indicate that the missingness may be related to the values of the missing data, so we can assume Missing Not at Random (MNAR) case. However, the correlation alone cannot determine whether the data are MAR or MNAR, so I'll try to understand what's going on in the data set while dealing with missing values to train the model.

If a test set has missing values: I will build a transformation pipeline that can handle all the necessary preprocessing steps on training set and apply this pipeline transformations on the test data. If I use test mean for that, I can get an information leak because calculating mean of test data set would give me algorithm information about mean of it and would probably falsely improve its score on said.

2.9.2 Second challenge - outliers

In my project, I am focusing on outlier values within a feature. Outliers in the features may refer to data points that deviate significantly from the expected or typical range of values for a particular variable or set of variables. For example, a glacier with a very low or very high surface area may be considered an outlier relative to other glaciers in the data set. Outliers can skew the results and lead to inaccurate predictions. I can observe the negative impact of outliers for the features '*total_area*' and '*max_length*': the violin plots are hard to interpret (to detect the median, mean values, etc.) because of the impact of outliers, and the histograms are also positively skewed, so that implies the presence of outliers in the data set.

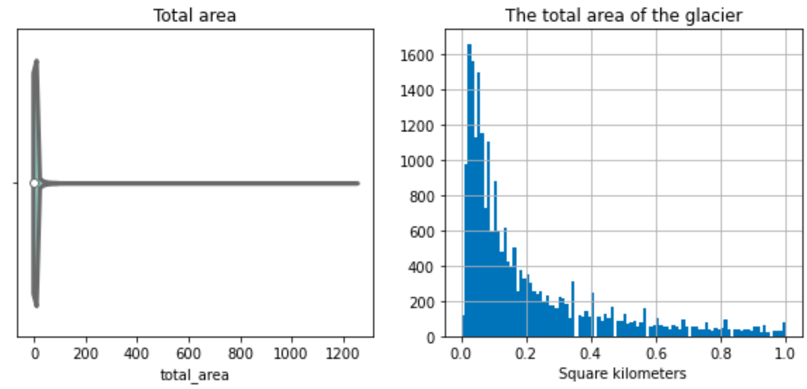


Figure 25: Total area

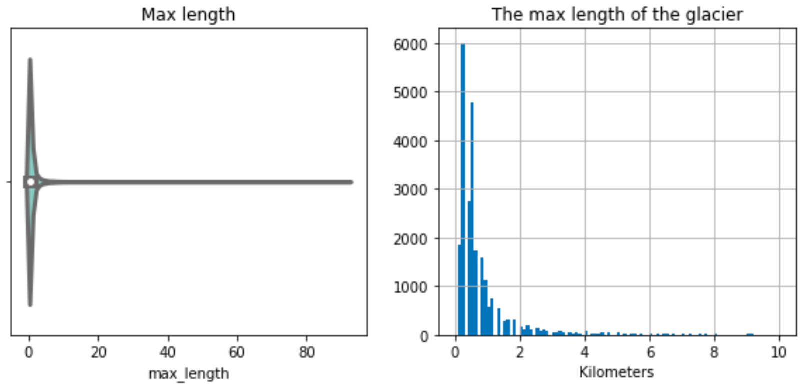


Figure 26: Max length

2.9.3 Third challenge - class imbalance

Also, I found the presence of a class imbalance in my data set: it refers to the situation when the distribution of classes is not balanced, and some classes have significantly fewer instances compared to others. Class imbalance can affect the performance of the models as they tend to perform well in the majority class but poorly in the minority class, leading to biased predictions.

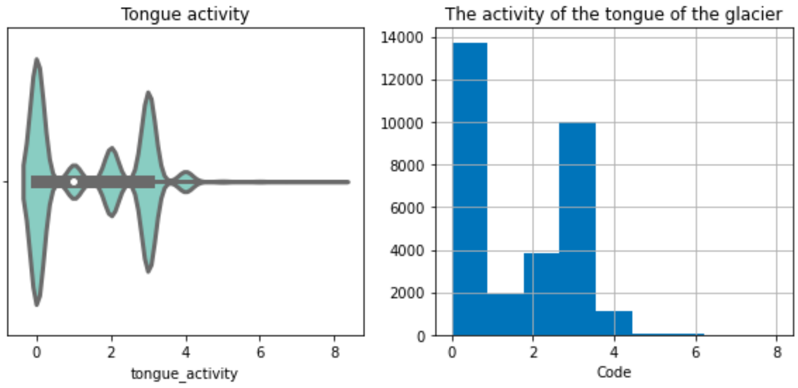


Figure 27: Tongue activity distribution

Here we can see that the classes of the *tongue_activity* from 0 to 4 are more or less well represented, but I can't tell the same for classes from 5 to 8.

2.10 For each challenge, describe three alternative strategies that you investigated. Explain in enough detail that someone else could use this strategy.

2.10.1 1.1 Delete missing items

Deleting items with missing values involves removing any data points containing one or more missing values.

To make a prediction

Advantages:
- simple to implement;
- effective in cases where missing values are limited in number;
- useful when missing values are thought to be completely at random (MCAR).

Disadvantages:
- a loss of information;
- can lead to a biased representation of the data if missing values are not completely at random (MCAR).

2.10.2 1.2 Impute missing values with average

Imputing missing values with the average strategy involves replacing the missing values in the data set with the mean value of that feature.

Advantages:
- simple and easy to implement;
- preserves the overall distribution of the variable.

Disadvantages
- can lead to an underestimation of the variance;
- may introduce bias if the proportion of missing values is large.

2.10.3 1.3 Linear Regression Imputation

Linear Regression imputation strategy involves predicting the missing values based on the linear relationship between the missing feature and other features in the data set. I am going to train a linear regression model on the observed values of the feature and then use the trained model to predict the missing values of the feature.

Advantages:
- the imputed values are more flexible than fixed values like the mean or median;
- the imputed values can be more accurate as LRImputation takes into account the correlation between the feature and other features in the data set.

Disadvantages:
- it assumes a linear relationship between the feature and other features, which may not always be true;
- can be computationally expensive and time-consuming, especially for large data sets.

2.10.4 2.1 Doing nothing

It is better not to delete outliers in some cases because outliers may contain valuable information and removing them can result in a loss of information or bias in the data. Outliers may indicate rare but important occurrences, extreme values, or errors in data collection or recording. They can also reveal patterns or relationships in the data that would not be evident without them.

2.10.5 2.2 Interquartile Range

Interquartile Range (IQR) involves calculating the difference between the first and third quartiles of the dataset (the 25th and 75th percentiles), which gives the range of values that are considered "normal". So, I'll define this range and leave only values that are within 25 and 75 quartiles for each feature with outliers separately.

Advantages:
- less sensitive to extreme values than other methods such as z-score or mean-sd methods;
- simple and easy to understand.

Disadvantages:
- it may identify too many or too few outliers;
- may not work well with small sample sizes or highly skewed distributions.

2.10.6 2.3 Square Root Transformation

The square root transformation will help me to deal with positive skewness in data. So that, I'll calculate the square root of each value in the feature with outliers, and then check the distribution of the transformed feature for normality.

Advantages:
- simple and easy to implement;
- can be effective at reducing the influence of extreme values on the overall data set.

Disadvantages:
- may not be effective if the extreme values are particularly extreme or if there are many outliers in the data set;
- may result in a loss of information as the original values are transformed.

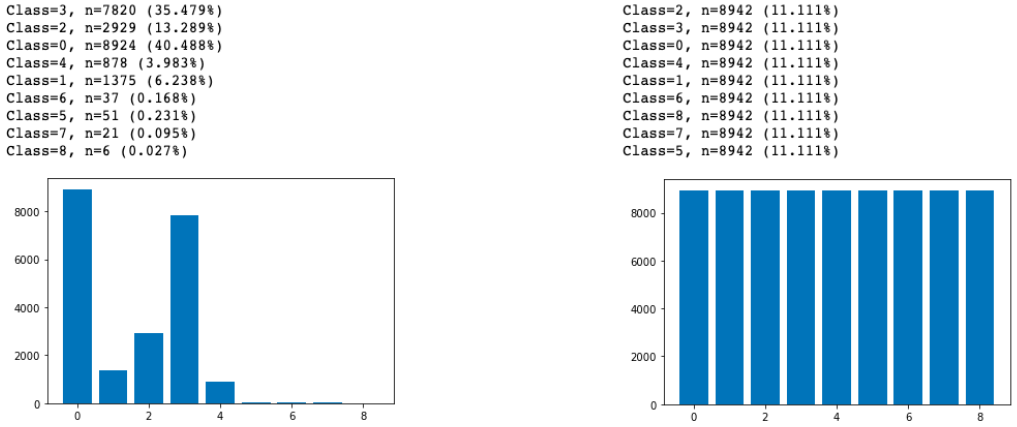
2.10.7 3.1 Balance = 'weighted'

I set the *class_weight* parameter to 'balanced' while training the model to implement a strategy for handling class imbalance. When 'balanced' is used, the algorithm gives more weight to under-represented classes and less weight to over-represented classes.
Advantages:
- a quick and easy way to address class imbalance without requiring much additional effort or knowledge;
- can work well when the classes are well-separated and the decision boundary is relatively clear.
Disadvantages:
- may not work well if the classes are highly overlapping;
- may not work well for data sets with very small minority classes, as the weights may become too extreme and lead to overfitting.

2.10.8 3.1 Upsampling

For upsampling, I used SMOTE (Synthetic Minority Over-sampling Technique)
https : //imbalanced−learn.org/stable/references/generated/imblearn.over_sampling.SMOTE.html

It generates synthetic samples for the minority class by creating new examples that are similar to existing minority class samples. So, I had to fit the SMOTE instance to the training data. I applied SMOTE to all classes except 0 as it represents the major part of the items in the data set.



```
Class=3, n=7820 (35.479%)
Class=2, n=2929 (13.289%)
Class=0, n=8924 (40.488%)
Class=4, n=878 (3.983%)
Class=1, n=1375 (6.238%)
Class=6, n=37 (0.168%)
Class=5, n=51 (0.231%)
Class=7, n=21 (0.095%)
Class=8, n=6 (0.027%)
```

```
Class=2, n=8942 (11.111%)
Class=3, n=8942 (11.111%)
Class=0, n=8942 (11.111%)
Class=4, n=8942 (11.111%)
Class=1, n=8942 (11.111%)
Class=6, n=8942 (11.111%)
Class=8, n=8942 (11.111%)
Class=7, n=8942 (11.111%)
Class=5, n=8942 (11.111%)
```

(a) Before SMOTE

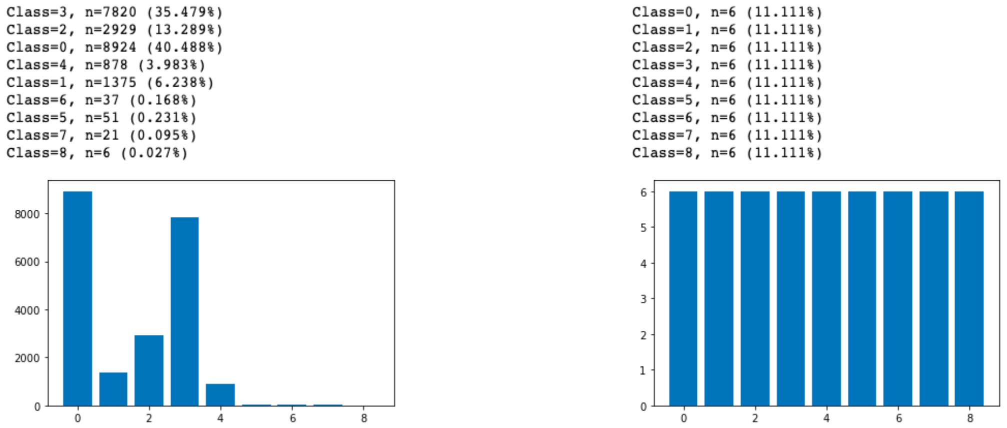
(b) After SMOTE

Figure 28: Upsampling with SMOTE

So after applying the SMOTE and generating synthetic samples, each class would have the same number of items (8942). After that, I trained the classifier on the resampled training data and evaluated the performance on the original validation data.
Advantages:
- generation of synthetic examples that are representative of the minority class, rather than simply duplicating existing examples.
Disadvantages:
- may generate synthetic examples that are too similar to existing examples, leading to overfitting;
- may not be effective for highly imbalanced datasets where the minority class is very small.

2.10.9 3.1 Downsampling

For downsampling, I used the NearMiss strategy by removing examples from the majority class and reducing the number of the items in each class equal to the amount of items in the minor class (class 8 - 6 items).



```
Class=3, n=7820 (35.479%)
Class=2, n=2929 (13.289%)
Class=0, n=8924 (40.488%)
Class=4, n=878 (3.983%)
Class=1, n=1375 (6.238%)
Class=6, n=37 (0.168%)
Class=5, n=51 (0.231%)
Class=7, n=21 (0.095%)
Class=8, n=6 (0.027%)
```

```
Class=0, n=6 (11.111%)
Class=1, n=6 (11.111%)
Class=2, n=6 (11.111%)
Class=3, n=6 (11.111%)
Class=4, n=6 (11.111%)
Class=5, n=6 (11.111%)
Class=6, n=6 (11.111%)
Class=7, n=6 (11.111%)
Class=8, n=6 (11.111%)
```

(a) Before NearMiss

(b) After NearMiss

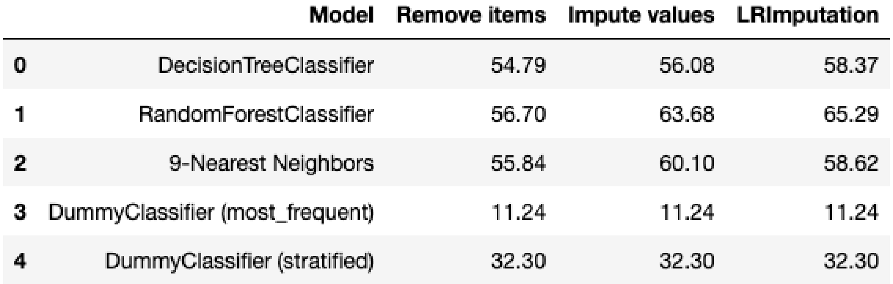
Figure 29: Downsampling with NearMiss

So, I imported the NearMiss class from the *imblearn.under_sampling* module fitted the NearMiss object to the training data and target labels using the *fit_resample* method, and then used the resampled data for training my classification models.
Advantages:
- can be effective at reducing the class imbalance, especially when the majority class has a much higher frequency than the minority class;
- a relatively simple and easy-to-implement approach.
Disadvantages:
- can result in information loss, as it discards some of the majority class samples;
- may not be effective if the majority class has significant intra-class variability.

Results. For each challenge: (use placeholders for any in-progress results and explicitly mention that they are in progress)

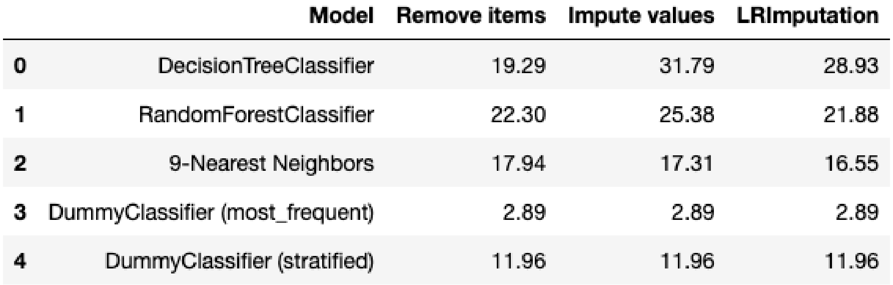
2.11 Show your experimental results (probably a table comparing the strategies for this challenge with one or more metrics)

2.11.1 First challenge - Missing values



	Model	Remove items	Impute values	LRImputation
0	DecisionTreeClassifier	54.79	56.08	58.37
1	RandomForestClassifier	56.70	63.68	65.29
2	9-Nearest Neighbors	55.84	60.10	58.62
3	DummyClassifier (most_frequent)	11.24	11.24	11.24
4	DummyClassifier (stratified)	32.30	32.30	32.30

Figure 30: Accuracies for the first challenge



	Model	Remove items	Impute values	LRImputation
0	DecisionTreeClassifier	19.29	31.79	28.93
1	RandomForestClassifier	22.30	25.38	21.88
2	9-Nearest Neighbors	17.94	17.31	16.55
3	DummyClassifier (most_frequent)	2.89	2.89	2.89
4	DummyClassifier (stratified)	11.96	11.96	11.96

Figure 31: F1-scores for the first challenge

According to the obtained results, we can see that all models beat the baseline.
Accuracies:
- Remove items: RandomForestClassifier with the highest accuracy 56.70%
- Impute values: RandomForestClassifier with the highest accuracy 63.68%
- LRImputation: RandomForestClassifier with the highest accuracy 65.29%

$F1\_scores$:
- Remove items: RandomForestClassifier with the highest score 22.30%
- Impute values: DecisionTreeClassifier with the highest score 31.79%
- LRImputation: DecisionTreeClassifier with the highest score 28.93%

2.11.2 Second challenge - Outliers

For the second idea in the second challenge, I applied IQR method. Let's look how features changed:
Total area before:

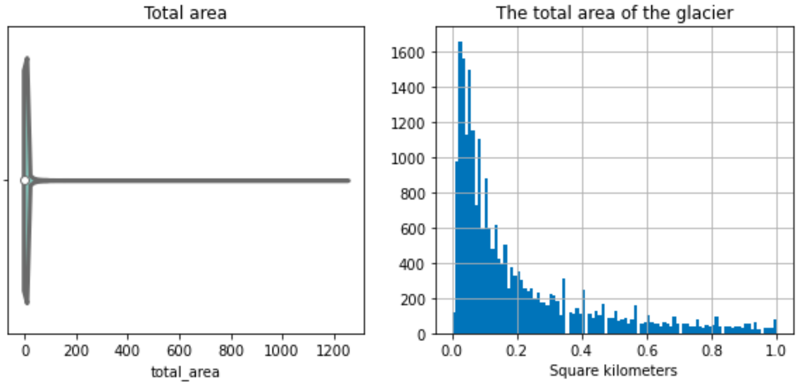


Figure 32: Total area before

Total area after:

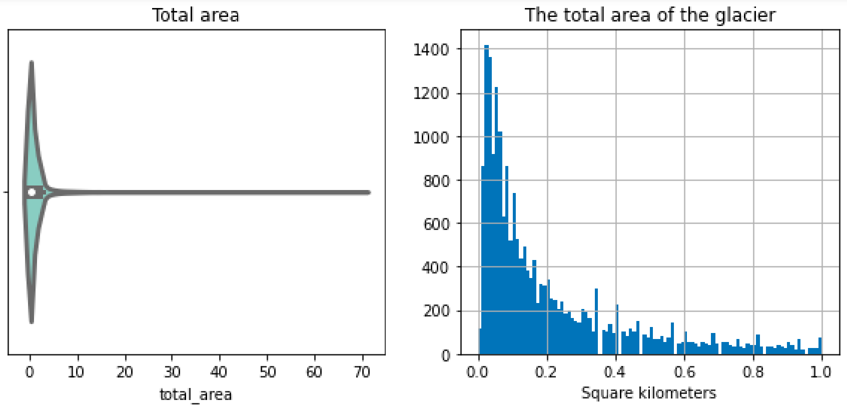


Figure 33: Total area after

Table 3: Total Area

Table 4: Before

count	27552
count	27552
mean	2.505
std	22.565
min	0.001
25%	0.062
50%	0.220
75%	1.220
max	1250

Table 5: After

count	27395
mean	1.292
std	4.380
min	0.001
25%	0.060
50%	0.210
75%	1.190
max	70.130

As we can see, standard deviation decreased almost 6 times, and the mean value became twice closer to the median. Maximum value decreased from 1250 to 70.130 square km.

Max length before:

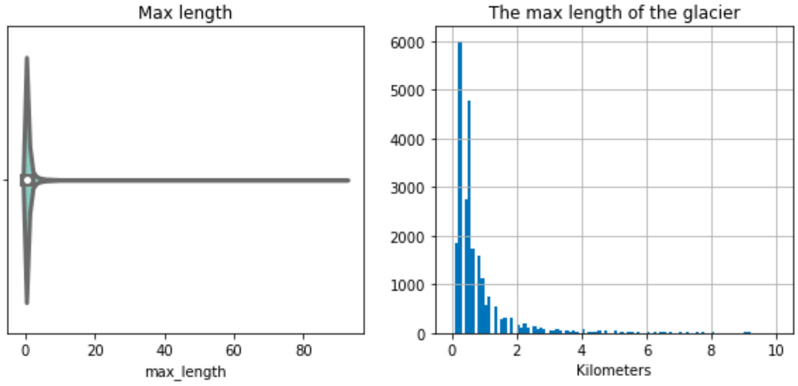


Figure 34: Max length before

Max length after:

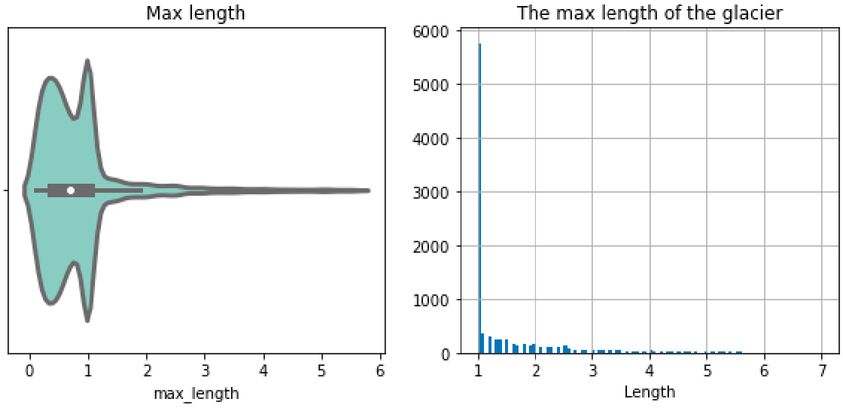


Figure 35: Max length after

Table 6: Max length

Table 7: Before

count	27395
mean	0.974
std	1.543
min	0.001
25%	0.400
50%	0.700
75%	1.000
max	92.000

Table 8: After

count	26874
mean	0.810
std	0.743
min	0.100
25%	0.400
50%	0.700
75%	1.000
max	5.600

For the max length, standard deviation decreased twice. Maximum value decreased from 92 to 5.6 km.
Max elevation before:

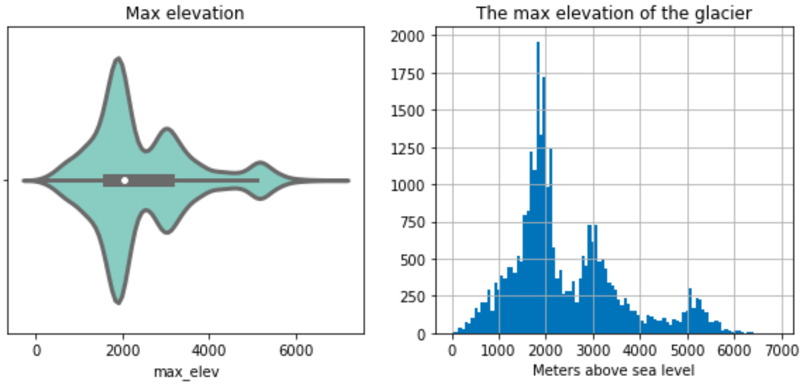


Figure 36: Max elevation before

Max elevation after:

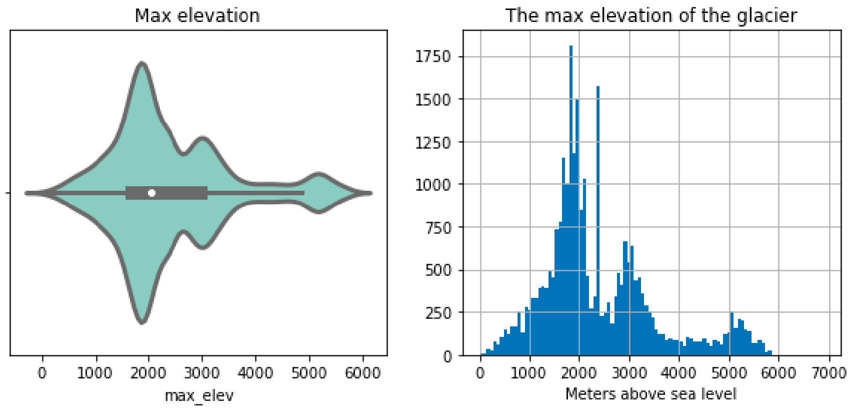


Figure 37: Max elevation after

Table 9: Max elevation

Table 10: Before

count	26874
mean	2392.749
std	1153.417
min	0.000
25%	1690.000
50%	2042.000
75%	2970.000
max	6900.000

Table 11: After

count	26770
mean	2378.118
std	1131.367
min	0.000
25%	1690.000
50%	2042.000
75%	2960.000
max	5850.000

With max elevation, standard deviation slightly decreased. Maximum value decreased from 6900 to 5850 m. The histogram looks like three different normal distributions are overlapping.

Min elevation before:

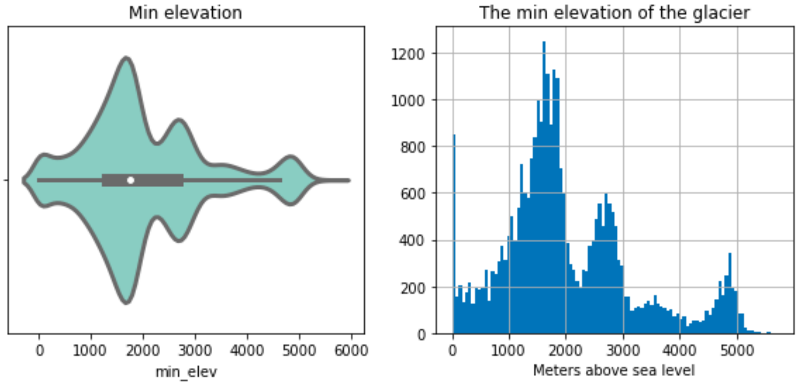


Figure 38: Min elevation before

Min elevation after:

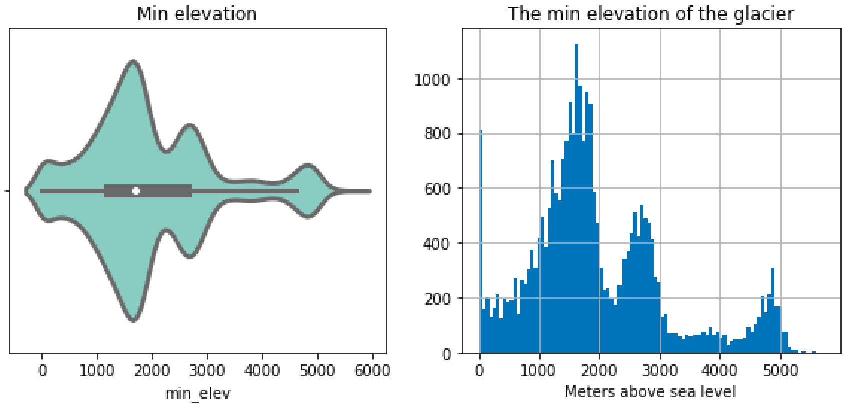


Figure 39: Min elevation after

Table 12: Max elevation

Table 13: Before

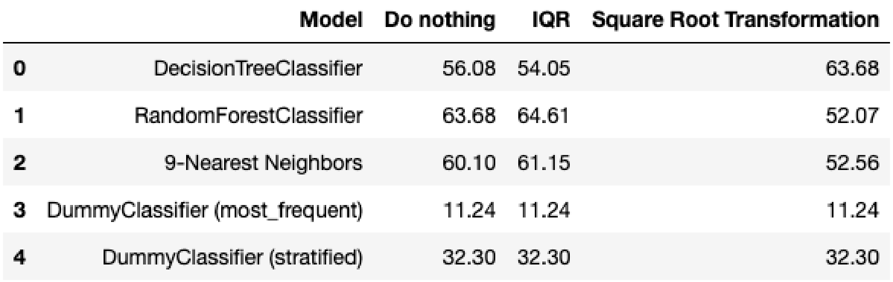
count	26770
mean	1981.271
std	1112.120
min	0.000
25%	1310.000
50%	1765.000
75%	2570.000
max	5645.000

Table 14: After

count	26759
mean	1979.832
std	1110.081
min	0.000
25%	1310.000
50%	1765.000
75%	2570.000
max	5315.000

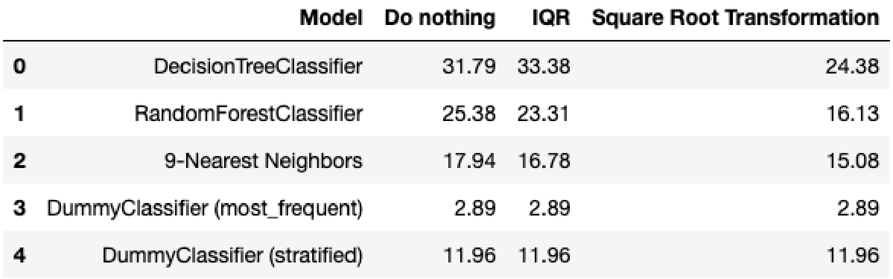
With min elevation, the maximum decreased from 5645 to 5315 m. We can also observe peaks of the values close to 0 - probably, there are many glaciers on the low elevation, so there is no reason to classify these values as weird and to delete them as they can convey significant information.

Let's look at the model performance:



	Model	Do nothing	IQR	Square Root Transformation
0	DecisionTreeClassifier	56.08	54.05	63.68
1	RandomForestClassifier	63.68	64.61	52.07
2	9-Nearest Neighbors	60.10	61.15	52.56
3	DummyClassifier (most_frequent)	11.24	11.24	11.24
4	DummyClassifier (stratified)	32.30	32.30	32.30

Figure 40: Accuracies for the second challenge



	Model	Do nothing	IQR	Square Root Transformation
0	DecisionTreeClassifier	31.79	33.38	24.38
1	RandomForestClassifier	25.38	23.31	16.13
2	9-Nearest Neighbors	17.94	16.78	15.08
3	DummyClassifier (most_frequent)	2.89	2.89	2.89
4	DummyClassifier (stratified)	11.96	11.96	11.96

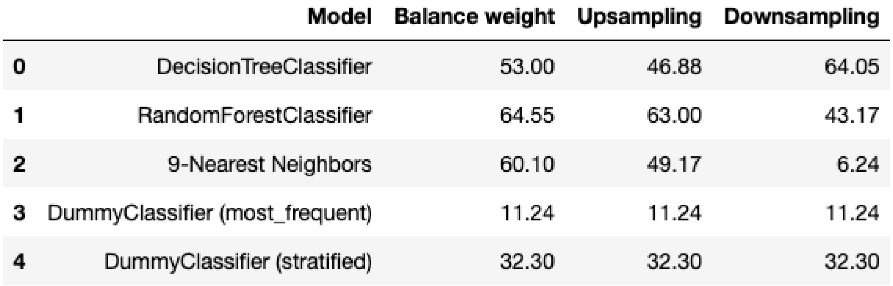
Figure 41: F1-scores for the second challenge

Accuracies:
- Do nothing: RandomForestClassifier with the highest accuracy 63.68%
- IQR: RandomForestClassifier with the highest accuracy 64.61%
- Square Root Transformation: DecisionTreeClassifier with the highest accuracy 63.68%

$F1\_scores$:
- Do nothing: DecisionTreeClassifier with the highest score 31.79%
- IQR: DecisionTreeClassifier with the highest score 33.38%
- Square Root Transformation: DecisionTreeClassifier with the highest score 24.38%

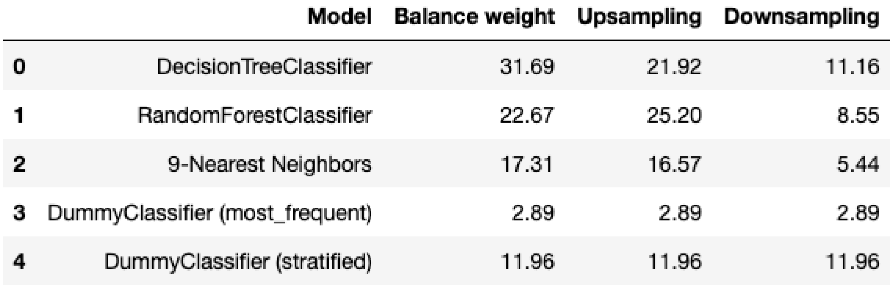
2.11.3 Third challenge - Class Imbalance

Let's look at the model performance:



	Model	Balance weight	Upsampling	Downsampling
0	DecisionTreeClassifier	53.00	46.88	64.05
1	RandomForestClassifier	64.55	63.00	43.17
2	9-Nearest Neighbors	60.10	49.17	6.24
3	DummyClassifier (most_frequent)	11.24	11.24	11.24
4	DummyClassifier (stratified)	32.30	32.30	32.30

Figure 42: Accuracies for the second challenge



	Model	Balance weight	Upsampling	Downsampling
0	DecisionTreeClassifier	31.69	21.92	11.16
1	RandomForestClassifier	22.67	25.20	8.55
2	9-Nearest Neighbors	17.31	16.57	5.44
3	DummyClassifier (most_frequent)	2.89	2.89	2.89
4	DummyClassifier (stratified)	11.96	11.96	11.96

Figure 43: F1-scores for the second challenge

Accuracies:
- Balance weight: RandomForestClassifier with the highest accuracy 64.55%
- Upsampling: RandomForestClassifier with the highest accuracy 63.00%
- Downsampling: RandomForestClassifier with the highest accuracy 64.05%

$F1\_scores$:
- Balance weight: DecisionTreeClassifier with the highest score 31.69%
- Upsampling: RandomForestClassifier with the highest score 25.20%
- Downsampling: DecisionTreeClassifier with the highest score 11.16%

2.12 Discuss which strategy(ies) were most effective for this challenge [1 paragraph]

All models were trained on the training data set and tested on the held-out data set. After obtaining the model with the best performance for one strategy, I used the results for the next.
Challenge 1: Overall, the best results were shown for the Impute average strategy, and all models beat the baseline score.
Challenge 2: After the first challenge, the results of the Impute Values strategy were used to work with the next challenge. After preprocessing the outliers, the best strategy was to do nothing, as the model performance is better while the performance of others varies. Also, I continue to use the data set with processed missing values with Impute average strategy. The data set with the 'doing nothing' strategy was used in the next challenge.
Challenge 3: Overall, I would say that the best strategy is to set up a parameter $'class\_weight'$ = balanced to get the highest accuracy and f1-score. RandomForest performs excellently, while KNN's accuracy can't beat any baseline.

2.13 Reflection

2.13.1 What surprised you about your results? [1 paragraph]

I was shocked when I saw that DecisionTreeClassifier showed the best performance for the class imbalance strategy with downsampling, as there were few items left after preprocessing the data. Also, it was tough sometimes to see that all my plans didn't improve the accuracy but decreased it, and the best strategy to apply was 'to do nothing .'I always say that a bad experience is an experience too, so now I can also say that zero results are also a result.

2.13.2 Show your model's performance results to someone in the class and ask them to identify one aspect of the results that seems surprising, or they would like to understand better. Describe their feedback. [1 paragraph]

I showed my results to a couple of classmates - they were surprised to see the relatively low F1 score for the Downsampling strategy in the third challenge. The RandomForestClassifier had the highest accuracy for this strategy, but the DecisionTreeClassifier had a very low F1-score of 11.16%. They suggested that while the model could identify many negative cases correctly, it struggled to identify positive issues. It would be interesting for me to investigate this further and explore other resampling techniques to see if they can improve the F1 score for this strategy.

2.13.3 What did you have to (or choose to) change (from your original plans)? [1 paragraph]

First, I had the features in the almost empty data set, so I needed clarification about which strategy would better handle this. But then I talked to the Ocean and Geography department representative at OSU, and they advised me to use other features that could be more useful and don't have so many missing values. Also, I wanted to use the AUC-ROC metric with strategy OvR, but I found a mistake in my code (I used incorrect features and made a mistake with information leakage). After seeing that, I used F1-score and accuracy metrics for all challenges.

2.13.4 If you had more time, what additional investigation would you do with this data set? (What are your unanswered questions?) [1 paragraph]

I want to try feature engineering because I found very highly correlated features, so for me, it's another big challenge to try different techniques and find the reason, or at least to improve the performance of my models. Also, I'd like to try out different machine learning algorithms, such as Support Vector Machines or Neural Networks, to see if they can provide better results compared to the models used in this project.

2.14 Extra credit User Feedback: Connect with someone who is or could be a user of your system (predictions made by your classifier on the data set you chose). In your final report, include a paragraph describing the user's feedback on how you addressed the challenges in the data. Ask the user to answer these questions:

2.14.1 Were the challenges addressed adequately?

I spoke with a Ph.D. student from the Ocean and Geography department; they told me the challenges I addressed in my work had been appropriately handled. They observed that I used different strategies to address missing values, outliers, and class imbalance and tested multiple machine-learning models to evaluate their performance.

The Ph.D. student was curious about why I chose a particular target to evaluate the melting of the glaciers. They suggested that I investigate the parameters that indicate whether a glacier has a clean snow surface or is debris-covered. They are working on distinguishing these types of glaciers using data from various bands. They also noted that debris cover could impact the speed of glacier melting. It is exciting that machine learning can detect the relationship between debris cover and tongue activity, which may not be evident to people.

2.14.2 What remaining questions/challenges exist that might prevent this classifier from being used?

They told me that the dataset I have used for training and testing the classifier might only be representative of some glaciers in the world, and different regions and climate conditions may require other models and strategies. Also, they mentioned that even though the Inventory I used is quite huge, it has many missing values and needs to be more representative (I had the same thought while working on my project). So, they advised me to tackle this topic using Computer Vision and Google Earth Engine, where I can obtain the data set of glacier images, and it's updated more often than World Glaciers Inventory.

References

scikit-learn 1.2.0
$https://towardsdatascience.com/multiclass - classification - evaluation - with - roc - curves - and - roc - auc - 294fd4617e3a$