

Article

Detection and Analysis of Molten Aluminium Cleanliness Using a Pulsed Ultrasound System

Kgothatso Matlala^{1,†,‡} , Amit Kumar Mishra^{1,†,‡}  and Deepak Puthal^{2,†} 

¹ Affiliation 1; Electrical Engineering Department, University of Cape Town

² Affiliation 2; Khalifa University

* Correspondence: kgothatso@excite-data.tech, akumishra@ieee.org; Tel.: +27-21-460-9333

† Observatory Road, Observatory, Cape Town, South Africa

‡ These authors contributed equally to this work.

Abstract: The analysis of data produced by the MV20/20 sensor, tagged with quality outcomes, is presented with the aim of developing a predictor model for real-time anomaly detection and classification. Three types of inclusions, undesired particles that deteriorate the quality of production, are used to tag the quality data using results from the lab. We explore both unsupervised and supervised learning, which both offer advantages in monitoring and controlling the quality of production. It is discovered that the dataset can be clustered using techniques like K-Means and DBSCAN. Bounding the data within a 95% confidence interval ellipse ensures we can detect anomalous events in real time. For supervised learning, a two-stage classifier is explored, which classifies the outcome of a cast and secondly the inclusion responsible for the negative outcome. We explore models from logistic regression and support vector machines, to two neural networks, namely the multi-layer perceptron and the radial basis function network. While the cast outcome is adequately predicted by all the models, the multi-layer perceptron provides a boundary performance for the inclusion type. A more advanced technique for model optimisation, namely grid search, is applied in order to improve on the results. The outcome for the grid search is not much better, which indicates a global maximum in the learning capacity of the model. Recommendations include the addition of sensor systems and an audit of data collection variation.

Keywords: MV20/20; PoDFA; anomaly detection; statistical process control; principal components analysis; K-Means; DBSCAN; multi-layer perceptron; inclusion; receiver operating characteristic; confusion matrix.



Citation: Matlala, K.; Mishra, A.K. Puthal, D. Detection and Analysis of Molten Aluminium Cleanliness Using a Pulsed Ultrasound System. *Preprints* **2022**, *1*, 1. <https://doi.org/>

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Background on Aluminium Casting

A typical casthouse consists of a sequence of machine centers that perform dedicated tasks on the Aluminium. Raw, recycled metal is fed into a melting furnace, where it is molten and initial cleaning takes place after large impurities are scraped from the surface of the cast. The cast is then transferred to a holding furnace, where it is held further to allow heavy inclusions to sink to the bottom while the lighter ones rise to the surface. The surface inclusions are scraped off. The metal is then flown through a launder, where a filter, degasser and metal rod are placed to trap smaller inclusions and other impurities [1,2]. The metal is finally cast into several billets ready for downstream processing. Each billet typically weighs over 10 tons.

1.2. Background on Casthouse Measurement Systems

To date, several prevalent analytical techniques exist, that are used to characterise metal quality during production.

The PoDFA (porous disk filtration apparatus) is a technique for collecting inclusions inside a fine porosity filter disk. The molten Aluminium is extracted from the cast and poured into a heated crucible. Once cooled, the sample is placed under a microscope for metallographic analysis. The PoDFA technique has its strength in its ability to accurately identify inclusions [3,4].

The LiMCA method provides electrical measurements, in which samples are measured every minute. The samples are based on the electrical resistivity of the metal sample, which is directly related to the metal cleanliness [5,6].

The MV20/20 system provides more real-time measurements by measuring 10 samples per second. This is achieved by the usage of ultrasound, where a pulse is transmitted in the metal and the return signal is measured. The MV20/20 measures cleanliness, particle size distributions and a count of inclusions [7,8]. This dataset provides a basis for our study, as it allows for a more comprehensive analysis of metal quality. The objective of this study is to implement an autonomous quality control system which realises real-time measurements, alerts on anomalies and classifies the inclusion types responsible for the deviation in quality. Anomaly detection is seen to be achievable through the use of clustering methods like K-Means and DBSCAN, while univariate statistical process control keeps track of individual variables. For classification, multilayer perceptrons have shown the best performance.

The remainder of this document contains the methodology applied, the experiments performed, the results and recommendations for future work.

2. Methodology

2.1. Data Exploration

The data exploration involves ingestion, standardisation, visualization and statistical analysis of the data in order to gain insight into the nature of the dataset. Once data is ingested, it is wrangled, which involves checking for missing and inconsistent values. Finally, plots are generated to visualize the behavioral patterns of the dataset.

2.2. Univariate Statistical Process Control

Univariate statistical process control (SPC) is an industrial framework for statistically determining the control limits for target parameters. The charts implemented in this study include individual, run and moving range.

2.3. Multivariate Clustering

Two techniques are considered for multivariate clustering, which include K-Means and DBSCAN. The K-Means method uses principal components analysis and clusters using the Hartigan-Wong, Lloyd and MacQueen algorithms respectively. The DBSCAN algorithm is based on varying the values of ϵ to achieve an optimal configuration of clusters.

2.4. Classification

The classification involves using the sample result and inclusion type variables as target respondents respectively. For both of them, four algorithms are compared, namely logistic regression, support vector machines, multilayer perceptron and the radial basis function network.

3. Experiments

3.1. Data Exploration

A summary of the input data is shown in the following table:

field	type	count	distinct_count	min	mean	max	stddev	range
Cleanliness	uint	504	47	50	55.17	62	2.22	44
Filtered_Mass	float	504	43	1.00	1.15	1.31	0.09	0.31
Inclusion_Count	uint	504	97	1	24.58	73	19.67	72
Inclusion_Type	factor	504	3					
LPS_120_140 __m	uint	504	20	0	0.88	19	2.40	19
LPS_140_160 __m	uint	504	11	0	0.19	3	0.96	13
LPS_20_30 __m	uint	504	22	0	8.87	17	3.19	15
LPS_30_40 __m	uint	504	51	0	17	50	10.65	39
LPS_40_50 __m	uint	504	42	0	4.64	22	6.22	31
LPS_50_60 __m	uint	504	64	0	1.81	16	3.92	49
LPS_60_90 __m	uint	504	22	0	1.13	13	3.63	15
LPS_90_120 __m	uint	504	47	0	0.56	18	5.55	39
LPS_160 __m	uint	504	1	0	0	0	0	0
MV_Grade	uint	504	56	49	59.49	72	4.26	46
Mean_LPS __m	uint	504	95	27	48.55	111	21.00	84
No Signal	uint	504	111	10	56.12	96	23.41	86
PSP1000M	uint	504	111	4	43.88	90	23.41	86
Peak LPS __m	uint	504	87	32	73.23	152	38.22	120
Sample_Result	factor	504	2					

Table 1. Input data summary

The dataset has 19 features. Of the features, 17 are numeric and 2 are categorical (inclusion type and sample result). One feature, namely LPS_160__m, is constant and is therefore discarded from the dataset. In addition, the features Mean_LPS __m and Peak LPS __m are derived features which are calculated and not directly measured from the system. They are also therefore discarded from the dataset.

The following scatterplot shows the correlations between the features, coloured by the sample result categorical respondent:

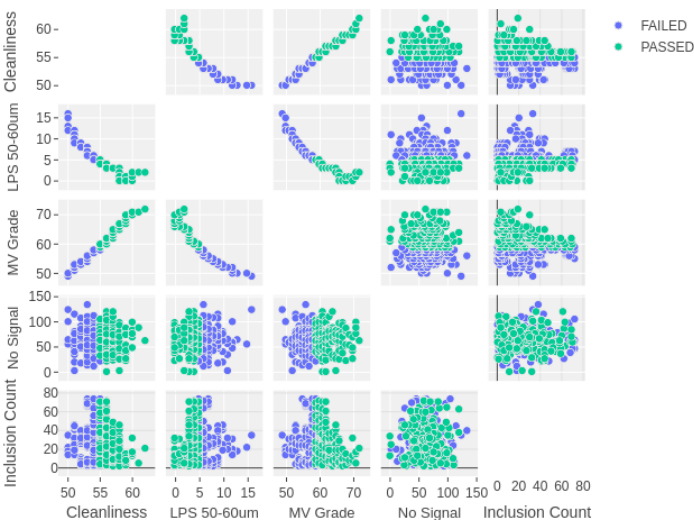


Figure 1. Scatterplot of numerical features coloured by sample result

The scatterplots show linear relationships between the cleanliness, MV grade and the LPS 50 - 60 μ m features. This is consistent with the fact that the MV grade is an estimate of the cleanliness without attenuation, and that the number of particles in the metal is inversely proportional to the cleanliness of the metal. The inclusion count and no signal features show no strong correlations to the other features. The “passed” category of the sample result shows a linear separation with all the features, except for some overlaps with the “failed” result around the centers. This is an indication that the cleanliness of the metal might have a strong influence on the result of the sample.

3.2. Anomaly Detection

The statistical process control framework establishes upper and lower control limits for variables [9]. These limits can be used to form triggers for anomalous events in production. Four variables are treated as the control variables:

1. Cleanliness index. The cleanliness index indicates the cleanliness of the cast.
2. The largest particle size count for particles between 120 and 140 μ m (LPS 140 - 160).
3. The largest particle size count for particles between 140 and 160 μ m (LPS 140 - 160). These two LPS variables represent the biggest sized inclusions, which are the most harmful to metal quality.
4. The inclusion count. This gives an indication of the abundance of inclusions, which can indicate when an anomalous injection of inclusions becomes present in the metal.

Control Charts

The run charts for the control variables are given in the following grid plot:

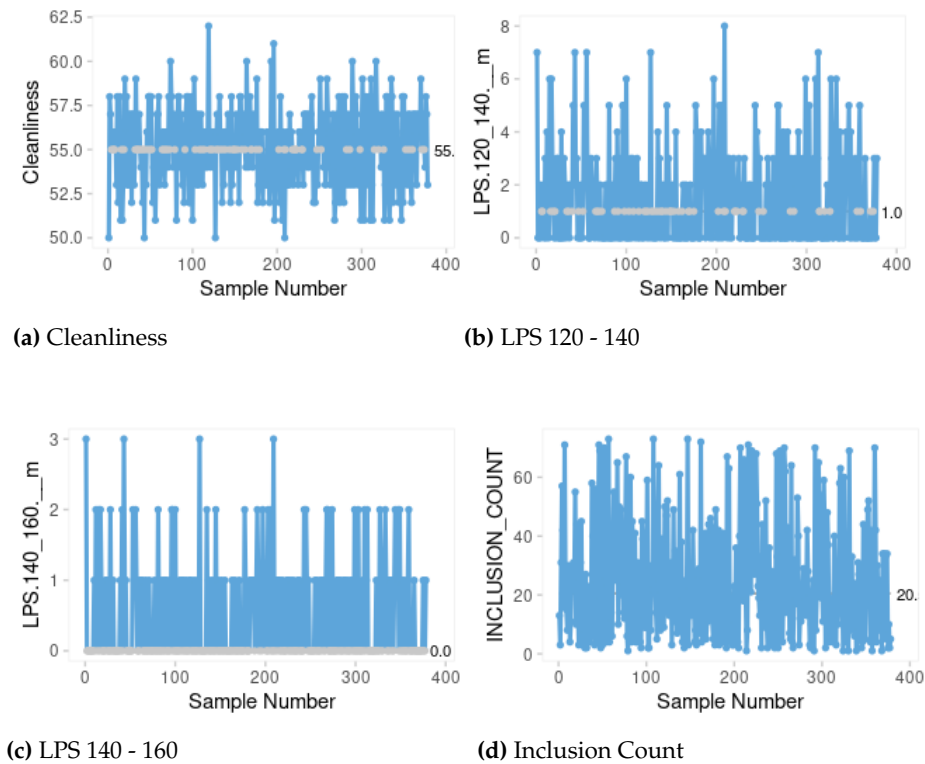


Figure 2. Run charts for control variables

As can be seen, the run charts show the time series progression of the datasets and the center lines. The individual charts for the control variables are shown in the following grid plot:

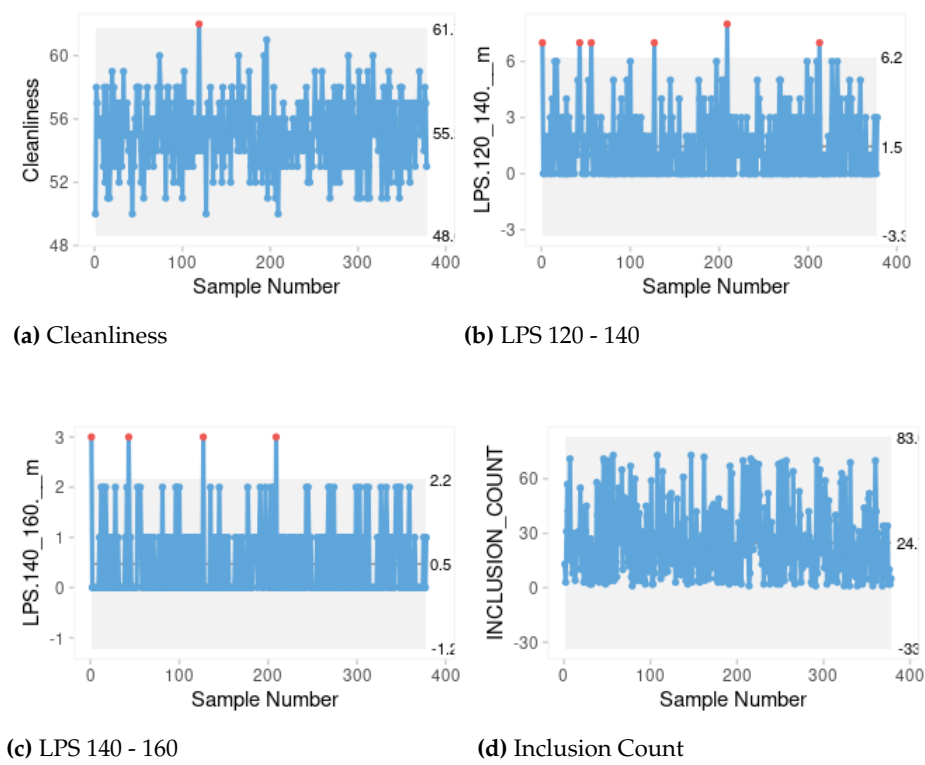


Figure 3. Individual charts for control variables

The individual charts show points outside control limits for the LPS control variables. This is an indication of points where the values were higher than 1.5 standard deviations from the center line [10,11]. They are correctly flagged as anomalies, and in a production environment, would prompt appropriate action and a decision for the quality of the cast.

The moving range charts for the control variables are shown in the following grid plot:

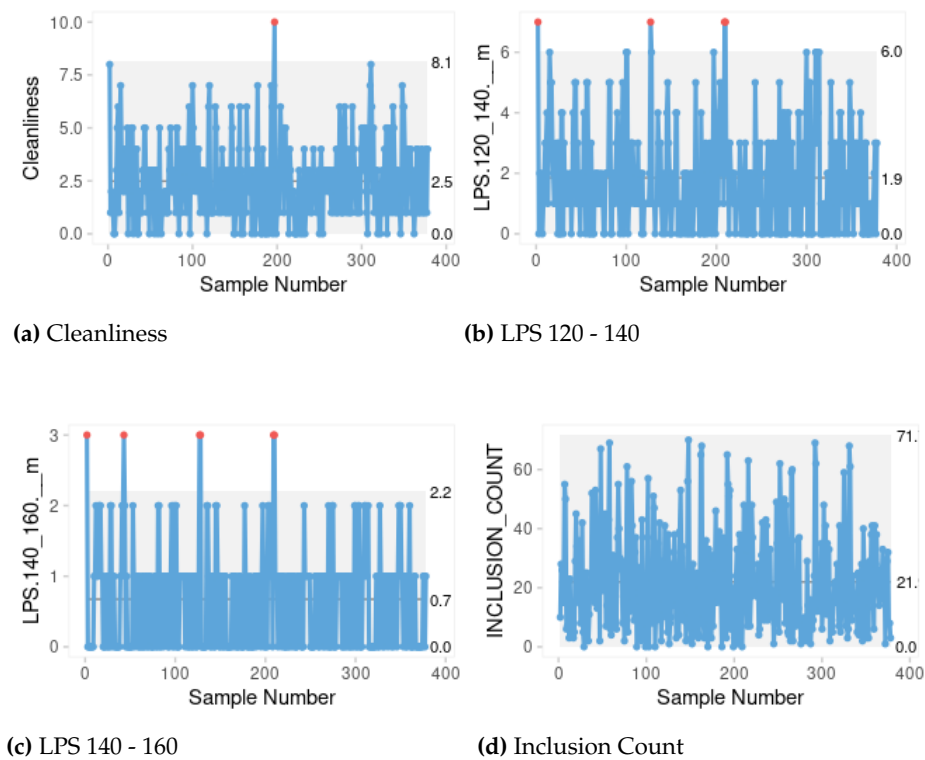


Figure 4. Moving range charts for important features

The range chart also indicates anomalous events for the LPS variables, including the one point for the cleanliness. This indicates that there are jumps in the average values of the control variables, and they can be likely attributed to certain causal events that are not part of normal operations.

Multivariate Charts

There exist three variants of the k-means algorithm, namely the Hartigan-Wong, Lloyd and MacQueen [12]. These algorithms are compared in the following figure:

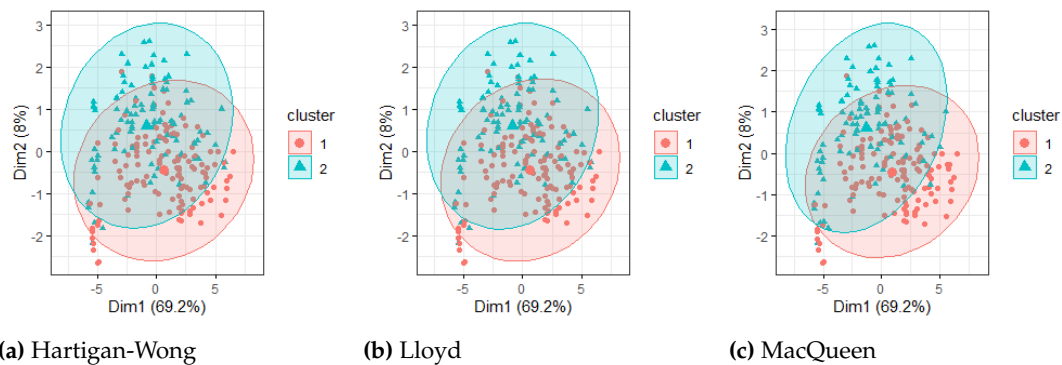


Figure 5. K-means cluster with two centroids

The following observations are made with respect to the k-means clusters:

1. The variance accounted for by the clusters is 77.2%. This is deemed adequate to represent the variance of the data, as it accounts for over two thirds of the variance.
2. There exist substantial spatial overlaps in the clusters. This can be seen on the number of points within the overlapping region.
3. Most of the data is concentrated between the two clusters. This indicates that the overlapping region represents good process performance.
4. The outliers constitute a minority of the data and could potentially indicate a process drift.

The k-means method is therefore considered adequate to be used as an anomaly detection technique, in which outliers can be flagged as anomalies. It is also noted that the three algorithms provided the same performance.

The DBSCAN method is applied to assess its clustering capability. The following figure shows the clustering when a small value of ϵ is applied. The minimum number of points, which is needed by the algorithm, is set at 10. The clusters are shown in the following figure for different values of ϵ :

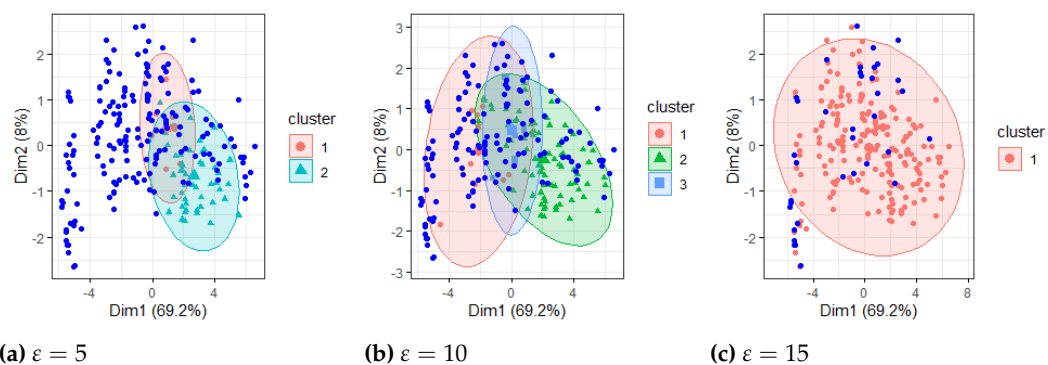


Figure 6. DBSCAN clusters for different center distances

The clusters show a gradual improvement, until a saturation point, when the distance has covered all points in the cluster at $\varepsilon = 15$. At this distance, the algorithm still recognises a substantial number of points within the 95% confidence interval ellipse as outliers. This is because it is a density-based algorithm [13].

3.3. Supervised Learning Classification

For supervised learning, the aim is to achieve classification by teaching algorithms using labelled datasets. The labels used in this study are the two categorical variables, namely sample result and inclusion count. The classification metrics used to assess model performance are accuracy, precision, sensitivity and specificity [14].

3.3.1. Logistic Regression

Logistic regression uses the logit function to perform a regression, and the output is treated as a categorical outcome [15]. The repeated cross-validation loss curve for the model is given in the following figure for the sample result target respondent:

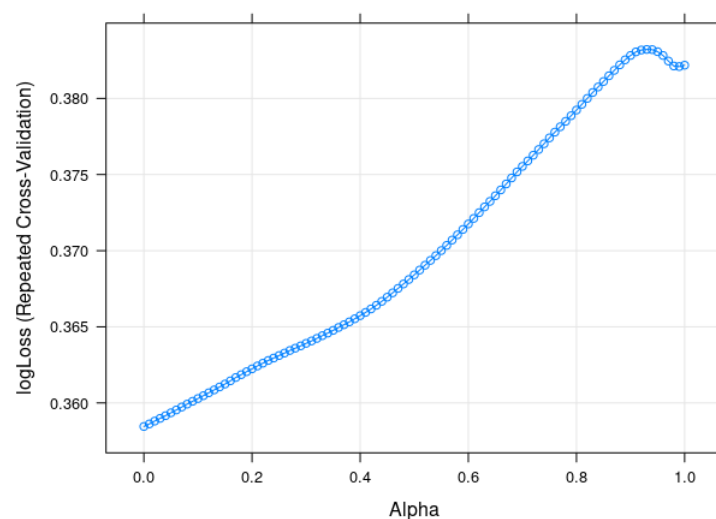


Figure 7. Sample Result target respondent cross-validation model training for different values of α

The curve shows a steady increase in log-loss as alpha increases, peaking around $\alpha = 0.9$. The optimal value of alpha is therefore 0, where the training loss is at its lowest.

The repeated cross-validation loss curve for the model is given in the following figure for the inclusion type target respondent:

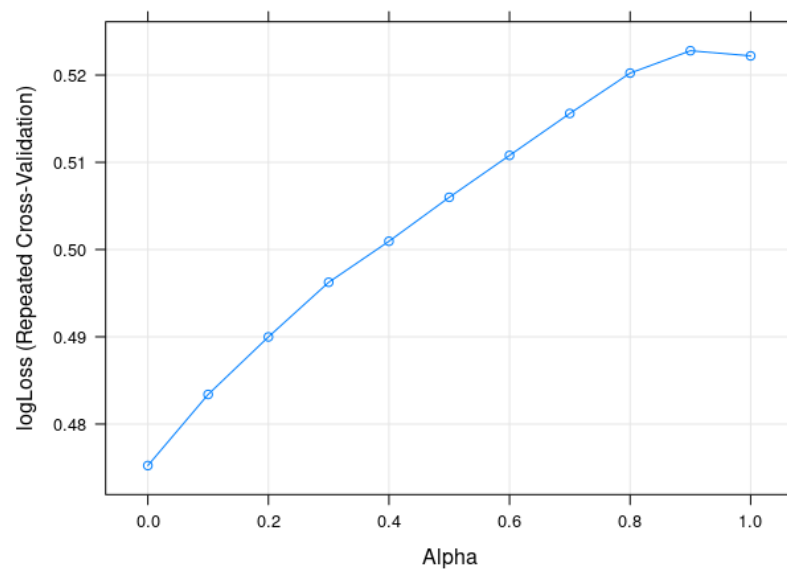


Figure 8. Inclusion type target respondent accuracy model training for different values of α

The curve shows that the training loss is at its minimum when $\alpha = 0$. This is therefore the optimal hyperparameter used to build the final model.

3.3.2. Support Vector Machine

The support vector machine has four main configurations, which are discussed below [16,17]:

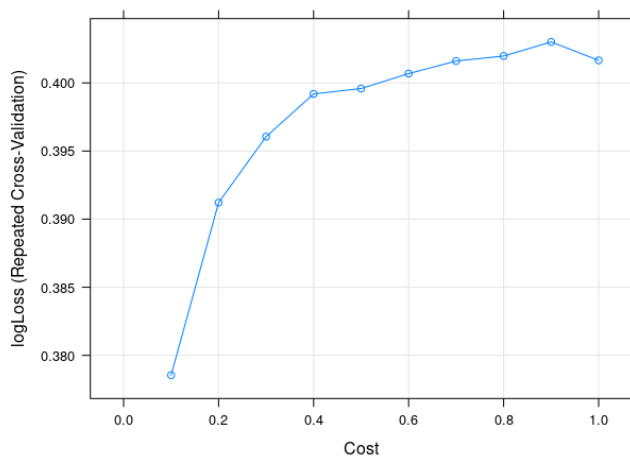
Linear The first parameter to optimise is the linear **cost function**, which is common among all the variants of the SVM model. In order to find the optimum cost coefficient, a linear variant of the activation function is used, and the cost function is incremented.

Polynomial The polynomial **degree** is another variant of the SVM that uses a polynomial function to separate the hyperspace. The degree of the polynomial is the hyperparameter to be optimised.

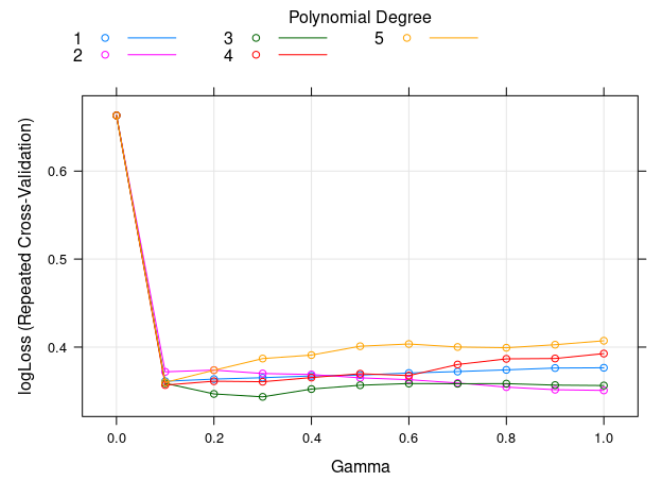
RBF The **Gamma** coefficient for the radial basis function optimises the radius of influence and therefore the sensitivity of the model to training data.

Kernel The kernel SVM uses a kernel function to search for the optimal hyperspace. In order to compare the kernel functions, the optimal hyperparameters are set for each kernel function respectively, and the training performances of the kernel functions are compared.

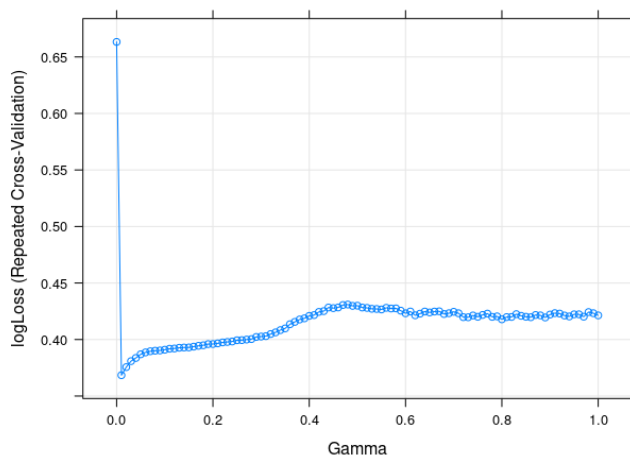
The following figures show the hyperparameter plots respectively as they are swept from zero for the sample result target:



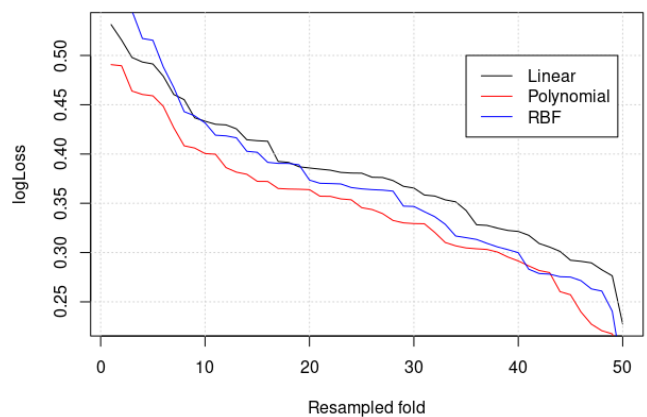
(a) Cost function



(b) Polynomial degree



(c) RBF Gamma



(d) Kernel function

Figure 9. SVM hyperparameters

The training results reveal the following:

- The optimal cost function is determined to be 0.1 as the loss of the model is minimal at that value. This value is therefore used for all the variants of the SVM.
- The log-loss curves show that the degree of 3 is the optimal degree for the polynomial variant of the SVM model. This is because it has the lowest loss at a corresponding gamma value of 0.3. These are therefore the selected hyperparameters for the polynomial variant.
- The best performance for gamma is at 0.01, where the lowest log-loss is achieved. This is therefore used to train the final model.

- The loss functions for the different kernels show little difference in performance. The polynomial kernel appears to provide the best loss, followed by the RBF kernel. The differences are negligible, which indicates training convergence. This implies that the polynomial and RBF kernel functions can be used with negligible difference in performance. The RBF kernel, however, is more computationally expensive, and therefore the polynomial kernel is used in the final model.

The following figures show the hyperparameter plots respectively as they are swept from zero for the inclusion type target:

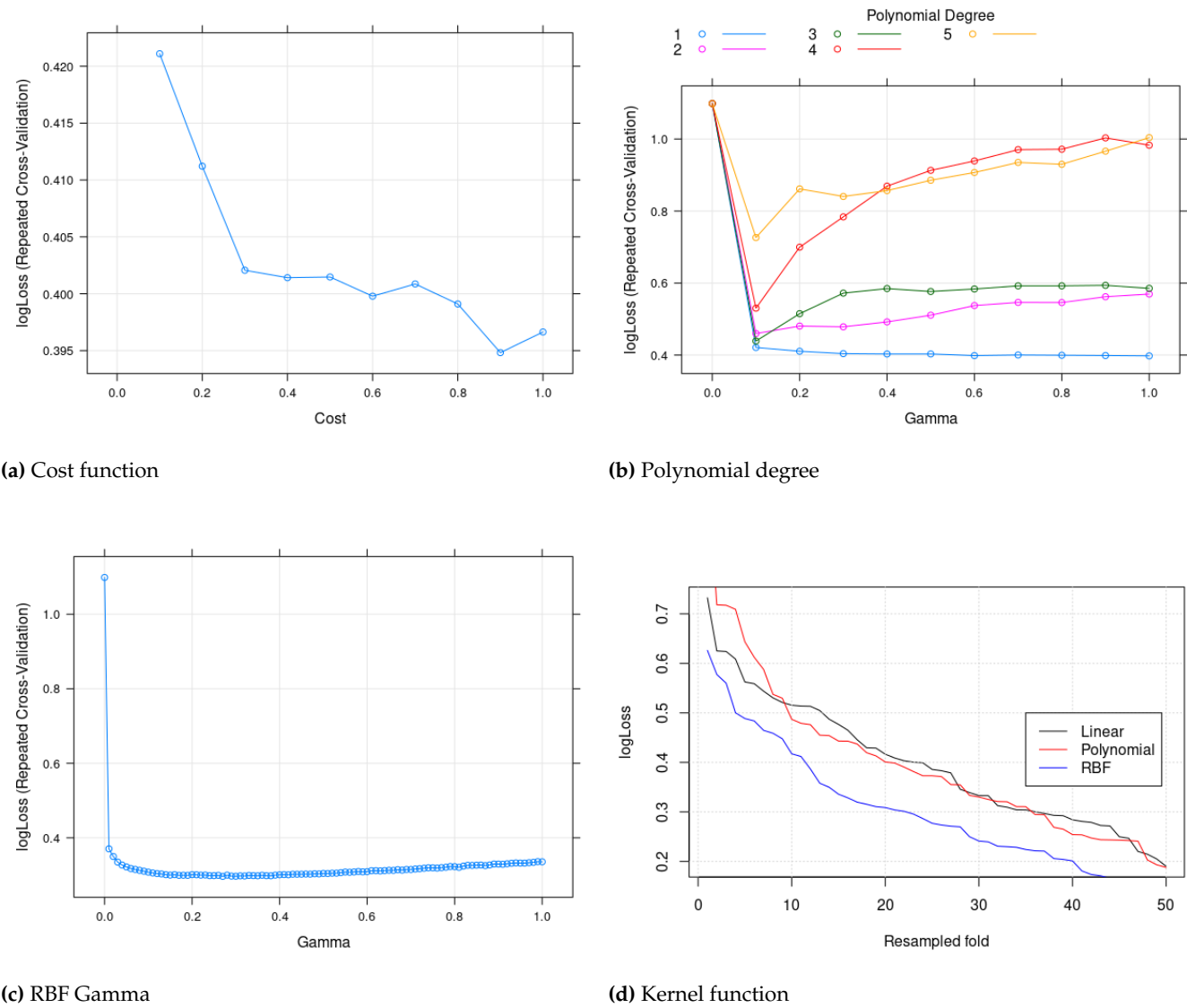


Figure 10. SVM hyperparameters

The following observations are made:

- The log-loss function sharply decreases down to a minimum of 0.395, where the cost function is 0.9. This is therefore the value used for training the SVM.
- The curves show that for higher degrees of the polynomial, the loss increases after a sharp drop at $\gamma = 0.1$. The first degree is the only order to maintain a decrease in the loss function for increasing values of gamma. The lowest loss is achieved at a value of $\gamma = 1$, where the loss is 0.4.
- From the curve, it can be seen that the loss function takes a sharp drop before slowly increasing. The optimal value of gamma is therefore where the loss makes a turning point, which is 0.27.
- The RBF has proven to be the optimal kernel function for fitting the data, as it offers the best overall performance in relation to the loss function. The linear and polynomial functions have comparable performance. The RBF is therefore the preferred kernel for building the final model.

3.3.3. Multi-Layer Perceptron

The multi-layer perceptron is an artificial neural network. It is the most basic form of the neural network, where the number of neurons, the number of layers and the activation functions can be tuned [18–21]. As a start, the model is trained with one hidden layer. The number of neurons and the activation function are optimised using cross-validation, and the model loss functions are presented in the following figure for the sample result target:

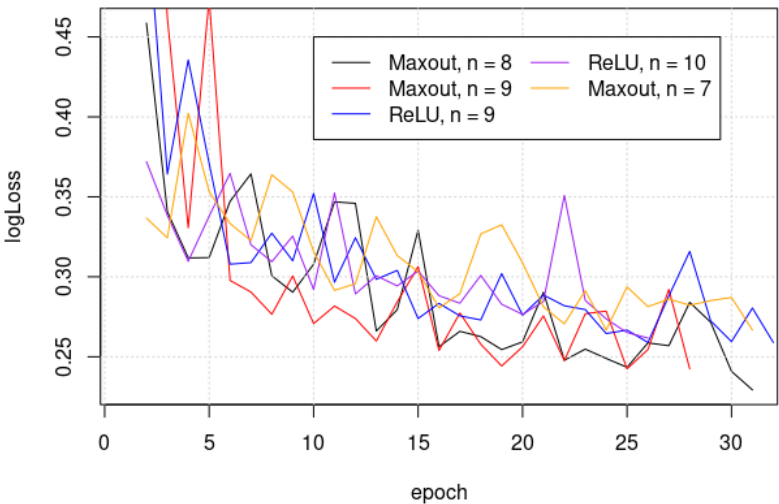


Figure 11. Sample result target respondent multi-layer perceptron training performance for one hidden layer. The activation function and number of neurons are the optimised parameters

It is difficult to tell from the model which of the combinations yields the best training performance. The maxout function with 8 neurons, however, appears to have the lowest training loss towards the last epoch [23]. The table below summarises the respective model configurations in order of increasing log-loss. As there are $4 \times 6 = 24$ models built from cross-validation, only the top 5 are presented.

Model	Hidden layers	Neurons	Activation function	Log-loss
Multi-layer perceptron	1	8	Maxout	0.2293
Multi-layer perceptron	1	9	Maxout	0.2425
Multi-layer perceptron	1	9	ReLU	0.259
Multi-layer perceptron	1	10	ReLU	0.2617
Multi-layer perceptron	1	7	Maxout	0.2667

Table 2. Sample result target respondent multi-layer perceptron training performance for one hidden layer

It is evident that the maxout activation function is dominating the performance, followed by the ReLU function. The optimal number of neurons for the first hidden layer is 8, as it presents the lowest training loss. The model might be overfitting in cases where $n > 8$. The addition of a second hidden layer, while keeping the units of the first hidden layer at the optimal value of 8, is presented in the following table:

Model	Hidden layers	Neurons	Activation function	Log-loss
Multi-layer perceptron	2	[8, 6]	Maxout	0.1356
Multi-layer perceptron	2	[8, 8]	Maxout	0.1622
Multi-layer perceptron	2	[8, 9]	Maxout	0.1973
Multi-layer perceptron	2	[8, 5]	Maxout	0.2038
Multi-layer perceptron	2	[8, 3]	Maxout	0.2203

Table 3. Sample result target respondent multi-layer perceptron training performance for two hidden layers

The table indicates that an additional hidden layer improves training performance. The best configuration involves the second hidden layer with 6 neurons. Since this is a significant improvement from the training performance of the model with one hidden layer, this configuration is the preferred one for building the final model.

The model training performance for one hidden unit is shown in the following figure for the inclusion type:

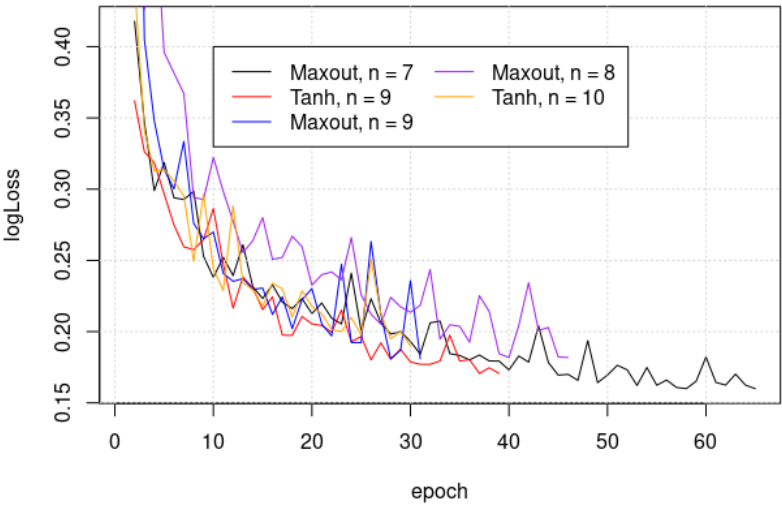


Figure 12. Inclusion type target respondent multi-layer perceptron training performance for one hidden layer. The activation function and number of neurons are the optimised parameters

The model configurations indicate comparable training loss performances, also indicating a convergence condition. The following table shows the model configurations ordered by increasing log-loss:

Model	Hidden layers	Neurons	Activation function	Log-loss
Multi-layer perceptron	1	7	Maxout	0.16
Multi-layer perceptron	1	9	Tanh	0.1707
Multi-layer perceptron	1	9	Maxout	0.1809
Multi-layer perceptron	1	8	Maxout	0.1818
Multi-layer perceptron	1	10	Tanh	0.1898

Table 4. Sample result target respondent multi-layer perceptron training performance for one hidden layer

The maxout activation function dominates the performance for the single hidden layer configuration of the model, followed by the tanh function. It is therefore the optimal activation function used in building the final model.

The second hidden layer is added to the configuration, and the training results are shown in the following table:

Model	Hidden layers	Neurons	Activation function	Log-loss
Multi-layer perceptron	2	[8, 3]	Maxout	0.1236
Multi-layer perceptron	2	[8, 8]	Maxout	0.1666
Multi-layer perceptron	2	[8, 9]	Maxout	0.1872
Multi-layer perceptron	2	[8, 5]	Maxout	0.1894
Multi-layer perceptron	2	[8, 10]	Maxout	0.2

Table 5. Inclusion type target respondent multi-layer perceptron training performance for two hidden layers

The performance for the configuration with the second hidden layer shows only a slight improvement from the configuration with a single hidden layer. This means that the configuration with a single hidden layer can be used without compromising too much training loss [24].

3.3.4. Radial Basis Function Network

Radial basis function networks are a specialisation of neural networks with a radial basis function as the activation function. They have been shown to have success in many cases where the boundary conditions are more complex [25–27,29]. The negative threshold tuning by means of repeated cross-validation is shown in the following figure for the sample result target:

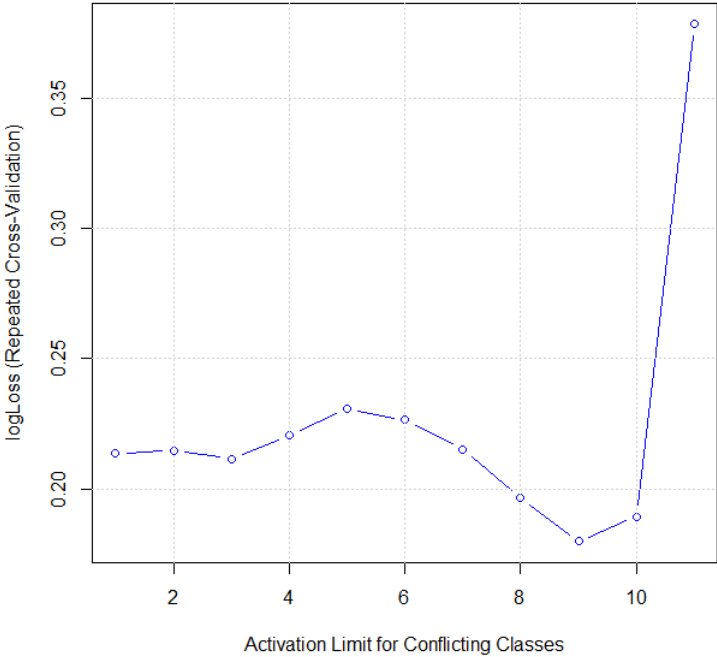


Figure 13. Sample result target respondent RBF training performance

The loss curve shows a dip at 0.8 and a sharp incline. The optimal threshold is therefore 0.8. The negative threshold tuning by means of repeated cross-validation is shown in the following figure for the inclusion type target:

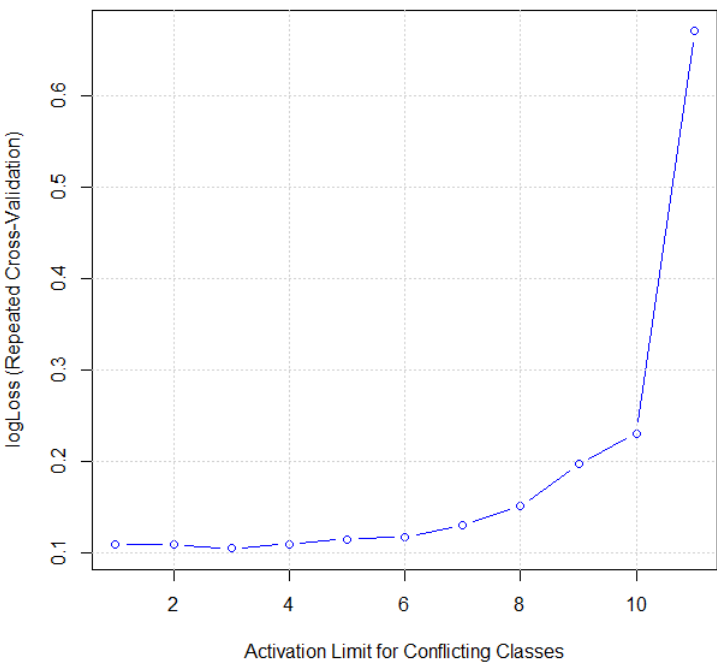


Figure 14. Inclusion type target respondent RBF training performance

The log-loss function has its minimum at a threshold of 0.2, before it steadily increases. The optimal threshold used is therefore 0.2.

4. Results

In this section, the models are tested on the test data split from the training data. The test data consists of 126 observations and constitutes 25% of the total data.

The test results are presented in the form of a confusion matrix, which quantifies how well the model performs on unknown data.

Within the context of unsupervised learning, tests data does not exist as all the data is unlabelled. This therefore means that unsupervised learning models have to be applied with domain knowledge in order to ensure the anomalies represent real life anomalies.

4.1. Supervised Learning Classification

The following table shows a side-by-side comparison of the models for the sample result target:

Metric	Logistic Regression	Support Vector Machine	Multi-layer Perceptron	RBF Network
Accuracy	0.91	0.95	0.95	0.92
Precision	0.95	0.93	0.96	0.95
Sensitivity	0.92	1	0.98	0.93
Specificity	0.9	0.85	0.9	0.9
ROC	0.91	0.88	0.91	0.91
Kappa	0.8	0.89	0.89	0.82

Table 6. Model performance comparisons for sample result target respondent

The following figure shows the comparison between the models:

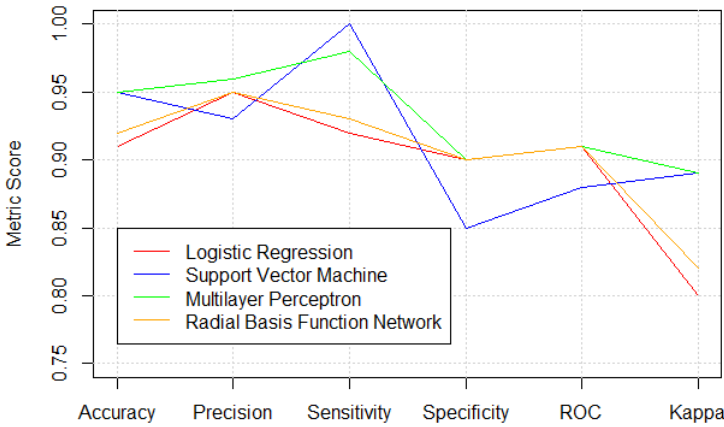


Figure 15. Model performance comparisons for the sample result target respondent

- The logistic regression model has performed satisfactorily as it satisfied the metrics except for accuracy, where it achieved 0.4% below the target. This is within the 95% confidence interval, so it is considered a success.
- The SVM model gave a better overall performance than the logistic regression model. It achieved a higher score for each of the performance metrics, with a perfect score for sensitivity. It is therefore regarded a success.
- The MLP model has so far shown the best performance as it has exceeded all the target scores.
- The RBF network model has also exceeded all target scores, although its performance is slightly below that of the MLP.

The models have all shown the capability to generalise well over the training data. This can be seen in the fact that the confusion matrices have shown good scores in testing performance over data that the models have not seen before. The logistic regression model, while the worst performing from the four, is still within the 95% tolerance of the target metrics. The MLP, SVM and RBF network models all performed well. The MLP gave the best performance, and is therefore recommended as the model to use. This is because the costs associated with each false alarm or miss are high within the context of an Aluminium manufacturing factory. Each loss can potentially cost the business hundreds of thousands of Rands.

For the multiclass problem, the metric scores are presented per class, so as to assess the performance of the model over individual classes in addition to the overall performance.

The following table shows a side-by-side comparison of the models:

Metric	Logistic Regression	Support Vector Machine	Multi-layer Perceptron	RBF Network
Accuracy	0.69	0.82	0.92	0.77
Precision	0.48	0.75	0.9	0.68
Sensitivity	0.57	0.75	0.89	0.69
Specificity	0.8	0.88	0.95	0.84
Kappa	0.4	0.62	0.84	0.53

Table 7. Model performance comparisons for sample result target respondent

The following figure shows the comparison between the models:

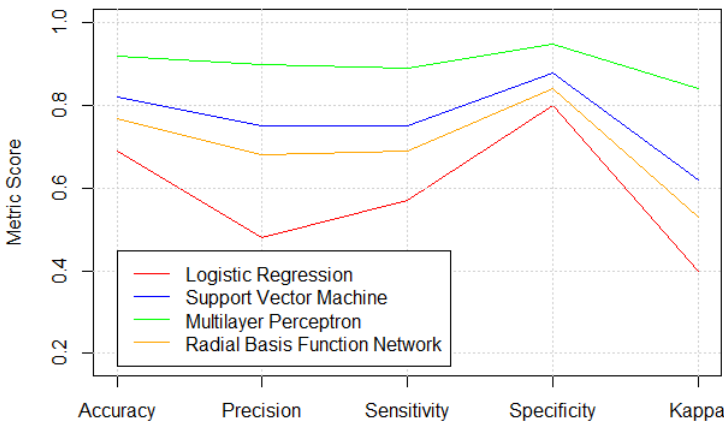


Figure 16. Model performance comparisons for the sample result target respondent

- The results show that the logistic regression model has scored below the target overall, except for specificity. Even for specificity, the per-class scores show that it achieved 0.63 for the SPINEL inclusion type, which is below target by 0.13. The best scores achieved are for FeO, which are also below target. This makes sense as the value of $\alpha = 0$ reduces the model to a constant logit function which is insensitive to the input.
- The SVM performance is better than the performance of the logistic regression model, with all the overall scores higher for the SVM than the logistic regression model. The model, however, did not meet all targets. The model scored above target only for the specificity class. The scores for accuracy, precision and sensitivity are not as far below target as for the logistic regression model. The value of kappa also indicates that there is substantial value in the model agreement with the dataset, as opposed to a completely random guess of the data [?]. The model, however, is considered inadequate as it does not satisfy the target metrics.
- The MLP is once again showing the best performance so far, with targets for precision, sensitivity and specificity met. The accuracy is slightly below target, but is still within the tolerance. The sensitivity and specificity have

been well exceeded, as the model especially gave few erroneous predictions for the SPINEL class. The MLP model is therefore considered a success.

- The RBF network model performance is worse than that of the MLP model for all the metrics. This implies that the application of radial basis functions as activation functions for the classification of inclusions gives a worse performance than applying maxout functions, which are used in the MLP.

The problem of generalising over the inclusion types has proven to be much more difficult to solve than predicting the outcome of the metal quality. This might be attributed to the following:

- The attenuation caused by the different inclusions is similar from an ultrasonic point of view.
- The inclusion sizes and counts for the different classes are similar and not easily separable. This could be due to the filter that the metal passes through just before the casting stage.
- The results of the metallographic analysis used to classify the inclusions are not entirely reliable due to operator error.

The MLP can therefore be considered as it provides the best results, and subsequent tuning of the model can improve performance.

4.2. Performance Optimisation

The previous subsection has shown that all the models are capable of providing good predictions over the sample result target respondent. The same cannot be said for the inclusion classification problem, as the prediction scores for the models were largely below target. In order to improve the model, hyperparameter tuning is considered with even more parameters.

4.2.1. Hyperparameter Tuning

The best performing model, namely the MLP, is tuned further in this section with the intention of assessing whether an improvement in performance can be achieved. In order to achieve this, more specialised tuning parameters can be iterated over using repeated cross-validation. The parameters are given in the following table:

Parameter	Value
model_id	multi-layer perceptron
number of hidden layers	1 (universal approximation)
number of neurons	8 - 10 (8 optimal, change for reference)
loss function	categorical crossentropy
activation function (hidden layer)	maxout
activation function (output layer)	softmax
epsilon	0 - 1 (selection randomness probability)
l1	0 - 0.2 (Lasso regularisation)
l2	0 - 0.2 (Ridge regularisation)
rho	0.9 - 1 (gradient descent term)

Table 8. Multi-layer perceptron model hyperparameters

The additional parameters from the table include:

- epsilon, which changes the selection randomness probability for the learning gradient. A large value of ϵ would mean that the learning diverges, while a small value would mean the the learning converges too slowly.
- L1, which is the Lasso regularisation parameter. It ensures that the model is penalised for learning loss so as to minimise the effect of some weights. A high value of L1 would see more weights being set to zero.

- $L2$, which is the Ridge regularisation parameter. It also penalises the cost function, but never sets the weights to zero.
- ρ , which is the learning rate decay factor. It is responsible for ensuring that the gradient descent is smooth. Higher values of ρ tend to give better smoothing results.

4.2.2. Grid Search Results

The grid search produced 187 500 models based on the given hyperparameters. This is because a grid search runs all possible combinations as opposed to a random search, which randomly selects a combination of hyperparameters. A random search is given a stopping metric, whereas a grid search guarantees the best possible performance based on the given hyperparameter ranges. It is slower and more computationally expensive.

The grid search results revealed the following points:

- The number of hidden layers does not significantly improve the performance of the model beyond neurons. It is therefore confirmed that keeping the number of neurons at 8 and applying the law of universal approximation (one hidden layer) is sufficient for achieving an optimal model.
- The regularisation parameters $l1$ and $l2$ do not have a significant effect on the training performance of the model. This can be seen in the grid search plot, where their values are closely related with respect to the loss function of the model.
- The gradient descent term ρ has an inversely proportional relationship with the training loss of the model. It can therefore be set at its highest value in order to achieve the lowest training loss.
- The selection randomness probability ϵ has an inversely proportional relationship with the training loss of the model. It can therefore be set at its highest value in order to achieve the lowest training loss.

The following table shows a summary for the parameters for the top 5 models based on the lowest training log-loss:

ϵ	hidden	$l1$	$l2$	ρ	logloss
1e-8	8	0	0.05	0.99	0.15622
1e-8	10	0	0.1	0.98	0.15838
3e-9	8	0.05	0	0.99	0.16224
4e-9	8	0	0.05	0.99	0.16750
8e-9	10	0.15	0	0.99	0.17360

Table 9. Grid search model log-loss performance

Based on the table, it can be seen that the training performance of the model does not improve much as the hyperparameters are changed. It should also be noted that the training performance of the model is comparable to that of the multi-layer perceptron prior to the employment of a grid search.

4.2.3. Final Model Results

The model is built based on the best parameters, and tested on the test data. The following confusion matrix shows the performance of the model:

(a) Confusion matrix

<i>Prd</i> \ <i>Act</i>	FeO	MgO	SPINEL
FeO	45	2	0
MgO	5	24	0
SPINEL	0	11	39

(b) Metric scores

Metric	Target	95% CI	FeO	MgO	SPINEL	Overall
Accuracy	0.95	0.9 - 1	0.94	0.86	0.91	0.86
Precision	0.9	0.86 - 0.95	0.96	0.83	0.78	0.86
Sensitivity	0.8	0.76 - 0.84	0.9	0.65	1	0.85
Specificity	0.8	0.76 - 0.84	0.97	0.94	0.87	0.93
Kappa	0.7	0.67 - 0.74	0.78			

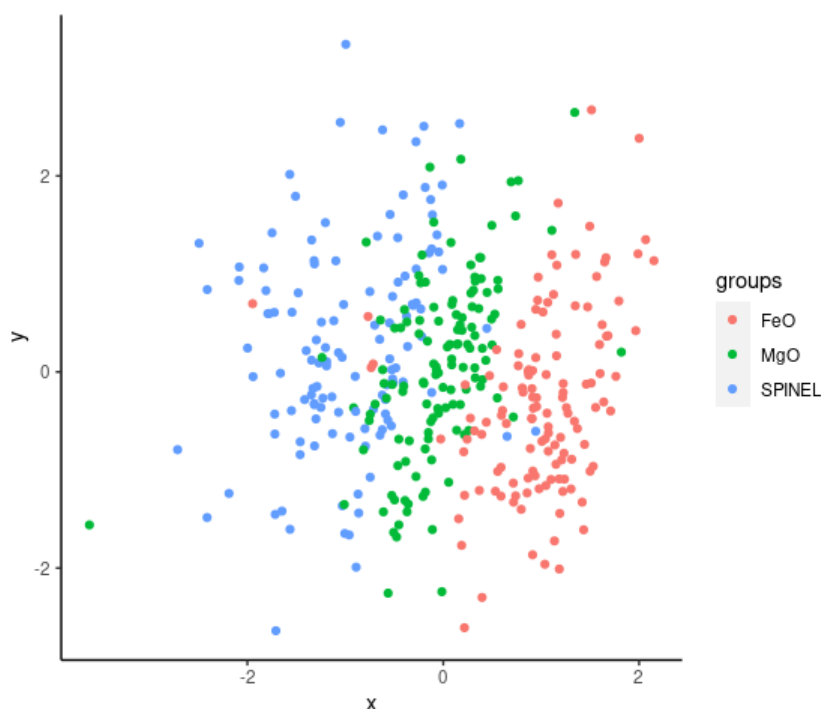
Table 10. MLP model performance after grid search

5. Discussion

Based on the confusion matrix and metric scores shown in the table, the following observations are made:

1. The model after grid search is not much better than the model before grid search. This is most likely an implication of the model having reached its learning potential.
2. The MgO inclusion has the worst performance. The metrics are below target except for specificity. This implies that the model is not able to generalise well over this inclusion type.
3. The SPINEL inclusion type is within the target limits except for the precision metric. For the other metrics, it has exceeded targets.
4. The FeO inclusion type has the best performance and has exceeded the targets for all metrics.

The model does not therefore generalise well over the inclusion types. A plot of the model's decision boundaries is shown in the following figure:

**Figure 17.** Model decision analysis grouped by inclusion type

As can be seen from the figure, there is substantial overlap between the SPINEL and MgO inclusion types. It is therefore unlikely that the model can separate the two classes sufficiently for it to reach all the performance metrics.

6. Conclusions

6.1. Summary of Work Done

An opportunity has been identified in an Aluminium manufacturing plant to improve quality control by means of a pulsed ultrasound system. This system is capable of performing real-time measurements on molten metal, which reveal the cleanliness of the metal. In order to automate the process of accepting the metal as clean, unsupervised and supervised learning approaches are applied.

The unsupervised component of this project focuses on anomaly detection for real-time alerting of operators and relevant personnel. This is achieved by exploring dimensionality reduction techniques including principal components analysis, K-means and DBSCAN clusters. A 95% confidence interval ellipse is drawn around the cluster as a means of identifying potential and would-be outliers.

The supervised learning component involves the development of a two-stage classifier. The first stage determines whether the metal quality is adequate for production. The second stage determines the dominant inclusion responsible for the quality deterioration. Four models are trained, namely logistic regression, support vector machine, multi-layer perceptron and a radial basis function network. While the inclusion type classifier gave a boundary performance on accuracy and precision, the values are within the 95% tolerance range. The project is therefore considered a success.

6.2. Recommendations for Future Work

During casting, the metal forms a thin oxidation layer on the surface, which is an indication of the presence of some inclusions at the top of the metal. A vision system can be employed to analyse the texture, colour and other visual properties of the metal in order to provide more insights relating to the nature of inclusions, the intensity of the inclusions and the effects of different casting parameters on the texture of the metal.

The attenuation levels of inclusions compared to pure Aluminium could produce different infrared signatures, which could be measured and analysed using Fourier Transforms. This is because different elements possess different reflectance and attenuation properties at different wavelengths. Classifiers can then be built to determine the types and intensities of inclusions based on the spectral properties of the measurements.

1. R. Gallo, *Differentiating Inclusions in Molten Aluminium Baths and in Castings*. Pyrotek Inc. OH, USA, 2017
2. E. Eckert and B. Cochran, *The Importance of Metal Quality in Molten Secondary Aluminium*. The Minerals, Metals and Materials Society. 2000
3. ABB Inc., *PoDFA | The complete solution for inclusion measurement*. Revision A01, 2016.
4. D. Veillette and D Paquin, *Metallographic Analyses*. International Aluminium Casting, Canada. 2006
5. ABB Inc., *Mobile liquid aluminium cleanliness analyser*. ABB Inc., Oerlikon, Zurich, Switzerland. July 2017. [Online]. Available: <https://new.abb.com/products/measurement-products/analytical/metallurgical-analysers/limca-iii>
6. ABB Inc., *LiMCA III | Liquid Metal cleanliness analyser*. 2017
7. D D. Smith, B Hixson, H Mountford and I Sommerville, *Practical Use of the MetalVision Ultrasonic Inclusion analyser*. JW Aluminium, 528 Old Mt Holly Road, Goose Creek, South Carolina, 29445, USA. 2015
8. R. Gallo, H. Mountford and I. Sommerville, *Ultrasound for On-Line Inclusion Detection in Molten Aluminium Alloys: Technology Assessment*. First International Conference on Structural Aluminium Castings, Orlando, Florida, USA. 2003
9. S. Agrawal and J. Agrawal, *Survey on Anomaly Detection using Data Mining Techniques*. State Technological University of Madhya Pradesh, Bhopal, India. 2015
10. G. James, D. Witten, T. Hastie and R. Tibshirani, *An Introduction to Statistical Learning: with Applications in R*. Springer, New York. 2013

11. V J. Hodge, *A Survey of Outlier Detection Methodologies*. Dept. of Computer Science, University of York, York, UK. 2004
12. M E. Celebi, H A. Kingravi and P A. Vela, *A comparative study of efficient initialization methods for the k-means clustering algorithm*. School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, Georgia, USA. 2012
13. M. Ester, H P. Kriegel, J. Sander and X. Xu, *A Density-Based Algorithm for Discovering Clusters*. Institute for Computer Science, University of Munich, Oettingenstr. 67, D-80538, Munich, Germany. 1996
14. S. Minaee, "An introduction to the most important metrics for evaluating classification, regression, ranking, vision, NLP, and deep learning models", *20 Popular Machine Learning Metrics. Part 1: Classification & Regression Evaluation Metrics*, Oct. 2019. Accessed on: Sept. 26, 2021 [Online]. Available: <https://towardsdatascience.com/20-popular-machine-learning-metrics-part-1-classification-regression-evaluation-metrics-1ca3e282a2ce>
15. P. Peduzzi, J. Concato, E. Kemper, T R. Holford and A R. Feinstein, *A Simulation Study of the Number of Events per Variable in Logistic Regression Analysis*. Departments of Medicine (Clinical Epidemiology Unit) and Epidemiology and Public Health, Yale University School of Medicine, New Haven, Connecticut, USA, 06510. 1996
16. C. Hsu, C. Chang and C. Lin, *A Practical Guide to Support Vector Classification*. Department of Computer Science, National Taiwan University, Taipei 106, Taiwan. 2016
17. T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning Second Edition*. Stanford University, Stanford, California. 2001
18. I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*. MIT Press, 2016. Accessed on: Oct. 4, 2021 [Online]. Available: <https://www.deeplearningbook.org/>
19. A. Abraham, *Artificial Neural Networks*. Oklahoma State University, Stillwater, OK, USA. 2005
20. P. Baheti, *12 Types of Neural Network Activation Functions: How to Choose?*. V Labs, 2021. Accessed on Oct. 5, 2021 [Online]. Available: <https://www.v7labs.com/blog/neural-networks-activation-functions#choose-activation-function>
21. S. Sharma, *Activation Functions in Neural Networks*. Towards Data Science, 2021. Accessed on Oct. 5, 2021 [Online]. Available: <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>
22. C. E. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, *Activation Functions: Comparison of Trends in Practice and Research for Deep Learning*. arXiv:1811.03378v1 [cs.LG]. 2018
23. I. J. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville and Y. Bengio, *Maxout Networks*. D'epartement d'Informatique et de Recherche Operationelle, Universit'e de Montr'eal 2920, chemin de la Tour, Montr'eal, Qu'ebec, Canada, H3T 1J8. 2013
24. K. Kawaguchi, L. P. Kaelbling and Y. Bengio, *generalisation in Deep Learning*. Massachusetts Institute of Technology and University of Montreal, 2020
25. Y. Wu, H. Wang, B. Zhang and K. L. Du, *Using Radial Basis Function Networks for Function Approximation and Classification*. Enjoyor Laboratories, Enjoyor Inc., Hangzhou 310030, China and Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada H3G 1M8. 2011
26. J. Park and I. W. Sandberg, *Universal Approximation using Radial-Basis-Function Networks*. Department of Electrical and Computer Engineering, University of Texas at Austin, Austin, Texas 78712, USA
27. N. B. Karayiannis, *Reformulated Radial Basis Neural Networks Trained by Gradient Descent*. IEEE Transactions on Neural Networks Vol. 10, No. 3. 1999
28. J. Moody and C. J. Darken, *Fast Learning in Networks of Locally-Tuned Processing Units*. Yale Computer Science Neural Computation 1. 1989
29. A. L. I. Oliveira, B. J. M. Melo, F. B. L. Neto and S. R. L. Meira, *Combining Data Reduction and Parameter Selection for Improving RBF-DDA Performance*. IBERAMIA 2004. IBERAMIA 2004. Lecture Notes in Computer Science, vol 3315. Springer, Berlin, Heidelberg. 2004