Machine learning in gold exploration

Brendan Garner

Abstract

To date, no other significant gold deposit has been discovered surrounding the [redacted] gold mine.

The purpose of this report is to determine whether a machine learning model could be developed to identify a hidden deposit.

An extensive geochemical dataset has been obtained from the substantial amount of drilling in the area. A robust Principal component analysis (PCA) was conducted on isometric logratio transformed data, confirming the importance of bismuth and tellurium for gold mineralisation. PCA outliers related to gold were isolated and subjected a clustering method that considered spatial proximity and attribute similarity. A Naïve Bayes classifier was also implemented on the full dataset.

It was found that clustering of PCA outliers when plotted on a map and combined with other data can be useful for identifying areas of interest.

The Naïve Bayes classifier achieved an overall accuracy of around 80%. Further improvement was deemed necessary before the model could be useful.

Introduction

Gold exploration in Australia has progressively moved to areas of deep transported cover as shallower targets have been exhausted (Anand and Robertson, 2012). Better use of down-hole geochemical data as an indicator for proximity to mineralisation provides an opportunity to increase value from exploration drilling (Hill, 2013).

Despite a substantial amount of drilling that has taken place surrounding the [redacted] gold mine, no further significant deposits has been discovered. An extensive multivariate dataset has been collected from drilling. This presents an opportunity to test whether a machine learning exploration model could assist in identifying a hidden deposit.

Data

The Source

The data used in this report was obtained from the [redacted] dataset in the [redacted] database via SQL query.

Collection method

The data was collected through an observational study, namely multiple air core (AC) drilling campaigns. The AC drilling contractor was instructed to drill to failure. The AC drilling method can penetrate the unconsolidated material close to the surface (regolith) but is able to penetrate more than a couple of metres into the fresh rock. Rock chips are brought to the surface using high pressure air and placed in piles on the ground corresponding to every metre drilled. A geologist assigned to the AC rig then takes a handful sized sample from the last pile (bottom of hole) using a sieve. The rock chips are then washed, bagged and sent for analysis

Sample size and sampling process

A total of 2,655 bottom of hole AC rock chip samples were sent to MinAnalytical Laboratory Services Australia Pty Ltd for aqua regia digest with ICPMS finish. Aqua regia is commonly used in exploration geochemistry and is a mixture of nitric and hydrochloric acids (MinAnalytical, 2018). Each sample was pulverized before a 10g subsample was analysed for gold (Au), silver (Ag), arsenic (As), bismuth (Bi), copper (Cu), nickel (Ni), lead (Pb), selenium (Sb), tellurium (Te), tungsten (W) and zinc (Zn).

Number and type of variables

Assay data on gold and 11 other elements for 2,655 samples were received and imported into the database. All variables are numerical.

Methods

The following procedure was conducted using R version 1.3.1093 (R Studio Team, 2020).

Data subset selection

Microsoft SQL Server Management Studio was used to query the [redacted] database, extracting collar, sample and assay data for all air core (AC) drilling in the [redacted] dataset. A subquery was used to isolate the bottom of hole sample data from every AC drill hole. All fields in the assay table were initially selected before a new query was written to select only fields with limited amounts of NULL values. The record set featuring 23 variables and 2655 rows was exported to CSV for data cleaning.

Data cleaning

Imputation was not considered appropriate, all rows with NULL values were removed from the data. Rows with barren (unmineralized) lithologies (rock types) of lamprophyre and Proterozoic dolerite were removed. Rows with [redacted] lithologies were also removed, leaving only rows with the target lithology and rows with a regolith unit as a logged lithology. The final dataset consisted of 2,452 rows with numeric fields for assay data recorded in parts per million for Au, Ag, As, Bi, Cu, Ni, Pb, Sb, Te, W and Zn.

The bottom of hole XYZ coordinates were calculated using the drill hole collar location in GDA94 / zone [redacted], collar dip, collar azimuth and drill hole length.

Principal Components Analysis (PCA)

R packages *dplyr* and *mvoutlier* were installed in R Studio, for basic data manipulation and performing a PCA, respectively. The CSV was imported into R with the gold and multi-element assay data assigned to an object. The data was isometric logratio (ilr) transformed by *mvoutlier* before conducting a Robust PCA. Geochemical data follow an Aitchison geometry and are required to be transformed by logratio to Euclidean geometry to overcome any spurious correlations (Filzmoser, Hron and Reimann, 2012). Of the three logratio transformations (additive logratio, centred logratio and isometric logratio), ilr is recommended (Filzmoser et al., 2012).

Standard PCA is highly sensitive to outliers (Polyak and Khlebnikov, 2017). As the purpose of the project was to identify outliers that could be used as a mineral exploration target, outliers were not removed from the data, requiring the adoption of a method (Robust PCA) that could handle grossly corrupted observations (Candès and Li, 2011).

Following the PCA on ilr transformed data, *mvoutlier* back transformed the results into centred logratio space to enable to PCA scores and loadings to be interpreted (Filzmoser, Hron and Reimann, 2009). The plot function of *mvoutlier* was used to show the two largest principal components, with all samples identified as multivariate outliers located on the biplot by their PCA scores.

A data frame was created in R to hold the PCA scores and bottom of hole XYZ coordinates. Rows were removed with negative scores for the first principal component (PC1) and only PCA outliers were kept, leaving 116 multi-element samples with anomalous Au-Bi-Te values (Figure 5). These values were used in the clustering algorithm.

Clustering

Professional geographic information system (GIS) software ArcGIS Pro has a function for spatially constrained multivariate clustering (Esri, 2020). The application uses a connectivity graph (minimum spanning tree) and a method called SKATER (Spatial "K"Luster Analysis by Tree Edge Removal) to find natural clusters in the data, based on the procedure outlined by AssunÇão et al. (2006).

The same procedure was followed using R in RStudio. The first step is to convert the XY coordinates into a SpatialPoints object, using the *tri2nb* method from the *spdep* library. The spatial points are converted to a graph representation of neighbours using a Delaunay triangulation. This represents all Au-Bi-Te samples that were identified as PCA outliers, plotted in XY space (Figure 1).

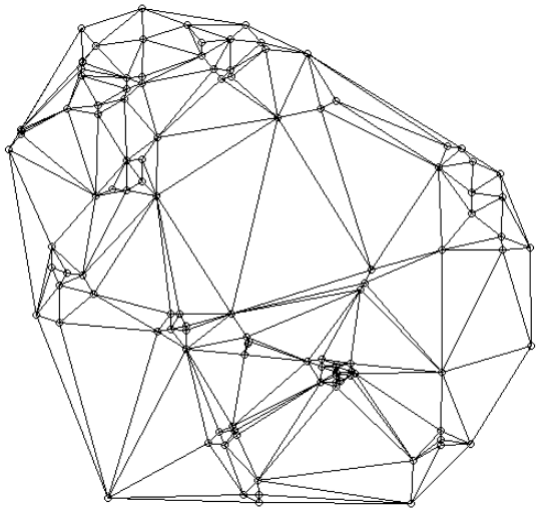

Figure 1. Delaunay triangulation of PCA outliers

The next step is to use the *nbcosts* function to calculate the cost of each edge that connects two nodes (samples). The edge cost is proportional to the dissimilarity between the ilr transformed Au-Bi-Te values of the neighbouring pair (AssunÇão et al., 2006). These costs are assigned to a list using the *nb2listw* function and are then used to calculate the minimum spanning tree (MST) using the function *mstree*. The MST is formed by pruning the edges of the interconnected graph (Figure 1) that have high dissimilarity. The pruning produces a reduced graph with edges that join similar areas. The MST is further pruned using the *skater* package into a number of connected subsets of nodes or spatial clusters (k). This number (k)

is set by the user and an appropriate number requires domain knowledge. In this instance, k was set to 10 (Figure 2).

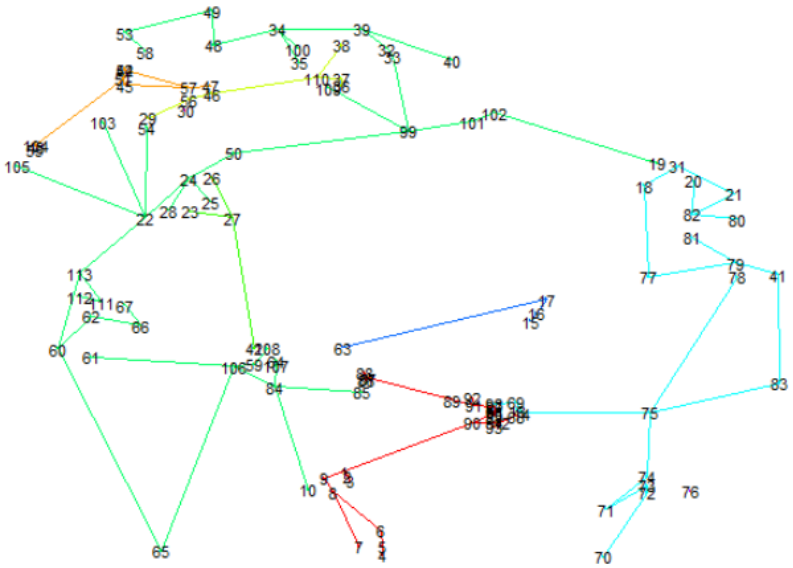

Figure 2. SKATER applied to the minimum spanning tree with K=10

Naïve Bayes Classifier

The isometric logratio transformed values of all elements (Au, Ag, As, Bi, Cu, Ni, Pb, Sb, Te, W and Zn) were plotted on a Q-Q plot to test for data normalcy (Figure 3). The data appears to be near normal and therefore passes one of the assumptions of the Naïve Bayes Classifier (NB) model. An examination of the PCA chart in Figure shows numerous elements grouping together and this suggests that the elements are unlikely to be statistically independent. This fails one of the assumptions under the NB model. However, the model usually exhibits good performance even when the independence assumption is violated (Hill et al., 2014).

The purpose of the project is to find gold mineralisation and therefore the chosen target of Au_ppm was selected. Gold assays are reported to the lowest detection limit (LDL) of 0.001 ppm with values below that recorded in the database at half LDL (0.0005 ppm). For the purposes of implementing a NB model, gold values which are continuous variables were converted into binary target variables. 2004 samples with untransformed gold values at or above LDL were recorded as Au_target = 1, and the remaining 448 samples below LDL were recorded as Au_target = 0.

The *caret* package was used to implement the NB model. The function *createDataPartition* was used to randomly split the data into a training dataset (80% or 1,963 samples) and a testing dataset (20% or 489 samples). The *trainControl* function was used to randomly partition the test dataset into 10 (k) equal subsamples (or k-folds). One subsample was retained as validation data to test the model and the remaining nine subsamples were used as training data. This technique called cross-validated was conducted a total of 10 times (k=10), during which each of the k subsamples were used exactly once as the validation data. The results are then averaged to produce a single estimation (James et al., 2013).

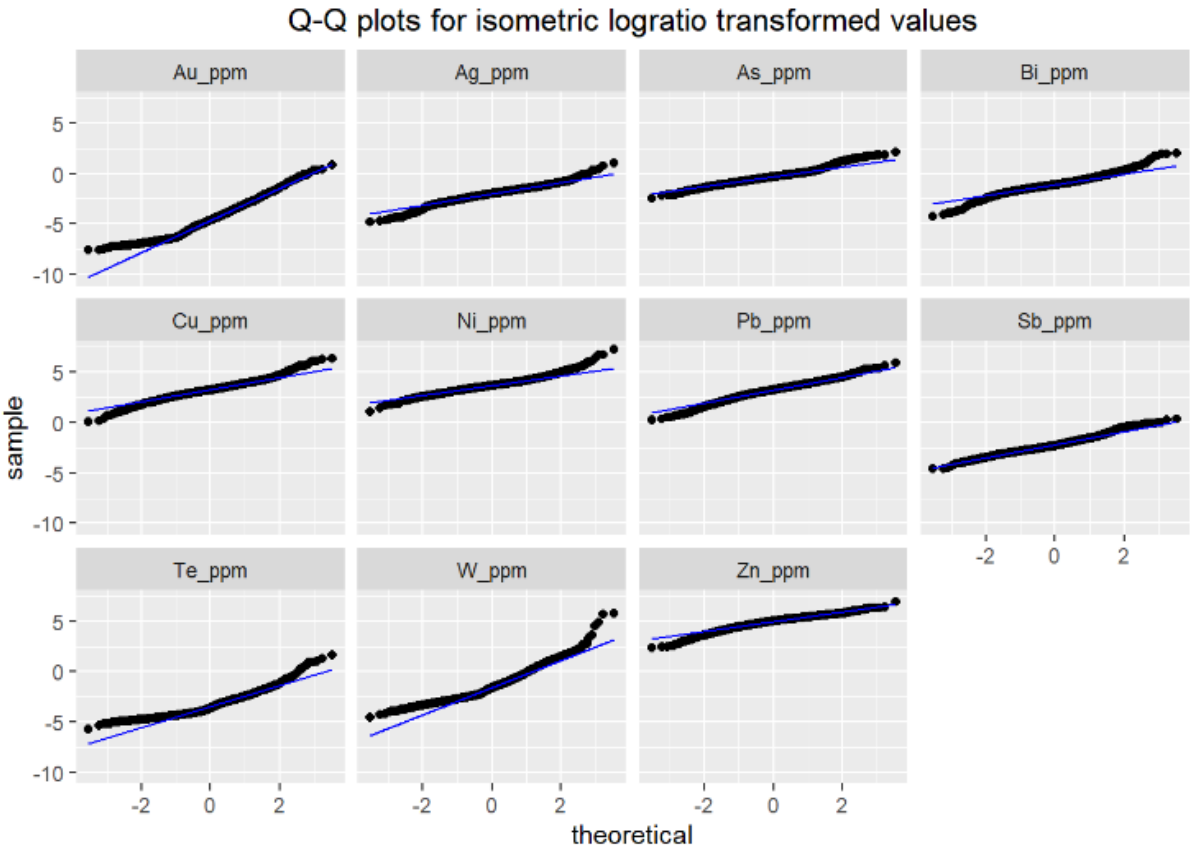

Figure 3. Q-Q plots show the data to be near normal

Results and discussion

Principal Components Analysis

A scree plot is shown in Figure 4 which highlights the contribution to the dataset variance for each of the principal components. This plot reveals that the variance can be explained by only a small number of principal components. A subjective test is applied when examining the chart to identify the 'elbow' in the curve or the point at which the proportion of variance explained by each subsequent principal component drop off (James et al., 2013). The plot in Figure 4 shows that 3 principal components can explain almost 76% of the total variance.

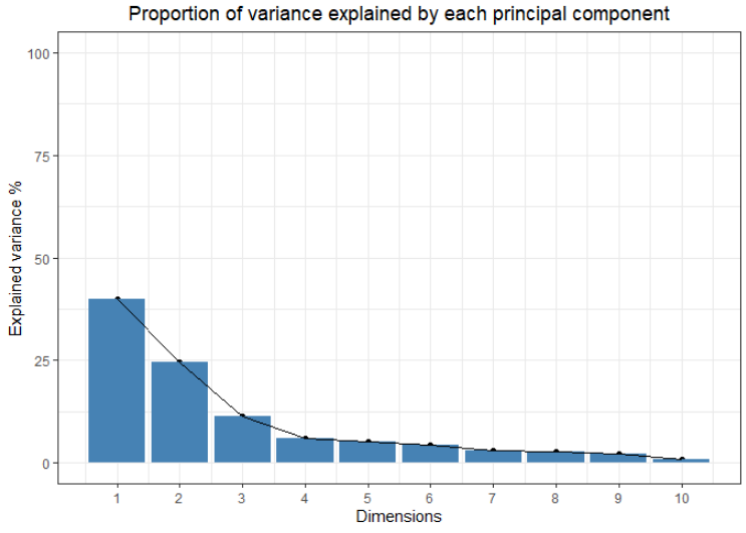

Figure 4. Scree plot of multi-element principal components

The two largest principal components are plotted in Figure 5, with the main interest being the angles between each ray (Filzmoser et al., 2012). This reveals that gold has a strong association with bismuth and tellurium, which confirms existing knowledge on the area. All samples identified as being PCA outliers are marked on the chart with a cross. The colours correspond to the level of 'outlyiness' with red signifying the largest outliers. 116 PCA outlier samples associated with gold (close to the Au_ppm line) with PC1 values >= 0 are outlined in the dashed box were separated from the dataset

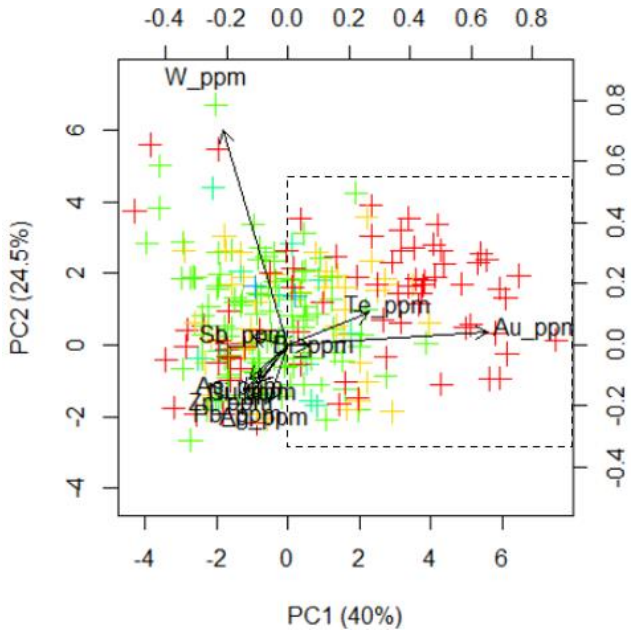

Figure 5. bismuth and tellurium are strongly associated with gold

Clustering

The clustering algorithm outlined by AssunÇão et al. (2006) and used by ArcGIS Pro was recreated in R for this project. The algorithm assigns clusters based on spatial proximity and attribute similarity. The algorithm clustered 116 PCA outliers into 10 groups using the *spdep* package. An ID associated with the cluster group was recorded for each PCA outlier and was added to the data frame before exporting to CSV. The CSV contained location data and was imported into mapping software, ArcGIS. A feature class was created from the imported CSV and loaded to the map. The layer symbology was adjusted with each cluster being assigned a different colour. The circle symbols were enlarged so that they stood out. Maximum gold values for every drillhole (aircore, reverse circulation and diamond) was extracted from the database and the downhole coordinates were calculated using the collar position, collar dip, collar azimuth and downhole depth of the sample. These are displayed on the map as small semi-transparent circles with a graduated colour scheme, darker colours representing higher gold grades.

The assignment of k number of groups is subjective. One characteristic remained stable when varying k. It is apparent that in areas with higher gold mineralisation, multiple PCA outlier groups are present. Despite being in close proximity, the PCA suggests the samples have noticeably different geochemistry. This is possibly due to the rock alteration from the passage of mineralised fluids. The area marked by a yellow circle in Figure 6 features multiple groups

of PCA outliers (combined Au-Bi-Te) and intersecting faults that may structurally control the mineralisation. The area is a known prospect but may require further investigation.

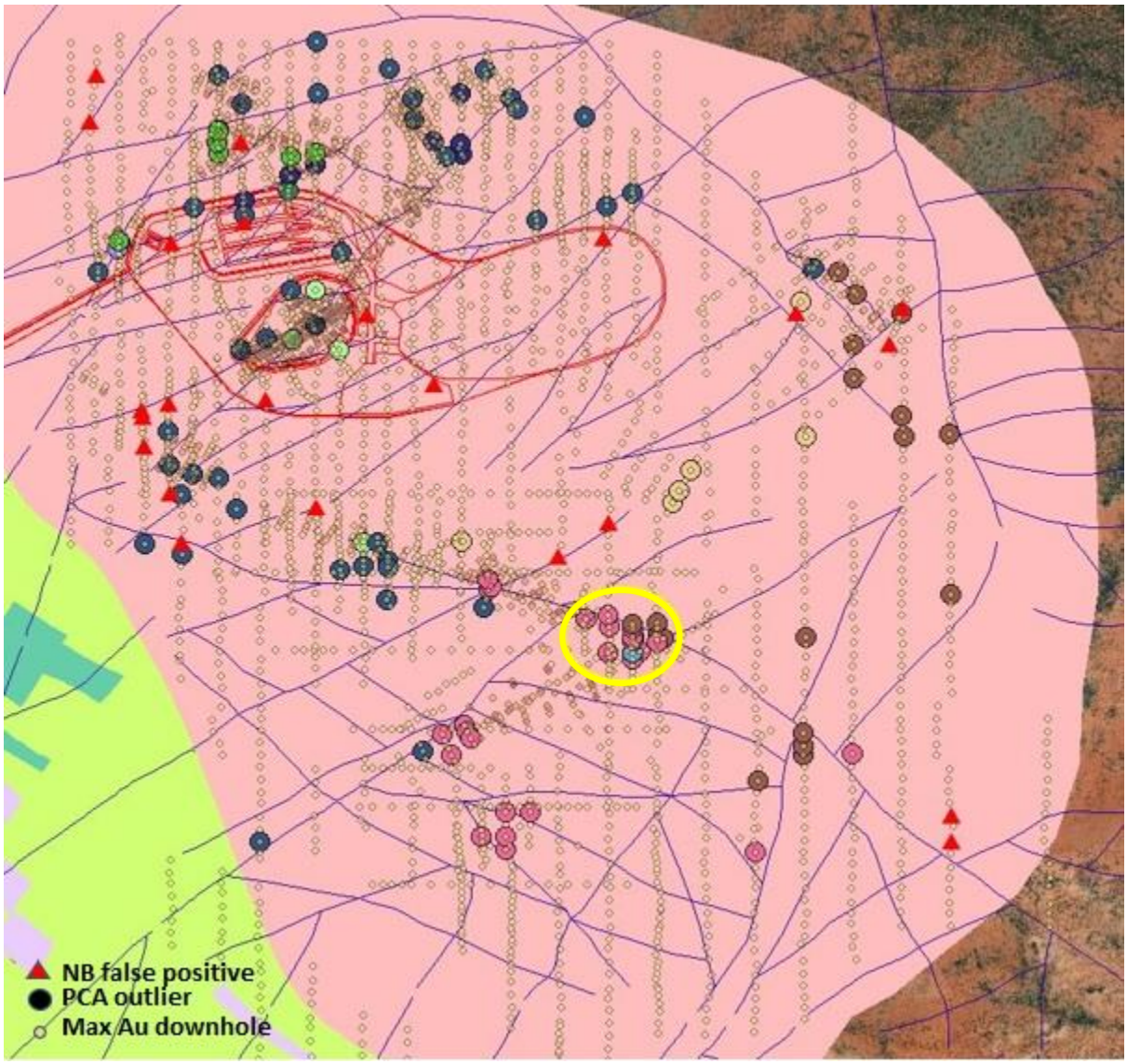

Figure 6. Principal component outliers (Au-Bi-Te) grouped by spatial proximity and attribute similarity. Multiple distinct geochemical groups are associated with increased gold mineralisation. A drill target with geochemical alteration is located on intersecting faults and is marked by a yellow circle

Naïve Bayes Classifier

The NB classifier obtained an overall accuracy of 80.34% using the training dataset of 1,963 samples. The ability of the model to correctly predict positive results is called the sensitivity. It is defined as the ratio of true positives to true positives and false negatives

$$\frac{TP}{TP + FN}$$

The NB classifier had 83.7% sensitivity to the training data.

$$\frac{68.4}{68.4 + 13.3}$$

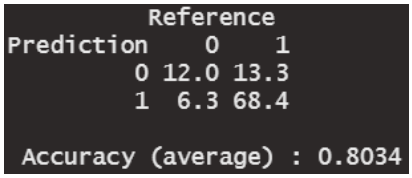

Figure 7. NB performance on training data

The ability of the model to correctly predict negative results in called the specificity. It is defined as the ratio of true negatives to true negatives and false positives.

$$\frac{TN}{TN + FP}$$

$$\frac{12}{12 + 6.3}$$

The NB classifier had 65.5% sensitivity to the training data. The NB classifier achieved an overall accuracy of 80.2% on the test data of 489 samples with a sensitivity of 82% and a specificity of 73%.

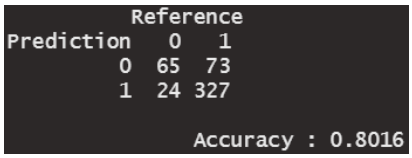

Figure 8. NB performance on test data

It is not surprising that the model outperformed in positive prediction compared to negative prediction. This discrepancy is likely due in part to an unequal training dataset, where samples above LDL were four times as frequent as samples below LDL.

Conclusion

The [redacted] dataset contains sufficient observations of multivariate data to allow for the implementation of machine learning exploration techniques. The majority of the drilling occurred in the same rock unit enabling the comparison of geochemical ratios across a large spatial area. Conducting a robust principal components analysis using isometric transformed multivariate data provides a theoretically sound method of identifying key pathfinder elements (strongly associated with gold). Bismuth and tellurium were identified as pathfinders, which confirms existing knowledge on the area.

Using graphs to cluster data for spatial proximity and attribute similarity is the method used by leading GIS software, ArcGIS Pro. While the choice of k clusters is somewhat subjective, the method does appear to successfully highlight spatial areas that have undergone geochemical alteration. When coupled with structural complexity, these areas represent potential drill targets for gold exploration.

The Naïve Bayes classifier had a relatively low overall accuracy of 80.34%. The model performed better when predicting samples with gold grades above 0.005ppm (LDL) compared to predicting samples below LDL. This discrepancy is likely due in part to an unequal training dataset, where samples above LDL were four times as frequent as samples

below LDL. Accuracy of the model could be improved by training the model on a subset of the data with equal proportions of samples above and below the LDL. Further refinement and advanced techniques would be necessary to improve the model to a level that could be useful for mineral exploration. Once accuracy has been improved, the false positives should be examined. These will represent samples that have statistically favourable geochemistry for gold but due to the sampling process or a nuggety occurrence of gold, the assay process was unable to detect its presence.

References

Anand, R, R., Robertson, I, D, M. (2012). The role of mineralogy and geochemistry in forming anomalies on interfaces and in areas of deep basin cover: implications for exploration. *Geochemistry: Exploration, Environment, Analysis.*12(1). 45-66.

AssunÇão, R, M., Neves, M, C., Câmara G., Da Costa Freitas, C. (2006). Efficient regionalization techniques for socio-economic geographical units using minimum spanning trees. *International Journal of Geographical Information Science.* 20:7, 797-811.

Comprehensive R Archive Network. URL https://cran.r-project.org/bin/windows/Rtools/

Candès, E, J., Li, X. (2011). Robust Principal Component Analysis? *Journal of the ACM*, 58, 3, Article 11.

Esri (2020). ArcGIS Pro. URL https://pro.arcgis.com/en/pro-app/tool-reference/spatial-statistics/how-spatially-constrained-multivariate-clustering-works.htm

Filzmoser, P., Hron, K., Reimann., C. (2009). Principal components analysis for compositional data with outliers. *Environmetrics, 20, 621-632.*.

Filzmoser, P., Hron, K., Reimann., C. (2012). Interpretation of multivariate outliers for compositional data. *Computers & Geosciences, 39, 77-85.*

Hill, S, M. (2013). Regolith Geochemistry: Expressions of Mineral Systems between the Fresh Rock and Fresh Air. *Advances in exploration and ore deposit geology.*

Hill, E. J., Oliver, N, H, S., Fisher, L., Cleverley, J, S., Nugus, M, J. (2014). Using geochemical proxies to model nuggety gold deposits: An example from Sunrise Dam, Western Australia. *Journal of Geochemical Exploration.* 145. 12-24.

James, G., Witten, D., Hastie, T., Tibshirani, R. (2013). An introduction to statistical learning: with applications in R. *Springer*.

MinAnalytical Laboratory Services Australia Pty Ltd. (2018). *Schedule of Analytical Services & Guide Book.* URL: https://minanalytical.com.au/wp-content/uploads/sites/7/2019/07/MINA_Services_Book_Aug2018.pdf

Polyak, B, T., Khlebnikov, M, V. (2017). Robust Principal Component Analysis: An IRLS Approach. *IFAC-PapersOnLine*, 50, 2762-2767.

RStudio Team (2019). RStudio: Integrated Development of R. RStudio, Inc., Boston, MA URL http://www.rstudio.com/

Appendix

```
# Assessment 3 - Brendan Garner

# Load the required packages
library(dplyr)
library(mvoutlier)
library(cluster)
library(spdep)
library(ggplot2)
library(caret, warn.conflicts = F, quietly = T)

# Import the CSV into an R data frame
Data <- read.csv("D:\\Master of Data Science\\Introduction to Data
Mining\\Assessments\\Assessment 3\\AC ME cleaned.csv", header = TRUE,
            sep = ",", dec = ".")

# Select only the columns with assay data
ME_Data <- Data[,16:26]

# Principal Components Analysis

# Apply multivariate outlier detection
res <- mvoutlier.CoDa(ME_Data)

# Plot the two largest principal components with outliers
plot(res, which = 'biplot', onlyout = TRUE, symb = TRUE, symbtxt = FALSE)

plot(res, which = 'map', coord = Data[9:10], onlyout = FALSE, symb = TRUE,
    symbtxt = FALSE)

# Get the clr transformed PCA loadings for two largest principal components
loadings <- data.frame(res$pcaobj$princompOutputClr$loadings[,1:2])

# Get the clr transformed PCA scores for two largest principal components
scores <- data.frame(res$pcaobj$princompOutputClr$scores[,1:2])

# Get the eigenvalues for plotting a scree chart
eigenvalues <- unlist(res$pcaobj[2])
eigenvalue_variance <- (100 * eigenvalues / sum(eigenvalues))
eigenvalue_variance_DF <- data.frame(Dimensions=1:length(eigenvalue_variance),
                          eigenvalue_variance = eigenvalue_variance)

# Display scree plot of the principal components
ggplot(eigenvalue_variance_DF,aes(x=Dimensions,y=eigenvalue_variance)) + geom_point() +
  geom_col(fill="steelblue") + geom_line() +
  theme_bw() + scale_x_continuous(breaks=1:nrow(eigenvalue_variance_DF)) +
  ggtitle ("Proportion of variance explained by each principal component") +
  theme(plot.title = element_text(hjust = 0.5)) +
  ylim(c(0,100)) + ylab("Explained variance %")
```

```
# Get the values for outliers and colours from the res object
# Get the desurveyed bottom of hole coordinates from the Data object
# Get ilr transformed values of gold and strongly correlated elements
sampleIDs <- Data[,13]
outliers <- res$outliers  # True or False
colours <- res$colcol  # Red is larger
XYZ <- Data[9:11]
ilr_variables <- data.frame(res$ilrvariables)

# Make data frame from outliers, colours and desurveyed coordinates
outliers_df <- data.frame(sampleIDs, XYZ, ilr_variables[,c(1,4,9)], scores, outliers, colours)

# Identify rows where outliers == TRUE
outliers_df <- filter(outliers_df, outliers == TRUE)

# Only keep rows where PC1 >= 0
outliers_df <- filter(outliers_df, Comp.1 >0)

# Export outliers_df to csv
# write.csv(outliers_df,"F:\\Master of Data Science\\Introduction to Data
Mining\\Assessments\\Assessment 3\\outliers.csv", row.names = TRUE)

# Clustering
# Convert XY coordinates from outliers_df into a spatial object
XY <- coordinates(outliers_df[2:3])

# Create neighbour list
neighbours <- tri2nb(XY)
plot(neighbours, coordinates(XY))

# Calculate costs
ilr_Au_Bi_Te <- data.frame(outliers_df[5], outliers_df[6], outliers_df[7])
link_costs <- nbcosts(neighbours, ilr_Au_Bi_Te)

# Make list of link weights
link_weights <- nb2listw(neighbours, link_costs, style="B")

# Find a minimum spanning tree
MST <- mstree(link_weights,5)

# The MST plot
par(mar=c(0,0,0,0))
plot(MST, coordinates(XY), col=2,
    cex.lab=.7, cex.circles=0.035, fg="blue")
plot(XY, border=gray(.5), add=TRUE)

# Apply SKATER to prune the minimum spanning tree
ME_clusters <- skater(MST[,1:2], ilr_Au_Bi_Te, 9)

# The SKATER plot
par(mar=c(0,0,0,0))
```

```
plot(ME_clusters, coordinates(XY), cex.circles=0.035, cex.lab=.7)

# Add the clustered group ID to the outliers_df
outliers_df <- cbind(outliers_df, ME_clusters$groups)

# Export outliers_df to csv
write.csv(outliers_df,"F:\\Master of Data Science\\Introduction to Data
Mining\\Assessments\\Assessment 3\\outliers.csv", row.names = TRUE)

# Perform Naive Bayes Classification
# Test for data normalcy after ilr transformation
ilr_variables_stacked <- stack(ilr_variables)

ggplot(ilr_variables_stacked, aes(sample = values)) +
  ggtitle ("Q-Q plots for isometric logratio transformed values") +
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_qq(distribution = qnorm) +
  geom_qq_line(line.p = c(0.25, 0.75), col = "blue") +
  facet_wrap( ~ ind)

# Create the binary factor response variable
Data_response <- mutate(Data, Au_ppm = ifelse(Au_ppm >= 0.001, 1, 0))
response <- as.factor(Data_response[,16])

# Replace the ilr transformed gold value with the response variable
Data_NB <- ilr_variables[,-1]
Data_NB$Au_target <- response
Data_NB <- Data_NB[,c(ncol(Data_NB),1:(ncol(Data_NB)-1))] # Move from last to 1st column

# Split Data into test and training data sets
set.seed(1234)
split <- createDataPartition(Data_NB[,1], p = 0.8, list = FALSE)
train <- Data_NB[split, ]
test <- Data_NB[-split, ]

# Count number of rows in train and test splits
c(nrow(train), nrow(test))

# Split the training dataset into response and predictors variables
response <- train[,1]
predictors <- train[,-1]

# Train the model
train_control <- trainControl(method = "cv", number = 10)
gold_NB <- train(x = predictors, y = response, method = "nb",
                 trControl = train_control)

# Review results of model performance on training data using a confusion matrix
confusionMatrix(gold_NB)

# Make predictions on the test data set
```

```
prediction <- predict(gold_NB, newdata = test)

# Review results of model performance on test data using a confusion matrix
confusionMatrix(prediction, test[,1])

# Create a data frame to compare each test result
Au_target <- test[,1]
NB_results <- Data[-split, ]
NB_results$Au_target <- Au_target
NB_results$prediction <- prediction

# Identify the false positives for further spatial analysis
False_positve <- mutate(NB_results, FP = ifelse(Au_target == 0 &
                                    prediction == 1, 1, ""))

# Export false positives to CSV for import to ArcGIS
write.csv(False_positve,"D:\\Master of Data Science\\Introduction to Data
Mining\\Assessments\\Assessment 3\\false_positives.csv", row.names = TRUE)
```