Datarock

Data Scientist - Geoscience in the Datarock Applied Science team – Coding Challenge

By: Stephanie Walmsley

Overview of the Solution Approach

1. Exploratory Data Analysis (EDA)
2. QA/QC
3. Create Training Features
4. Model Training
5. Feature Importance Assessment
6. Predict Labels onto Unlabelled Data
7. Conclusion
8. Next Steps

1. Exporatory Data Anaylsis

Understand the label split, *note class imbalance*

	label	Count	Percentage
0	proximal	2787	60.025845
1	distal	1118	24.079259
2	?	738	15.894896

```
ID          object
holeid      object
from         int64
to         float64
As         float64
Au          object
Pb         float64
Fe         float64
Mo         float64
Cu         float64
S          float64
Zn         float64
label       object
dtype: object
```

Check dtypes:
Noticed Au was an object so investigated further & found <LDL → fixed by changing value to ½ of <LDL to get id of <

Summary Stats:

	from	to	As	Au	Pb	Fe	Mo	Cu	S	Zn
count	4771.000000	4771.000000	3268.000000	4765.000000	4756.000000	4709.000000	4741.000000	4746.000000	4761.000000	4762.000000
mean	750.379585	760.353574	19.730855	0.051956	689.831232	49952.514598	9.991452	12.450601	9750.033213	59.389636
std	447.126995	447.114592	37.181529	0.089862	1047.642566	21490.606419	87.098943	107.438873	15557.657335	120.489477
min	71.000000	81.000000	1.000000	0.002500	1.600000	2080.000000	-999.000000	1.000000	26.000000	5.600000
25%	421.000000	431.000000	5.400000	0.010000	132.200000	39260.000000	1.400000	3.000000	1338.000000	29.800000
50%	641.000000	651.000000	9.200000	0.027000	396.700000	49020.000000	4.400000	4.600000	3636.000000	38.200000
75%	991.000000	1001.000000	20.000000	0.061000	940.200000	58420.000000	17.400000	8.000000	10988.000000	52.600000
max	2201.000000	2211.000000	827.800000	1.878000	29793.800000	397000.000000	1939.400000	6767.000000	217600.000000	3455.000000

2. QA/QC

Interval lengths:

```
interval_length
10.00     4723
6.00         3
9.50         3
7.50         2
9.40         2
5.30         2
6.60         2
5.40         2
9.90         2
8.60         2
5.70         2
9.30         2
6.70         1
8.32         1
6.30         1
6.50         1
9.20         1
6.40         1
8.40         1
5.10         1
9.70         1
5.43         1
8.20         1
7.40         1
6.75         1
5.80         1
7.40         1
5.20         1
8.60         1
8.70         1
7.70         1
7.20         1
5.60         1
8.00         1
7.60         1
7.80         1
Name: count, dtype: int64
```

None are extremely different from the standard/majority (10), so would assume these are fine

Check number of samples per unique hold ID:

	holeid	Count
0	SOLVE236	164
1	SOLVE279A	163
2	SOLVE237	154
3	SOLVE197	141
4	SOLVE196	136
5	SOLVE198	129
6	SOLVE127	126
7	SOLVE064	119
8	SOLVE195W3	105
9	SOLVE080	103
10	SOLVE291	96
11	SOLVE040	86
12	SOLVE045	85
13	SOLVE179	84
14	SOLVE177	79
15	SOLVE040W1	77
16	SOLVE044	77
17	SOLVE161	76
18	SOLVE143	75
19	SOLVE145	73
20	SOLVE147	71
21	SOLVE176	70
22	SOLVE146	69
23	SOLVE195W1	69
24	SOLVE225	69

Top 25 largest hole Ids by number of samples (between 164 – 69 samples per hole)

	holeid	Count
115	SOLVE234	6
116	SOLVE104	6
117	SOLVE108	6
118	SOLVE181W1	6
119	SOLVE099	6
120	SOLVE195W2	6
121	SOLVE201	5
122	SOLVE003	5
123	SOLVE015	5
124	SOLVE010	4
125	SOLVE133	4
126	SOLVE046	4
127	SOLVE149W1	4
128	SOLVE169	4
129	SOLVE218	4
130	SOLVE090	4
131	SOLVE122	3
132	SOLVE132	3
133	SOLVE124	3
134	SOLVE080W1	2
135	SOLVE055	2
136	SOLVE199	2
137	SOLVE057	1
138	SOLVE047	1
139	SOLVE206	1

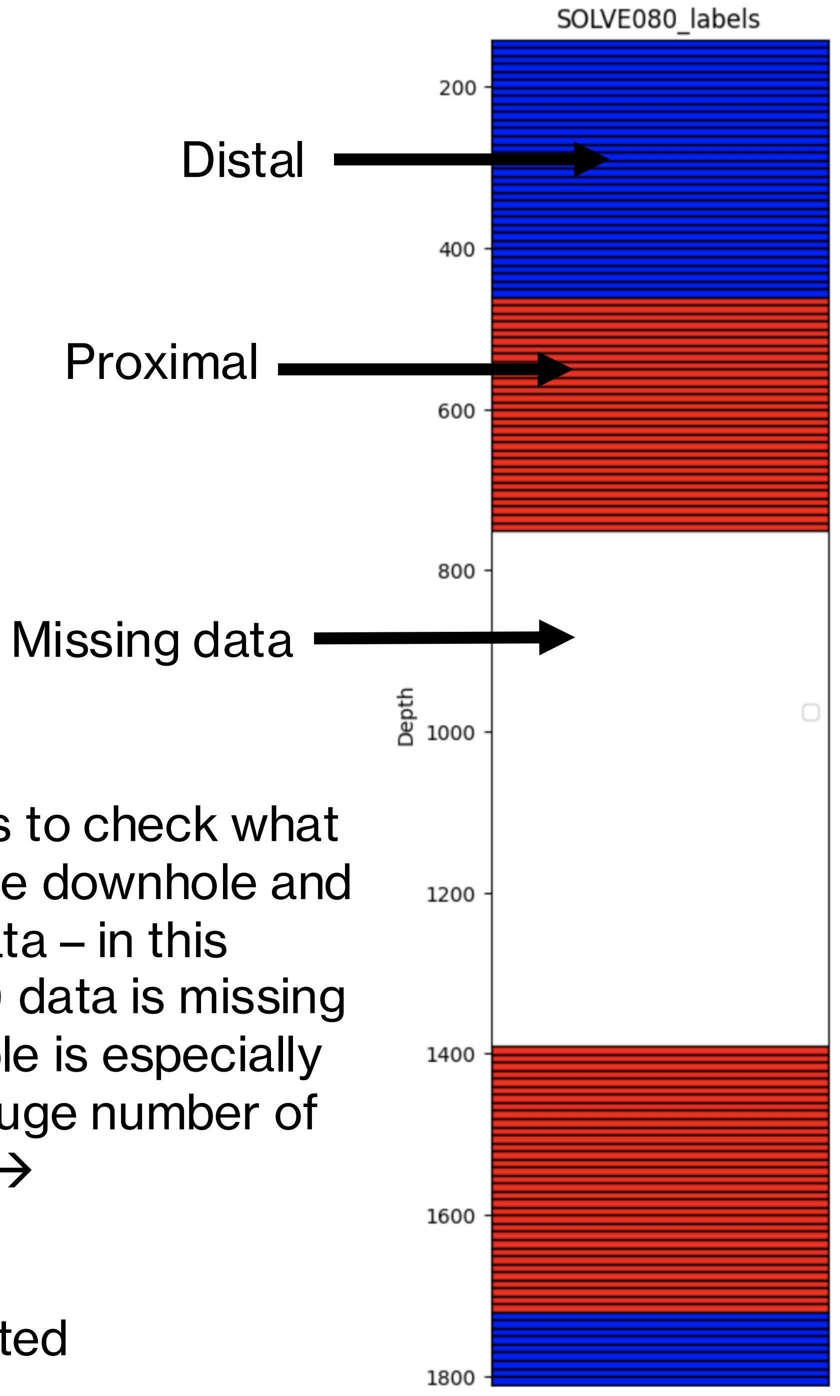
Bottom 25 smallest hole IDs by number of samples (between 1 – 6 samples per hole)

*** these lower samples seem concerning – I would check for missing samples** But I included them due to time constraints of this task*

2. QA/QC Cont'd

holeid	Unique_Count	Unique_Labels
SOLVE003	1	[proximal]
SOLVE004	1	[proximal]
SOLVE007	1	[proximal]
SOLVE008	2	[distal, proximal]
SOLVE010	1	[proximal]
SOLVE011	1	[proximal]
SOLVE015	1	[proximal]
SOLVE016	1	[proximal]
SOLVE017	1	[proximal]
SOLVE021	1	[proximal]
SOLVE025	1	[proximal]
SOLVE026	2	[distal, proximal]
SOLVE027	2	[distal, proximal]
SOLVE028	1	[proximal]
SOLVE030	2	[distal, proximal]
SOLVE031	1	[proximal]
SOLVE036	1	[proximal]
SOLVE037	2	[distal, proximal]
SOLVE039	2	[distal, proximal]
SOLVE040	2	[distal, proximal]
SOLVE040W1	2	[distal, proximal]
SOLVE041	2	[distal, proximal]
SOLVE042	1	[proximal]
SOLVE043	2	[distal, proximal]
SOLVE044	2	[distal, proximal]
SOLVE045	2	[distal, proximal]
SOLVE046	1	[proximal]
SOLVE047	1	[proximal]
SOLVE048	1	[proximal]
SOLVE052	1	[proximal]
SOLVE055	1	[proximal]
SOLVE056	1	[proximal]
SOLVE057	1	[proximal]
SOLVE064	2	[proximal, distal]
SOLVE066	1	[proximal]
SOLVE067	1	[proximal]
SOLVE068	1	[proximal]
SOLVE069	1	[proximal]

←Checked to see what the labels per holeid looked like (would expect both proximal & distal) but looking at the first 50 results show that a lot of holes have only proximal which doesn't make sense unless only the proximal zone has been sampled – *suggests missing data that would be useful to have for this study or that samples have been mislabelled*



Plotted a few holeid to-from's to check what the labels and data looked like downhole and found evidence of missing data – in this example (holeid: SOLVE080) data is missing from 751 to 1391 - this example is especially concerning as its missing a huge number of samples from the ore zone →

** would look at all the holes in detail if time permitted

2. QA/QC Cont'd

- Check for Duplicates in the sample Id's & for intervals within unique sampleIds → *None found*
- Already dealt with lower detection limit issues for Au: fixed by changing value to ½ of <LDL to get id of <
  - *Would have been ideal to have the analytic LDL for all elements and then would take ½ of any at the reported analytical LDL*

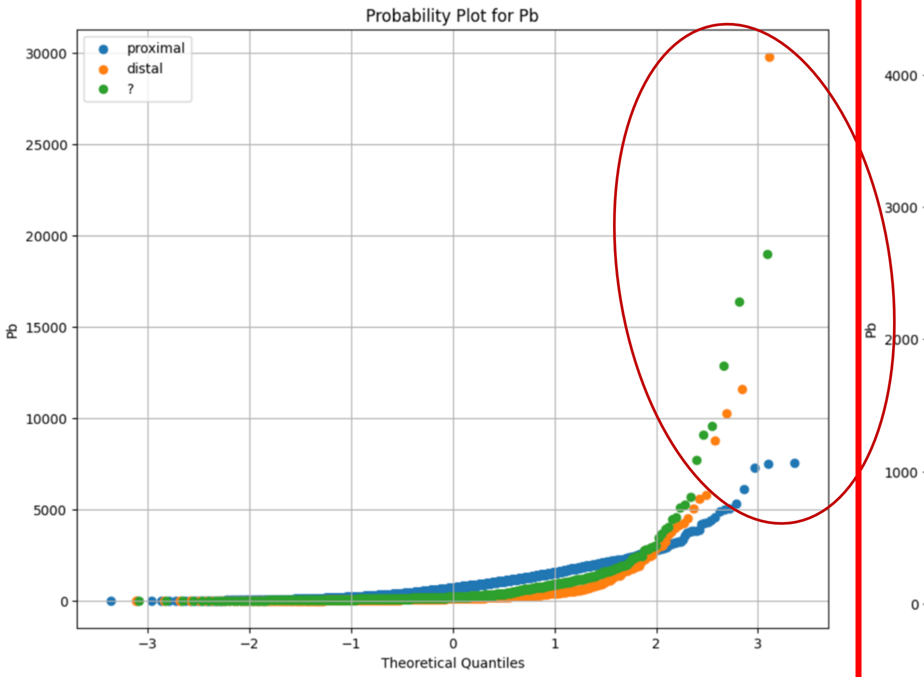
NaNs:

id	holeid	from	to	As	Au	Pb	Fe	Mo	Cu	S	Zn
A04812	SOLVE003	561	571.0	NaN	0.066000	1031.00	61380.0	138.2000	3.600	3586.0000	43.6000
A03356	SOLVE003	571	581.0	NaN	0.152000	1982.00	50860.0	75.4000	4.800	1822.0000	36.4000
A04764	SOLVE003	581	591.0	NaN	0.068000	1064.80	57940.0	29.2000	3.000	740.4000	36.6000
A04626	SOLVE003	591	601.0	NaN	0.074000	891.60	48620.0	63.0000	4.200	820.8000	39.6000
A05579	SOLVE003	601	611.0	NaN	0.043125	801.25	51025.0	56.0625	4.875	745.6875	32.3125
...	...	...	...	...	...	...	...	...	...	...	...
A04915	SOLVE291	1291	1301.0	12.2	0.064000	208.20	51500.0	2.6000	3.000	4200.0000	29.4000
A06596	SOLVE291	1301	1311.0	10.4	0.024000	145.40	55040.0	2.6000	3.000	6160.0000	34.6000
A07560	SOLVE291	1311	1321.0	10.0	0.011000	109.60	55460.0	2.6000	4.000	5700.0000	33.4000
A07802	SOLVE291	1321	1331.0	5.0	0.009000	69.20	42520.0	2.4000	3.000	3140.0000	29.6000
NaN_count	0	0	0.0	1503.0	6.000000	15.00	62.0	30.0000	25.000	10.0000	9.0000

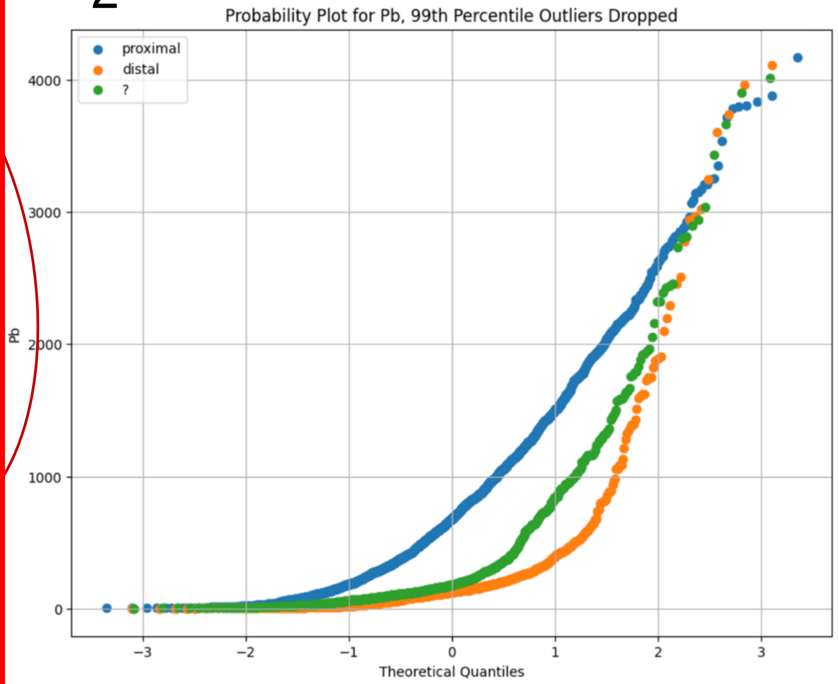
Counted NaN's per column and found As was missing 1503 values. Decided to throw out As and keep the others. If time permitted I coud have tried some imputation methods for filling in the NaNs.

2. QAQC Cont’d: Upper Outlier Detection

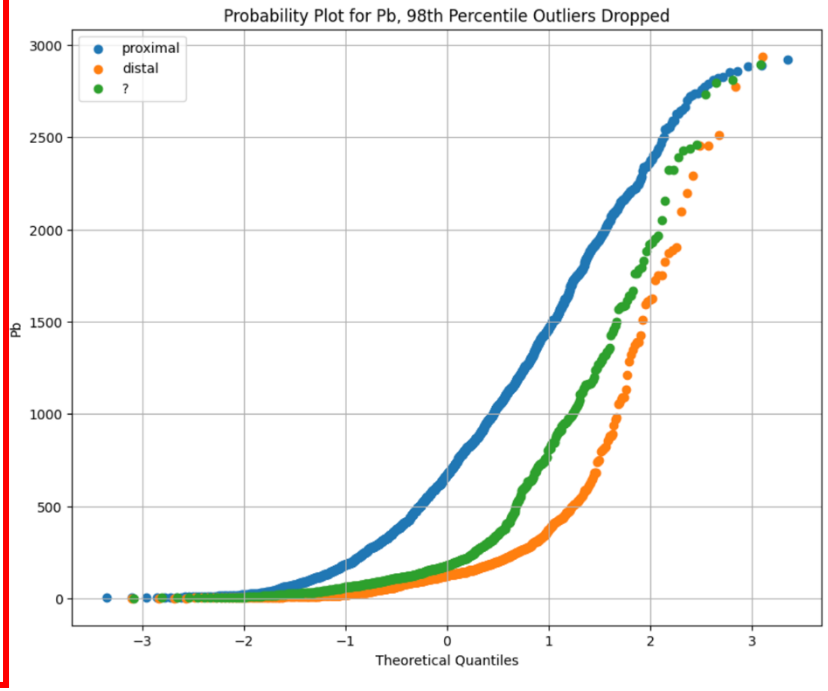
1



2



3



Looking at Probability Plots to decide on upper limit outliers. The distal group (orange) has one very distinct outlier in Plot 1 and some questionable values in the other 2 groups, possibly an UDL for the top 3 proximal points (blue). I have compared the plots for removing the 98% and 99% outliers and have chosen to use the 99th percentile cut off because it deals with the extreme outliers, whereas 98% removes too many high values.

Next step:

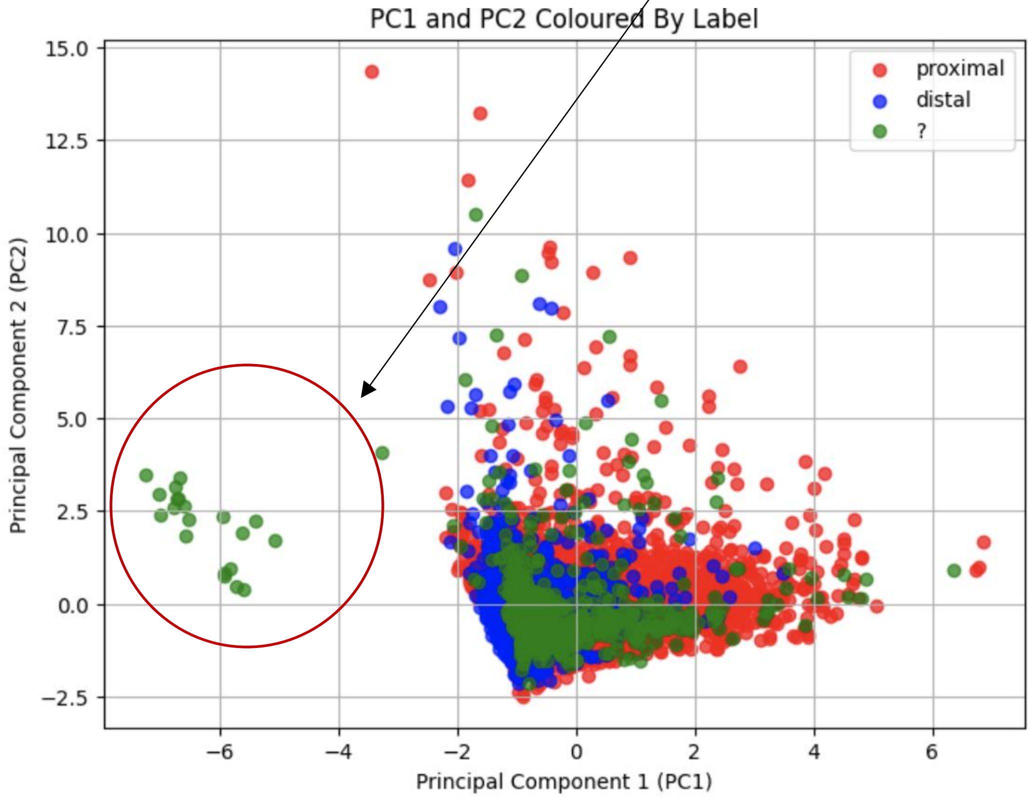
If I had more time I would consider assigning these outliers a upper limit value so that we did not loose them and we could ensure they didn’t mess up our stats moving forward. Would ideally know the analytical method upper limit to make these decisions.

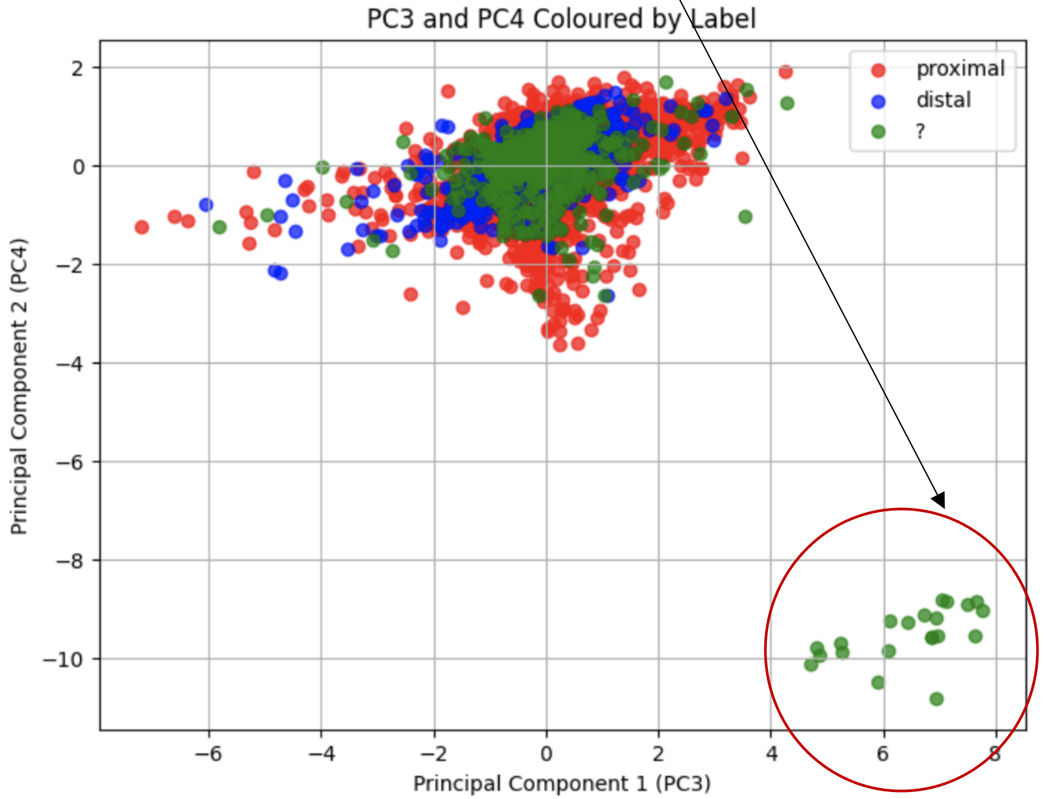
I would also do this process with every element, but again due to the limited time frame I will just show this example (Pb) and use the 99th percentile for all the elements.

3. Features - PCA

- Ran a PCA Analysis on the 7 elements (having excluded As)
- Included PC1-PC5 as training features

- Interesting separate population showing up in the unlaballed group, *suggests the prediction task might not work well if the unlabelled data is not from the same population as the labelled data*
- *A lot of overlap between the proximal and distal populations which may suggest that the elements we have will not do well at classifying the 2 labels.*






```
Explained Variance: PC1 = 0.23, PC2 = 0.23, PC3 = 0.17, PC4 = 0.13, PC5 = 0.10, PC6 = 0.07, PC7 = 0.06
```

Chose to include first 5 components as the explained variance becomes insignificant(less then .10) past PC5

3. Features

Z-scored the elements to use as features (in order to normalize the data):

	Au_z	Pb_z	Fe_z	Mo_z	Cu_z	S_z	Zn_z
0	2.606355	2.566352	0.270988	1.053457	-0.226598	-0.563341	-0.296302
1	0.672065	0.920221	0.801729	0.329657	-0.524861	-0.670266	-0.286839
2	0.810229	0.609373	0.103070	0.859191	-0.326019	-0.662318	-0.144904
3	0.099262	0.447219	0.283357	0.750503	-0.214170	-0.669744	-0.489688
4	0.303629	1.474436	-0.238763	0.545857	-0.524861	-0.579198	-0.882966

Created a number of Element Ratios to include as features (would have experimented with more ratios if time permitted):

id	Pb_Cu_Ratio	Pb_Zn_Ratio	Pb_Mo_Ratio	Pb_S_Ratio	Pb_Fe_Ratio	Cu_Mo_Ratio	Zn_Cu_Ratio	Cu_Au_Ratio	Zn_Mo_Ratio
A03356	412.916667	54.450549	26.286472	1.087816	0.038970	0.063660	7.583333	31.578947	0.482759
A04764	354.933333	29.092896	36.465753	1.438142	0.018378	0.102740	12.200000	44.117647	1.253425
A04626	212.285714	22.515152	14.152381	1.086257	0.018338	0.066667	9.428571	56.756757	0.628571
A05579	164.358974	24.796905	14.292085	1.074512	0.015703	0.086957	6.628205	113.043478	0.576366
A05248	457.866667	57.233333	31.944186	0.826673	0.031176	0.069767	8.000000	57.692308	0.558140

Final features included PC1-PC5, the 7 elements z-scores & 9 element ratios = 21 Features

4. Model Training

- Ran a few models to see which worked best for the classification task (predicting proximal vs. distal)
- chose a test/train split of 30/70

```
Model Accuracies:
Logistic Regression: 0.80
Random Forest: 0.84
SVM: 0.82
KNN: 0.81
Gradient Boosting: 0.82

Best Model: Random Forest with accuracy 0.84
```

Classifcation report and confusion matrix for the best (highest accuracy) model: **Random Forest**

```
Classification Report:
              precision    recall  f1-score   support

      distal       0.80      0.64      0.72       349
    proximal       0.85      0.93      0.89       747

    accuracy                           0.84      1096
   macro avg       0.83      0.79      0.80      1096
weighted avg       0.83      0.84      0.83      1096


Confusion Matrix:
[[225 124]
 [ 55 692]]
```

Note: class imbalance – proximal has a lot more labels then distal and so the model does a better job of predicting proximal compared to distal

Could have done hyperparameter tuning with more time

5. Feature Assessment

Feature importance for Random Forest Model

	Feature	Importance
1	Pb_Zn_Ratio	0.106538
4	Pb_Fe_Ratio	0.089718
15	Pb_z	0.081769
8	Zn_Mo_Ratio	0.075802
9	PC1	0.055381
0	Pb_Cu_Ratio	0.048444
13	PC5	0.044436
3	Pb_S_Ratio	0.044252
11	PC3	0.044158
19	S_z	0.043956
5	Cu_Mo_Ratio	0.042490
17	Mo_z	0.040232
12	PC4	0.035840
2	Pb_Mo_Ratio	0.035023
7	Cu_Au_Ratio	0.033542
16	Fe_z	0.032803
6	Zn_Cu_Ratio	0.031515
20	Zn_z	0.030579
10	PC2	0.030190
14	Au_z	0.028815
18	Cu_z	0.024516

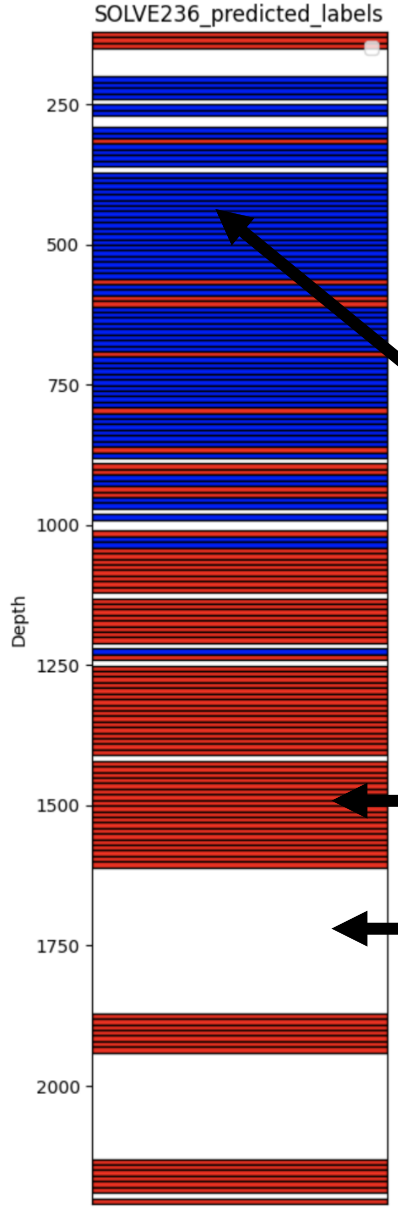
It appears that all feature classes were useful as all 3 types (ratios, z-scores & PC's) appear in the top 5

* Would look at other librarys/ methods for explainability if I had more time

6. Predict Labels on Unlabelled Data using the Chosen Model

```
predicted_label
distal      339
proximal    334
Name: count, dtype: int64
```

predicted_label	probability_proximal
proximal	0.99
proximal	1.00
proximal	1.00
proximal	0.95
proximal	1.00
...	...
proximal	0.82
proximal	0.79
proximal	0.75
proximal	0.76
proximal	0.84



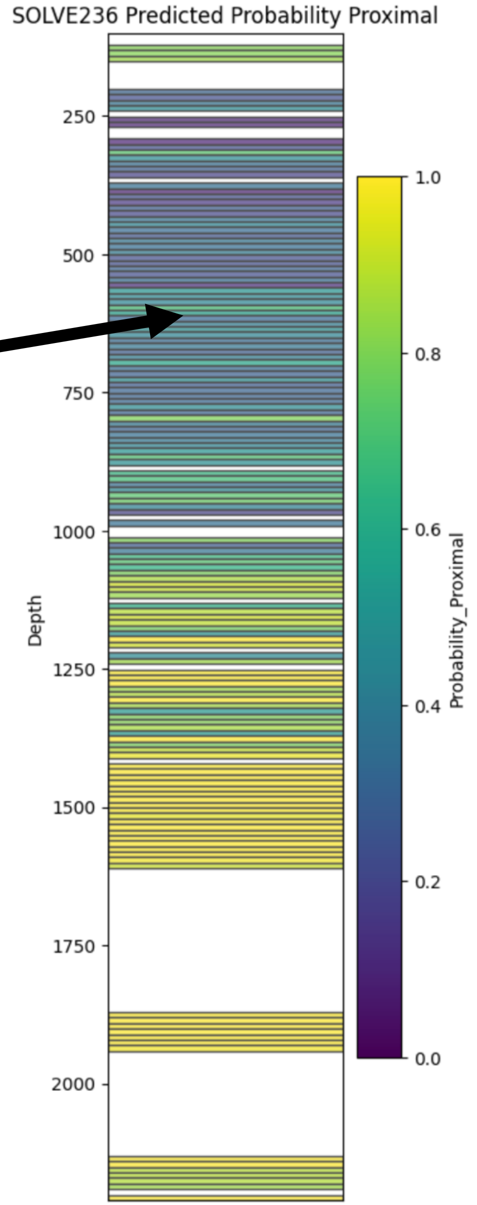
Example of an unlabelled drill hole (SOLVE326) with the predicted labels & predicted probability of “proximal” applied

Distal

Proximal

Missing data

A lot of uncertainty



7. Conclusion

- ***Can we use the same geochemical data and labels to generate a predictive model for future drill holes which can label samples on whether they are in class A or class B?***
  - Yes, however:
    - Need to address the class imbalance – need more distal labels, and/or need to weight labels differently (more weight on the distal)
    - Model predicts the labels well (F1 of 72% distal, 89% proximal) so it could work for a similar deposit, similar Geochem
- ***More data has been acquired since the geochemist completed her work - can we predict labels onto these data points (labelled "?")***
  - PCA discovered a unique population in the unlabelled data – suggests some of this data would not be predicted well
  - Based on the example drillhole prediction viz the unlabelled data can be predicted generally, but errors will definetly occur espiecally near proximal/distal boundaries
- ***Potential issues or pitfalls with the approach:***
  - Grouping background/ unmineralized zones into distal --> would be ideal to break out these classifcations further
  - Lots of zones will blend into eachother, having only 2 categories doesn't really make sense geologically
  - Lots of missing data thoughout the drill holes, drill holes containing only proximal labels (where is the distal? Missing or mislabelled?)
  - Concerns with overlapping populations in the PCA's – would investigate further

Next Steps:

- Find missing and/or mislabelled data
- Look at clustering the proximal and distal labels seperately to see if they can be broken down further for more fine grained vectoring/classification potential
  - Look at PCAs in more detail
- Near miss-modeling checking that DHs go through Distal-Proximal-Distal (checking if any only hit Distal or didn't make it all the way through the orebody)
- Visualizing the ratios (XY plots) to check which ones are making the best predictions/ validate the top feature importance
- Would always want to spatially check the data ** out of scope/ no coordinates provided
- Would suggest assaying for additional elements such as: Ag, Sb, Cd, Mn, Bi,