Genetic Variant Classification

●●●

Predicting whether a variant will have conflicting clinical classifications

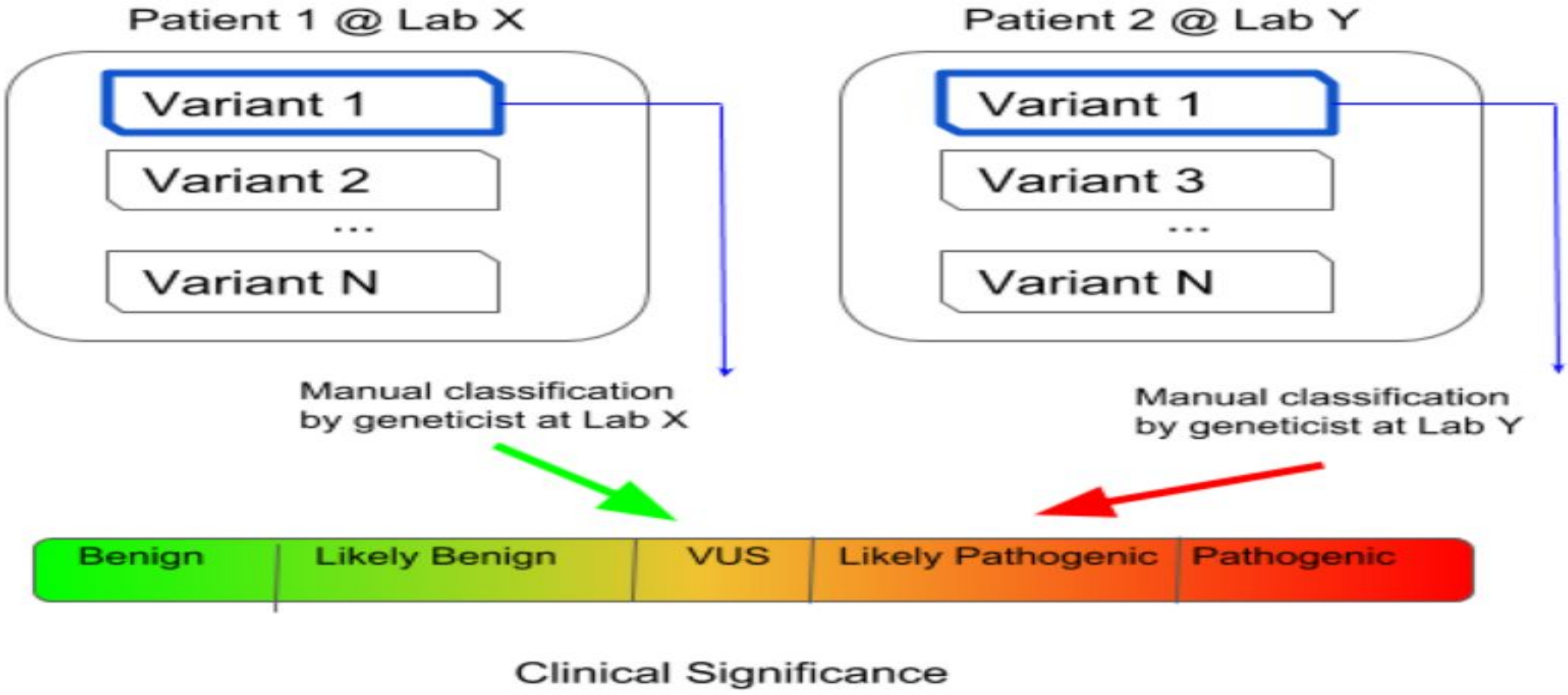
Summary

- Introduction
- Data
- Preprocessing
- Methods and Processes
- Results and comparison
- Conclusion

Introduction

- Genetic variants is an alteration in the DNA sequence.
- These variants are classified by clinical laboratories into different categories: benign, likely benign, uncertain significance, likely pathogenic, and pathogenic. From laboratory to laboratory this variant classification is not consistent which means that a laboratory A can consider a given variant as likely benign whereas a laboratory B can consider it likely pathogenic.
- The goal of this project is to predict whether a variants will have conflicting clinical classification.

Conflicting Variant Classification - **Class: 1**



Patient 1 @ Lab X

Variant 1

Variant 2

...

Variant N

Patient 2 @ Lab Y

Variant 1

Variant 3

...

Variant N

Manual classification
by geneticist at Lab X

Manual classification
by geneticist at Lab Y

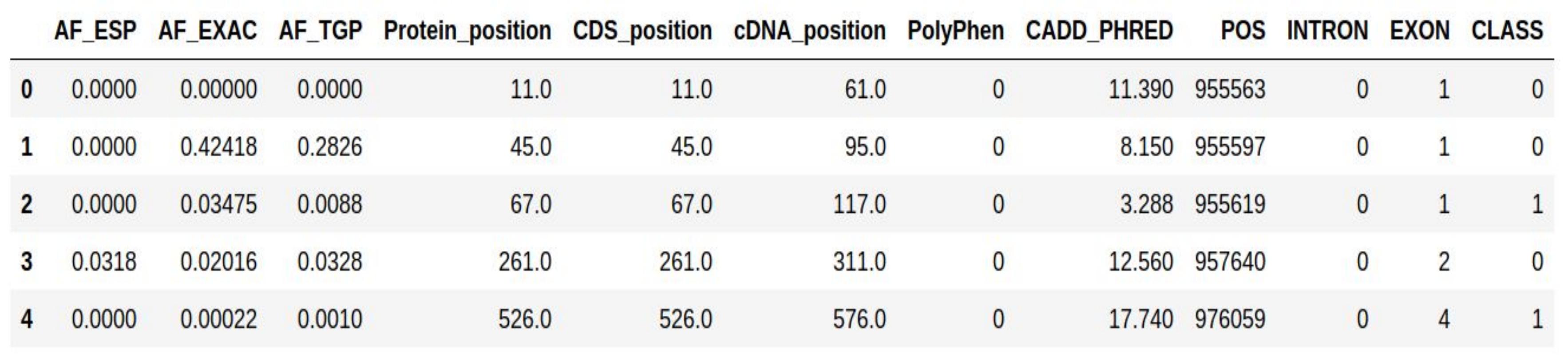
Benign | Likely Benign | VUS | Likely Pathogenic | Pathogenic

Clinical Significance

Data

The source of the data I used in this project is ClinVar platform. The data is also published on kaggle platform.

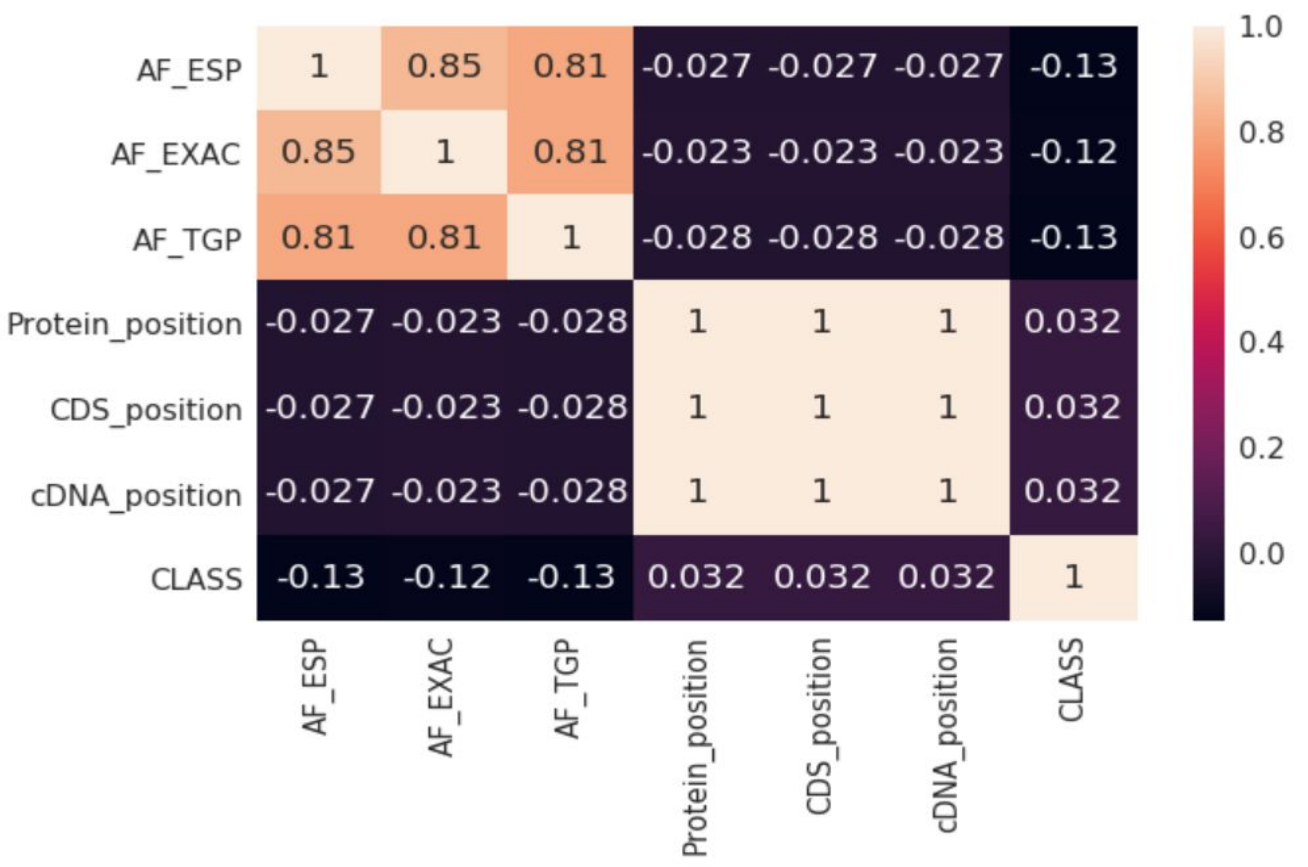
The Data set contains over 65,000 rows and 46 columns.



	AF_ESP	AF_EXAC	AF_TGP	Protein_position	CDS_position	cDNA_position	PolyPhen	CADD_PHRED	POS	INTRON	EXON	CLASS
0	0.0000	0.00000	0.0000	11.0	11.0	61.0	0	11.390	955563	0	1	0
1	0.0000	0.42418	0.2826	45.0	45.0	95.0	0	8.150	955597	0	1	0
2	0.0000	0.03475	0.0088	67.0	67.0	117.0	0	3.288	955619	0	1	1
3	0.0318	0.02016	0.0328	261.0	261.0	311.0	0	12.560	957640	0	2	0
4	0.0000	0.00022	0.0010	526.0	526.0	576.0	0	17.740	976059	0	4	1

Preprocessing

- I dropped the columns contained over 90% of NAs.
- For the position columns, I replaced the null values with the mean of the adjacent values.
- I parsed the Intron and Exon columns to extract their length and position.
- I dropped some highly correlated features.
- I mapped the ordinal features
- I created dummy variables for the nominal variables



	AF_ESP	AF_EXAC	AF_TGP	Protein_position	CDS_position	cDNA_position	CLASS
AF_ESP	1	0.85	0.81	-0.027	-0.027	-0.027	-0.13
AF_EXAC	0.85	1	0.81	-0.023	-0.023	-0.023	-0.12
AF_TGP	0.81	0.81	1	-0.028	-0.028	-0.028	-0.13
Protein_position	-0.027	-0.023	-0.028	1	1	1	0.032
CDS_position	-0.027	-0.023	-0.028	1	1	1	0.032
cDNA_position	-0.027	-0.023	-0.028	1	1	1	0.032
CLASS	-0.13	-0.12	-0.13	0.032	0.032	0.032	1

Methods and Processes

A) For feature selection I used:

- Random Forest

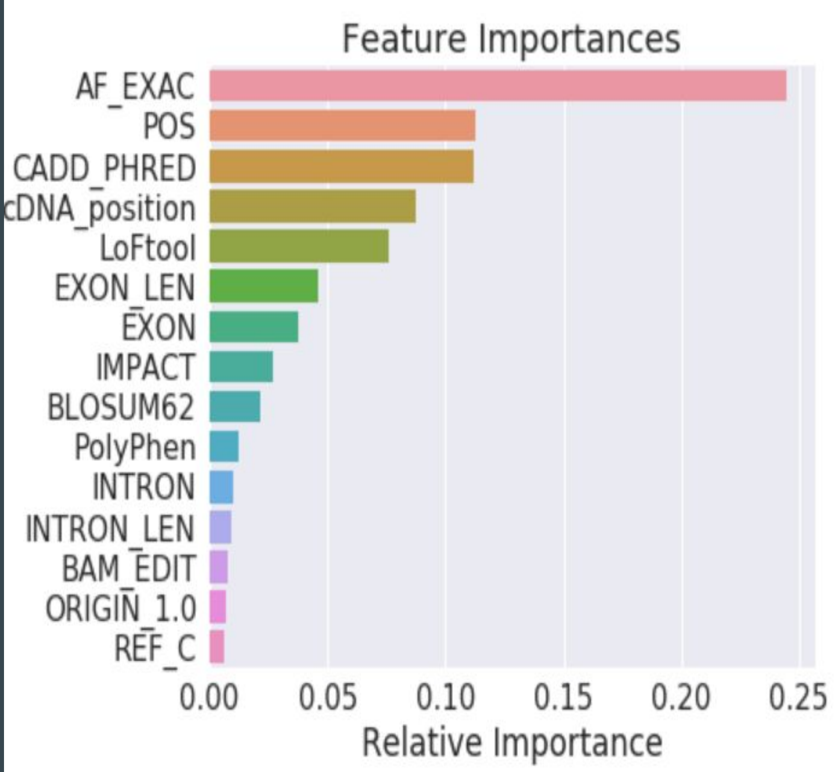
B) For prediction I used:

- Random Forest
- XGBoost
- Logistic Regression
- SGDClassifier
- Gaussian Naive Bayes

Results and Comparison

A) Feature selection

- Allele Frequency from ExAC is the most important feature.
- I selected a subset of 65 variables for predictions.

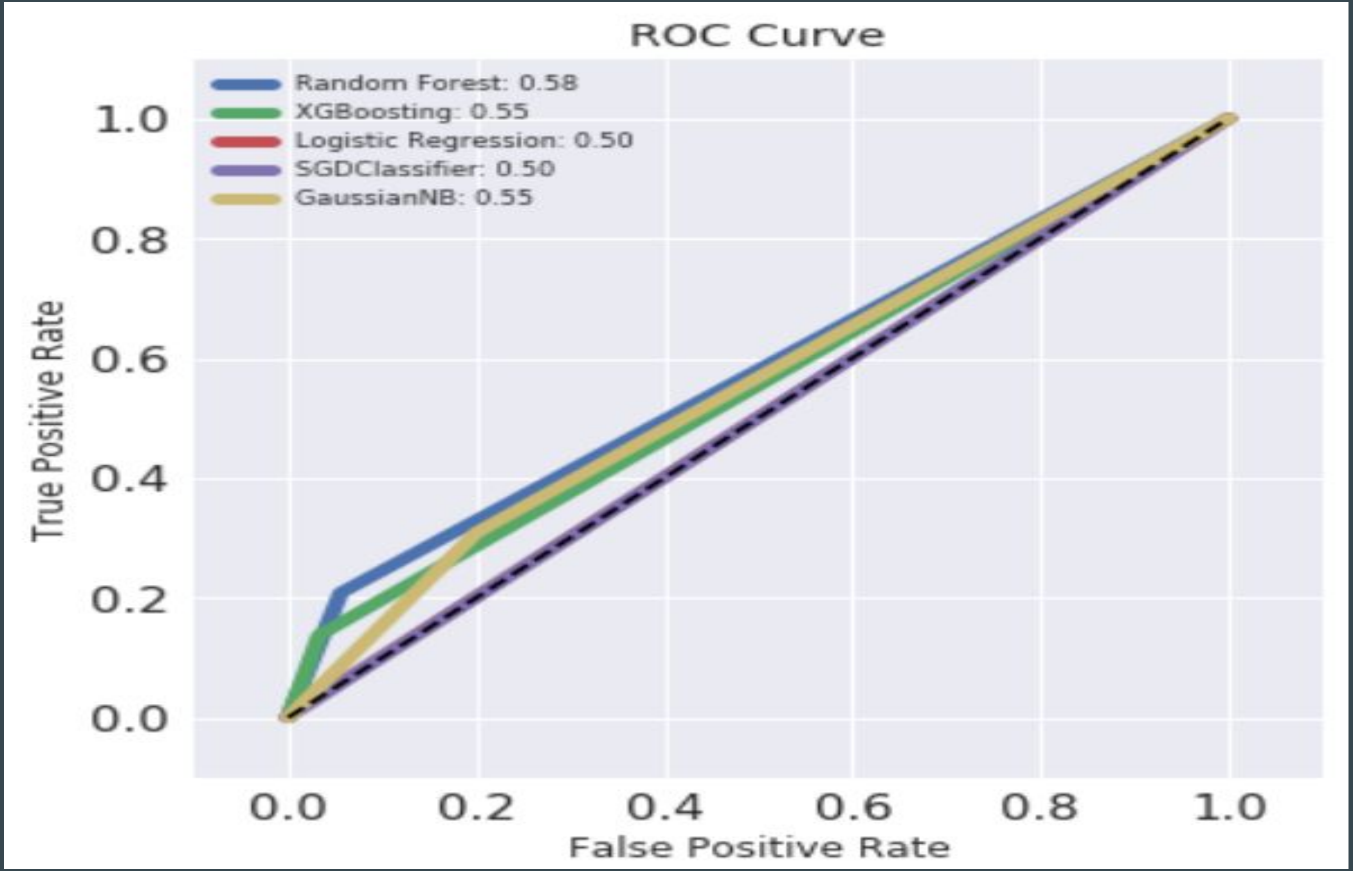


Results and Comparison

B) predictions

Model	XGBoost	Random Forest	Logistic Regression	GaussianNB	SGDClassifier
Accuracy	0.7565	0.7574	0.7461	0.675	0.7459
ROC curve	0.55	0.58	0.5	0.55	0.5

Random Forest classifier outperformed the other models in terms of accuracy, and ROC curve.


ROC Curve
Random Forest: 0.58
XGBoosting: 0.55
Logistic Regression: 0.50
SGDClassifier: 0.50
GaussianNB: 0.55
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0

Conclusion

- Allele frequency, variant position in the chromosome, and deleteriousness score of the SNV(single nucleotide variant) are the most important features.
- The models in this project didn’t perform well, but I believe with a more sophisticated models and more relevant features the performance can be improved substantially. Therefore, there is a possibility to solve this problem with machine learning algorithms.

Questions???