Classification with Drug Consumption

Tao Shan, Dan Chen, Zhengqi Sun,
Janny Liu, Ziwei Pan, Xinwei Lyu

https://archive.ics.uci.edu/ml/datasets/Drug+consumption+%28quantified%29

Schedule

01 Motivation

02 Data Exploration

03 Feature Selection

04 Model Prediction

05 Discussion & Conclusion

01 Motivation

- Drug consumption
  Biological measurement: NEO-FFI-R, BIS-11, ImpSS
  basic information: Level of education, age, sex
- Classification problem
  Predict when people uses Cannabis within one year
- Why prediction matters?

Introduction to the data set

- Shape: (1885, 32)
- Attributes description:
  - Basic information: column 2-6
  - Biological measurement: column 7-13
  - Legal and illegal drugs: column 14-32
- Type of data
- Target variable (CL0 - CL6)

Data Cleaning

a **Check if there is any missing values**
Using the sum(is.na()) function in R

b **Change the elements in the data to meaningful values**
It can help us to understand the data better

c **Adding the column name to the data**
Since it can help us to select whole column of data

d **Transform the data to suit our models**
All the models require data to be digital, change some of our data to fit

02 Data Exploration

- Correlation

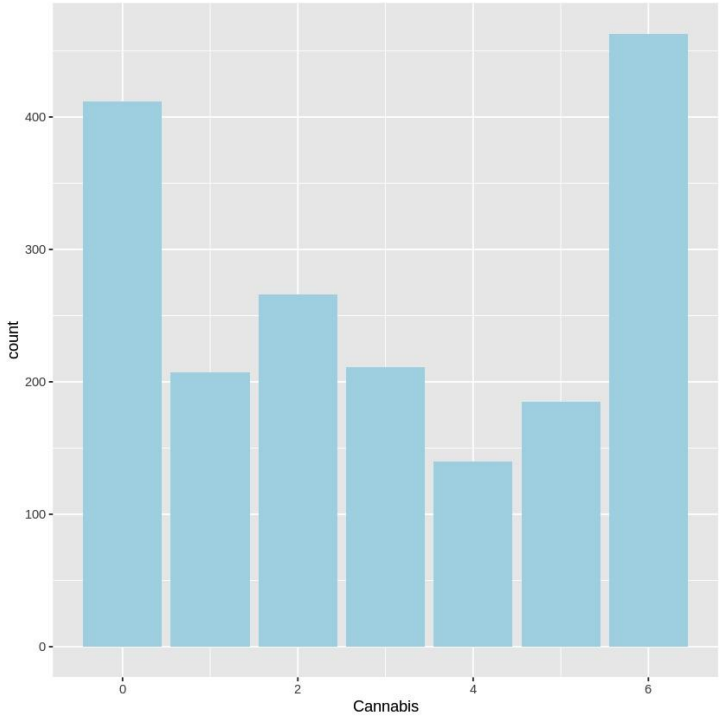
  - Max correlation value and name of column as the first column researched object

  - For example (Yellow Highlighting) Focusing on object Cannabis, highest correlation: Mushrooms Correlation value: 0.579735959383825

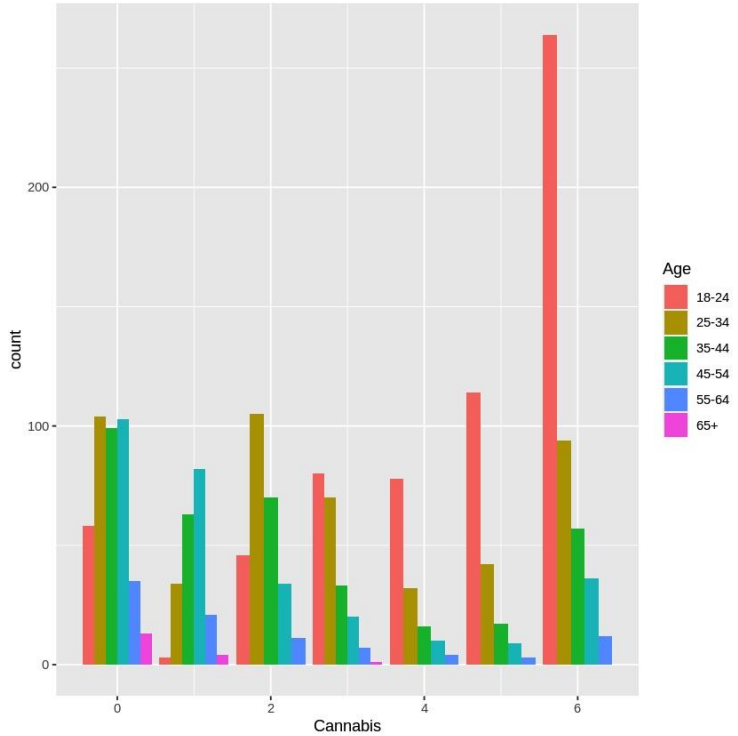
Age	0.354095419716683	Country	Caff	0.126879148256191	Alcohol
Gender	0.220339231983922	Ascore	Cannabis	0.579735959383825	Mushrooms
Education	0.240416985604378	Cscore	Choc	0.124574585387424	Country
Country	0.354095419716683	Age	Coke	0.61065639330959	Ecstasy
Ethnicity	0.138182754852153	Cannabis	Crack	0.527087582650511	Heroin
Nscore	0.272186016518824	Benzos	Ecstasy	0.61065639330959	Coke
Escore	0.308048827313314	Cscore	Heroin	0.527087582650511	Crack
Oscore	0.421534861718457	SS	Ketamine	0.508246586396852	Ecstasy
Ascore	0.247534431601236	Cscore	Legalh	0.553991012261566	Ecstasy
Cscore	0.308048827313314	Escore	LSD	0.668527578338545	Mushrooms
Impulsive	0.623223378722803	SS	Meth	0.519414622062725	Benzos
SS	0.623223378722803	Impulsive	Mushrooms	0.668527578338545	LSD
Alcohol	0.130668713640973	Education	Nicotine	0.515067092600041	Cannabis
Amphet	0.531203890998502	Coke	Semer	0.0991112664292815	Mushrooms
Amyl	0.377136702373278	Coke	VSA	0.319523257338639	Legalh
Benzos	0.519414622062725	Meth	Caff	0.126879148256191	Alcohol

- Bar Chart (Cannabis Counting)

- The proportion of people who used Cannabis last day is largest, which is more than 450.

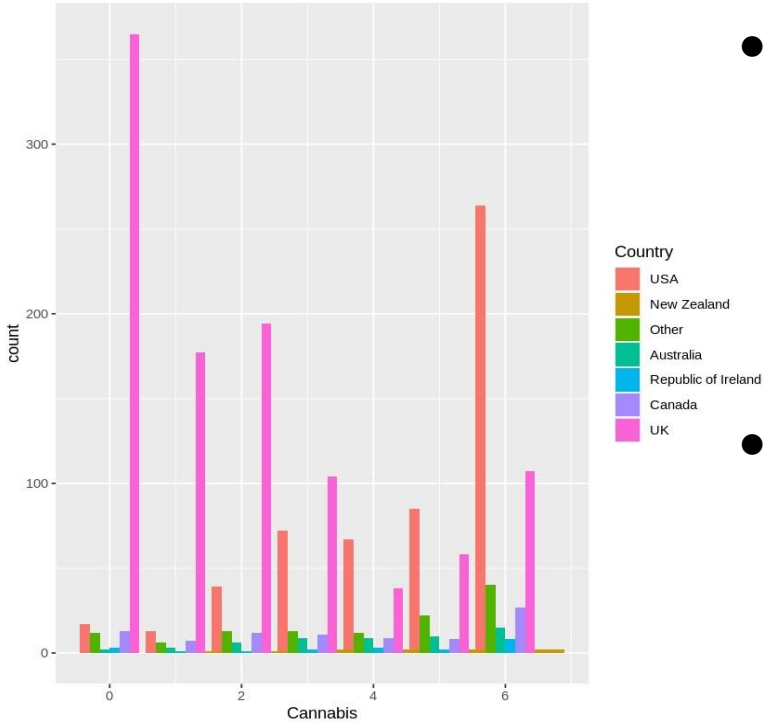


- Bar Chart (Cannabis - Other Variables - Count)



- People who take Cannabis most frequently are young adults, from 18-24 years old.
- And older adults like those more than 55 years old seldom take Cannabis.

- Bar Chart (Cannabis - Other Variables - Count)

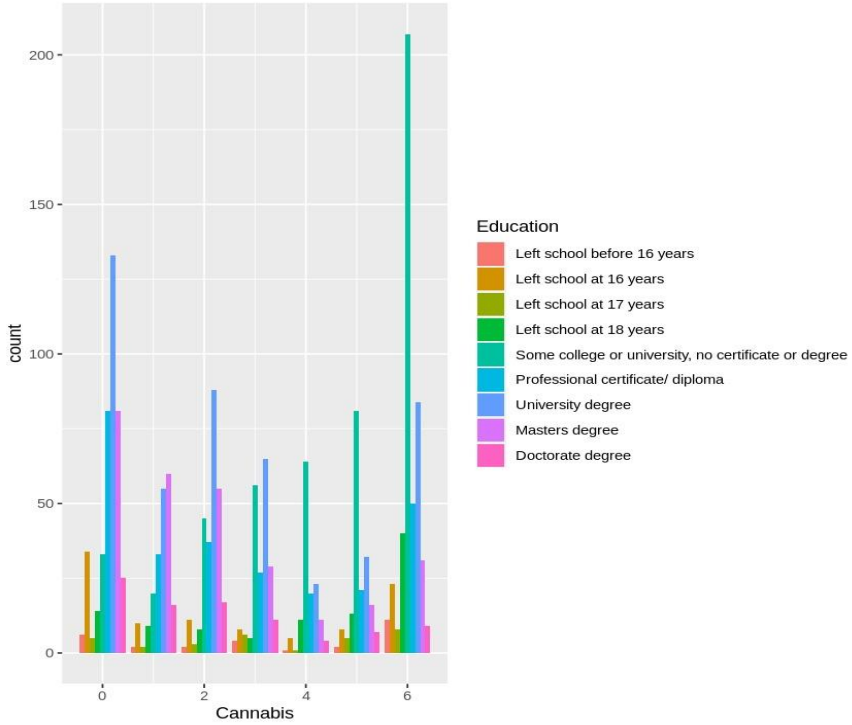


- The most people who take Cannabis is the US, and then the UK;
- Other countries contribute little Cannabis Consumption.

- Bar Chart (Cannabis - Other Variables - Count)

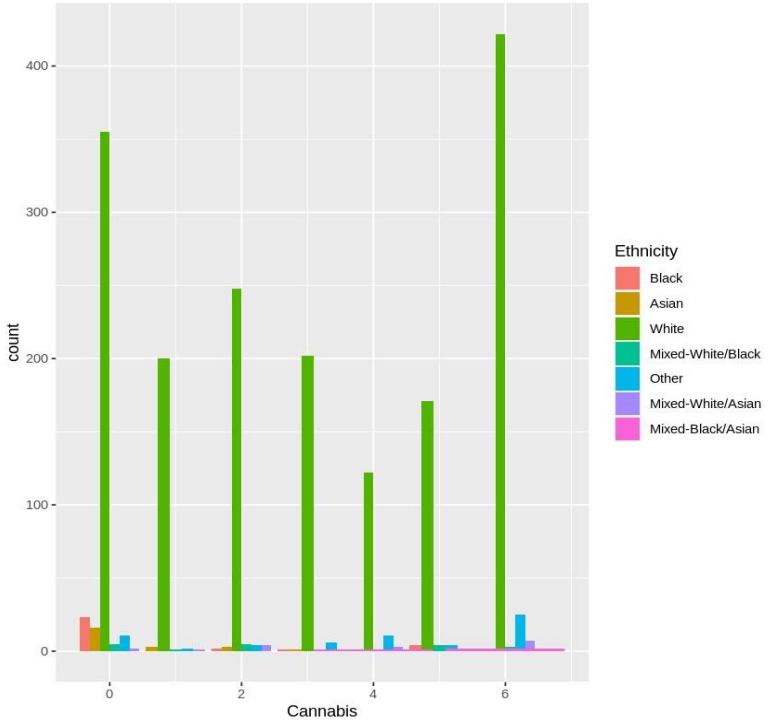
- Those who often buy Cannabis have a relatively high level of Education.

- i.e., entered the University, however, are not successfully graduated.



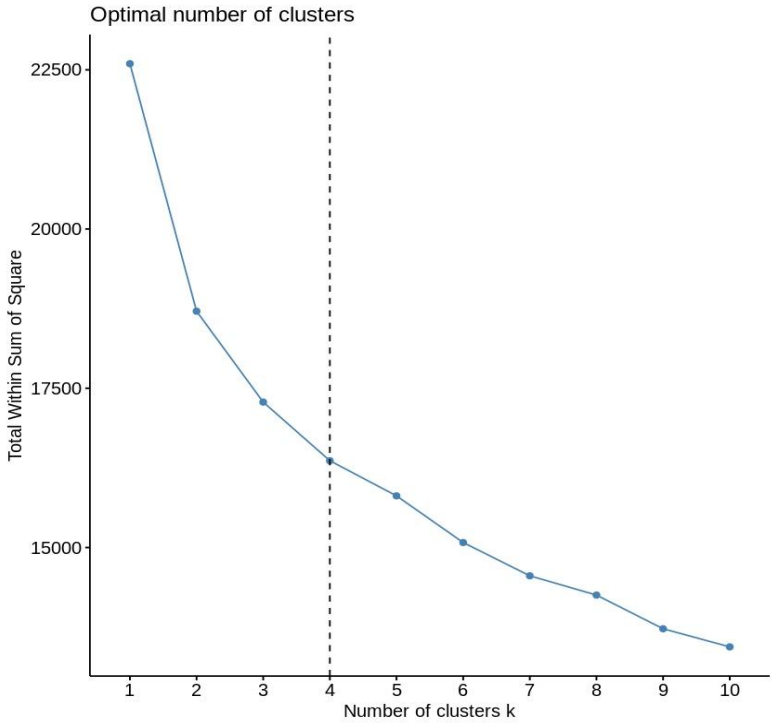
- Bar Chart (Cannabis - Other Variables - Count)

- Most people who buy Cannabis are white, and other species get poorly involved.

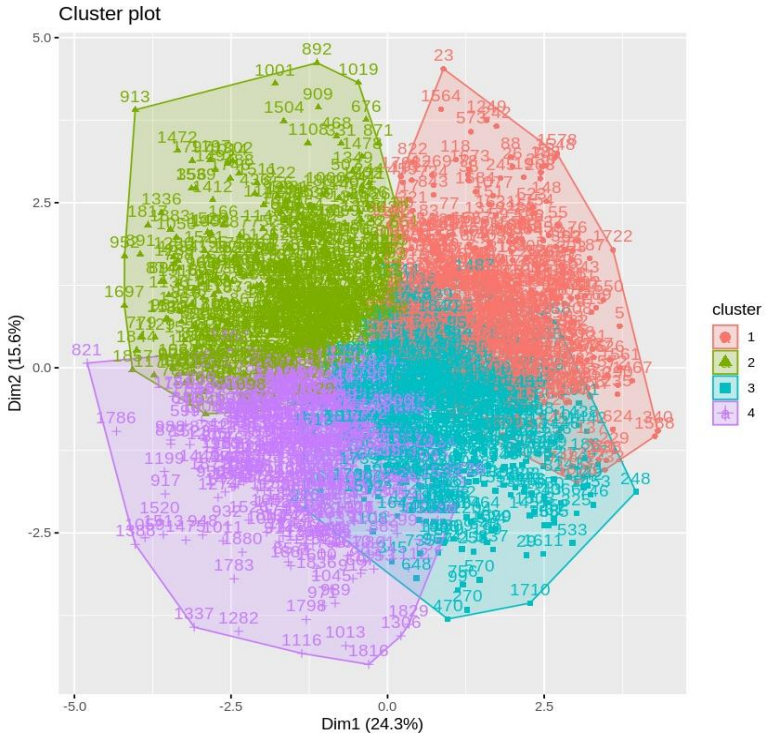


- K-Means Clustering
- Assign similar data to the same cluster.
- Based on squared Euclidean distance and k-means clustering algorithm, we could set k = 2 first to perform the clustering.

- K-Means Clustering
  - Use factoextra package writing a function to choose the most suitable k value.
  - based on the graph and line, we found k=4



- K-Means Clustering



- 4-means clustering

03 Feature Selection

- Stepwise feature selection
  step(), klaR StepClass()
- Feature importance - Tree methods
  Random forest, decision tree (rpart)

Feature importance-

Tree methods

Random Forest

- Use random forest package
- Classification tree and get an error rate
- Variable Importance

Decision Tree

- Variable Importance
- Work with data discrete labels
- Reduce the overfitting to improving the accuracy

05 Model Prediction

- Split 75% data as training, 25% data as testing by Stratified Folds. (R spiltools library)
- Our models:
  - Neural Network
  - KNN
  - Decision tree
  - Random forest
  - SVM
  - Boosting
  - XGboost

Neural Network

- Caret (nnet)
  Nnet is feed-forward neural network with single hidden layer
- Hyper pruning parameter
  Size, decay
- Result
  Final model: size = 3 and decay = 0.1
  Accuracy: 0.636

K Nearest Neighbor

- Choosing the distance calculating function (Euclidean Metric)

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \ldots + (x_n - y_n)^2} = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

- Define the most suitable k value for the data
- Using min-max standardization with the following method:

$$X_{new} = \frac{X - min(X)}{max(X) - min(X)}$$

- Fit the model and check the performance by using result table
- Score: 0.640

Decision Tree

- C50
  - → Model Training
  - → Evaluate the Performance using Testing Data
  - → Calculate the accuracy of testing date

- Result
  - → Accuracy: 0.5596
  - → Error: 0.4404

	2	3	4	5	6
2	191	11	3	3	13
3	15	10	8	7	13
4	10	6	3	8	8
5	11	5	5	9	17
6	23	12	5	24	50

Random Forest

- Use the “caret” package
- Plot OOB error as the number of trees increase
- Select the tuning parameter and specify the grid for the key parameter mtry
- Result

  Final model: mtry = 6

  Accuracy: 0.636

Support Vector Machine (SVM)

- Supervised learning models
- The sets to discriminate are not linearly separable in that space
- The original finite-dimensional space be mapped into a much higher-dimensional space, making the separation easier.

- The accuracy is 0.651

Boosting

Comparison

	randomForest	BoostingTree	SingleTree
Accuracy	★★	★★★	★
Interpretation	★	★	★★★
Easy-to-Use	★★★	★	★★★
Computation	★	★★	★
Tree Size	Large	Small	
Parallel	Yes	No	

- Boosting
- Model: method='cv', number=10, summaryFunction = multiClassSummary
- interaction.depth = c(2,3,4,5,6), n.trees = c(100,300,500)
- The best result: 5 levels: 0.6255

XGBoost

- Gradient boost tree
- Hyper pruning parameter

  Find the best parameter by cross_validation score in training set, choose different parameters: eta, max_depth, subsample, etc.)
- Result

  Final model: eta = "0.01", max_depth = "4", subsample = "6"
  Accuracy: 0.653

05 Discussion

	Accuracy 5 levels
Neural network	0.636
KNN	0.640
Decision Tree	0.557
Random Forest	0.636
SVM	0.651
Boosting	0.632
XGboost	0.653

- Result: (accuracy table for all models)
- XGBoost: 0.653
- SVM: 0.651

05 Discussion

- Accuracy table for 5&7 levels:

- Boosting : (5 levels : 0.6255319)

	X2	X3	X4	X5	X6
X2	205	18	6	9	20
X3	5	7	2	3	7
X4	1	0	2	1	3
X5	2	3	4	3	7
X6	8	25	21	31	77

(7 levels: 0.5010616)

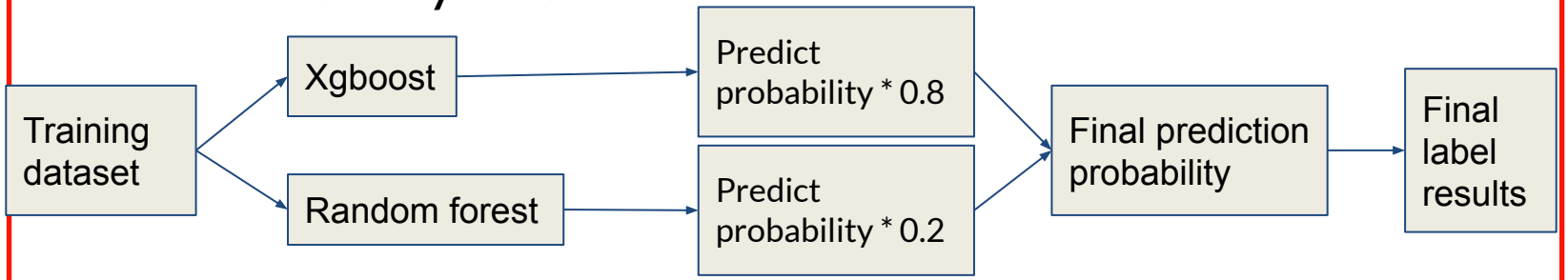
	X0	X1	X2	X3	X4	X5	X6
X0	85	25	14	4	2	2	5
X1	10	17	5	2	0	1	3
X2	6	7	35	10	3	4	7
X3	0	0	5	4	3	2	5
X4	0	0	0	0	1	0	0
X5	0	0	0	1	1	1	2
X6	2	2	8	32	25	37	93

05 Discussion

- Not enough data for some levels (3,4,5)
- What about merge levels of target variable?
  - Accuracy increases
  - But model itself should be the same
  - Solve the problem of not enough data

05 Discussion

- Model Sampling
- Combine different models, reduce bias
- Predict probability for each labels.
- 80% from Xgboost, 20% from Random forest
- Accuracy: 0.657



05 Conclusion

- Final Result: model sampling (XGB + RF)
- Merge levels solve not enough data
- Application

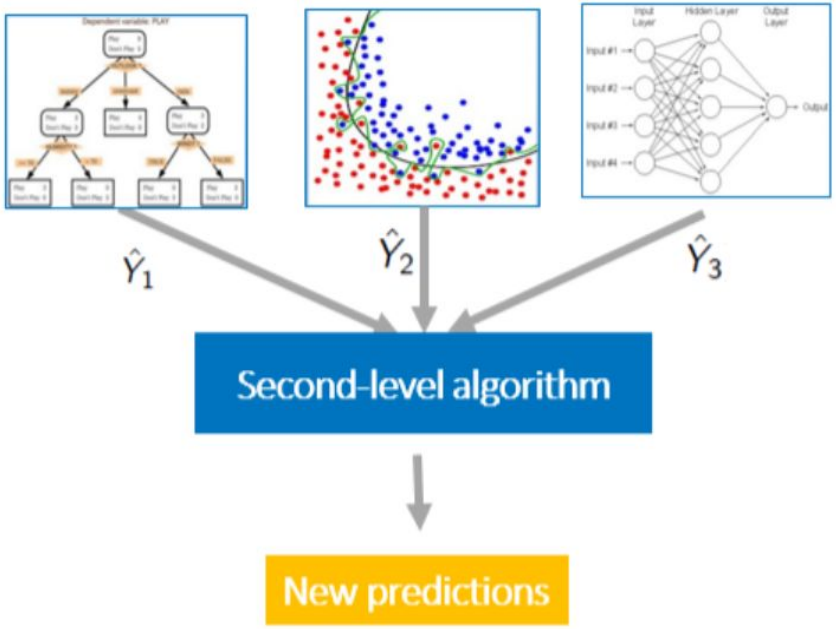
  2-level prediction problem (CL0 with drug, CL1-6 no drug)

  Our group has more meaningful result

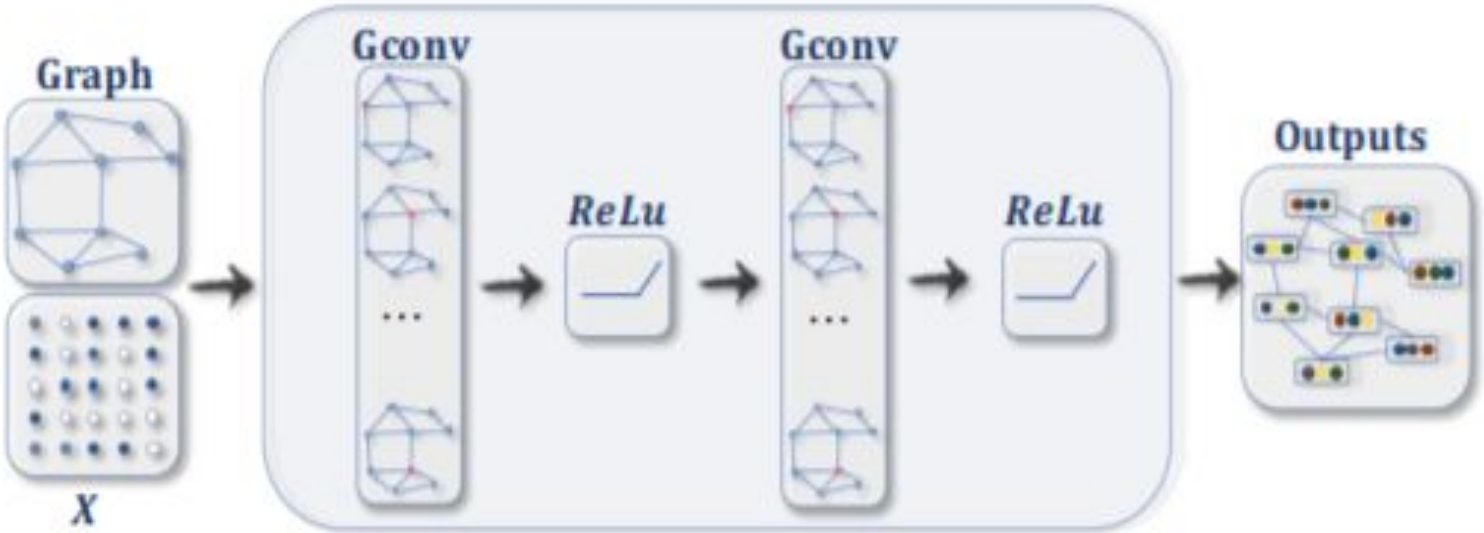
05 Conclusion

- Further improvement
  (might use these later in the future improvement, we didn't use them yet)
  - Collect more data for the research
  - Stacking classifier
  - Advanced Deep learning models (CNN)
  - PCA, SVD

Stacking Classifier



CNN



PCA & SVD

- Principal Component Analysis
- Explained variance ratio
  PCA uses 50% columns, SVD uses 70% columns,
  With 90% of explained variance ratio.
- Reduce computational cost
- Small size of data

Thank you