Project Theme: Human Resources Analytics

Team Information:

Name	Role	Student_ID	Work
李雪菲	Leader	MF1372074	Grasp the process of project and timeline, data exploration, data preprocessing,construct decision tree
孔敏	Member	MF1732063	Data exploration,preprocessing,construct decision tree
安磊	Member	MF1732001	PCA and Clustering
户胜浩	Member	MF1732004	Clustering

1.Introduction:

Our project is based on the dataset from kaggle.com with the link:
https://www.kaggle.com/ludobenistant/hr-analytics

And the dataset, code and process records can be found in this link:
https://github.com/DataMining2017NJU/HR_Analytics/tree/develop

When we implement this report, we refer to some kernels in Kaggle.com,there are many excellent suggestions on it, when we learnt a lot from them and use some of the method in it, what`s more, we add some new interesting things into our project. Also, we practice some algorithm in the textbook by Weka and also solving many problems when we code.

1.1 About Dataset:

This dataset is simulated, with left rate 23.808%. Fields, explaination and type of this dataset are listed below:

Table 1 fields in the dataset

satisfaction_level	Level of satisfaction (0-1)	Numeric
last_evaluation	Time since last performance evaluation (in Years)	Numeric
number_project	Number of projects completed while at work	Numeric

average_montly_hours	Average monthly hours at workplace	Numeric
time_spend_company	Number of years spent in the company	Numeric
Work_accident	Whether the employee left the workplace or not (1 or 0) Factor	Numeric
left	Whether the employee left	Numeric
promotion_last_5years	Whether the employee was promoted in the last five years	Numeric
sales	Department in which they work for	String
salary	Relative level of salary (high)	String

1.2 Task:

We want to clustering dataset to see the class of the data and use decision tree to conclude what kind of people in this dataset, why people leave their job and what kind of people who is possible to leave.

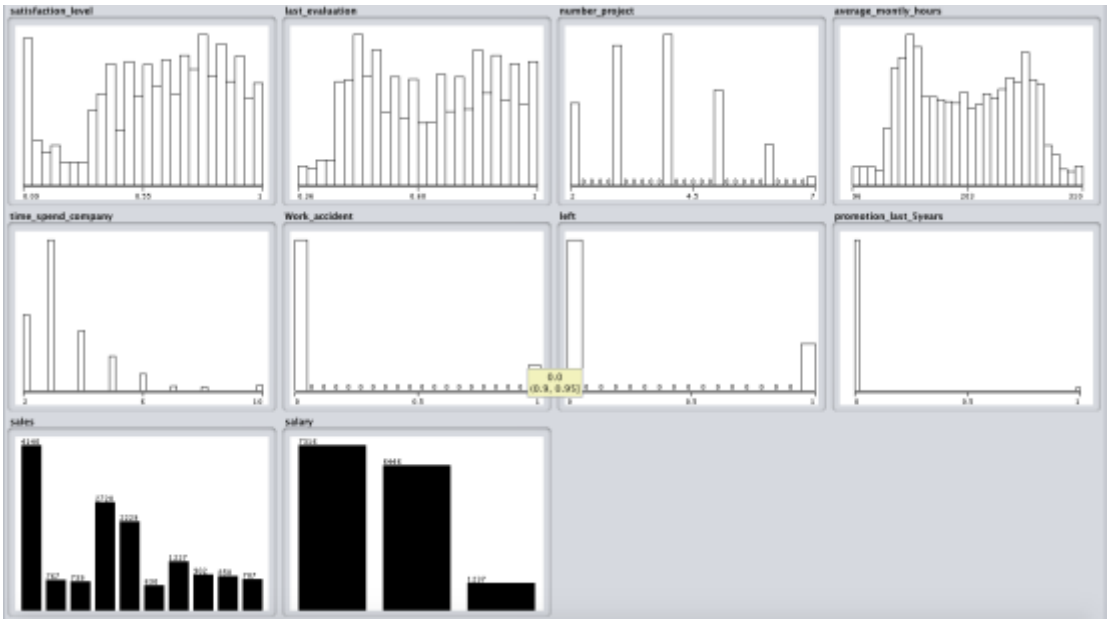
2. Data exploration and preprocessing.

Typically, cleaning the data requires a lot of time.This dataset from Kaggle is simulated and contains no missing values.So after we check the dataset, we do some data explorations and preprocessing.

2.1 Data exploration

We use weka to draw histographs of each attributes and their value. We found attributes: *satisfaction_level, last_evaluation, average_monthly_hours, number_project* has a similar shape with normal distribution.

Figure1 Statistics



2.1.1 Statistic analysis

We also use weka to calculate the statistics data of dataset, and get the result below:

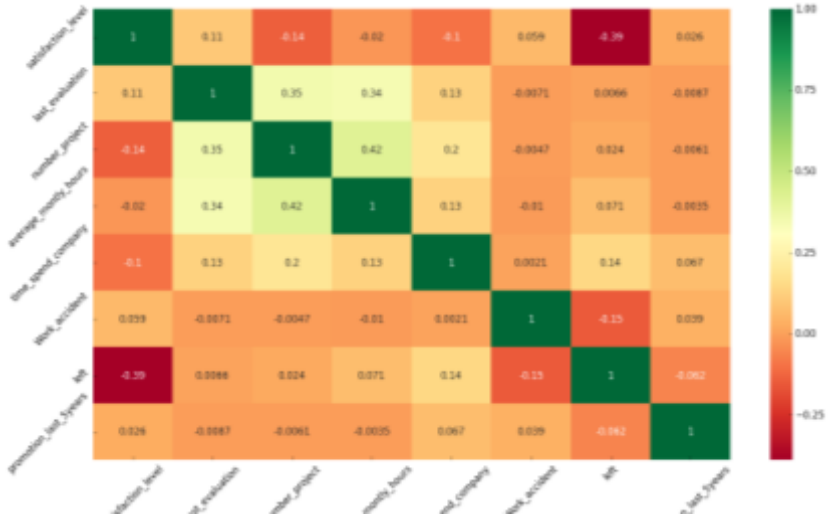
Table 2 statistics of fields

	Min	Max	Std	Mean	Mode	Median
satisfaction_level	0.09	1	0.249	0.613	0.1	0.64
last_evaluation	0.36	1	0.171	0.716	0.55	0.72
number_project	2	7	1.233	3.803	4	4
average_montly_hours	96	310	49.943	201.037	0 135/ 1 156	200
time_spend_company	2	10	1.46	3.498	3	3
Work_accident	0	1	0.352	0.145	0	0
left	0	1	0.426	0.238	0	0
promotion_last_5years	0	1	0.144	0.021	0	0

From table 2, we found that the mean/mode/median of *number_project* are familiar with each other. So it`s possible number_project fit the normal distribution.

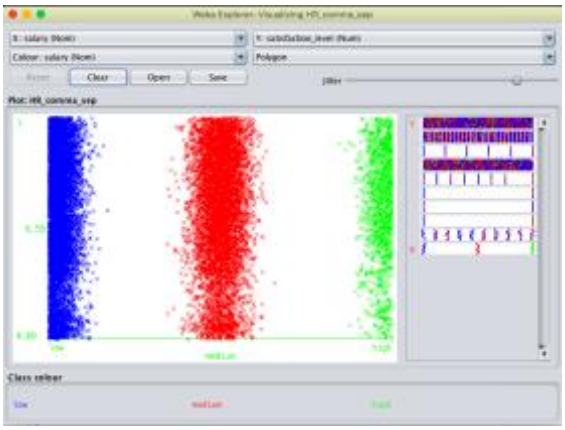
2.1.2 Calculate correlation

Figure2 Correlation of attributes



In Figure2, we use python to calculate the correlation of attributes in this dataset, the result show that *number_project* and *average_monthly_hours* has the most big positive correlation which is 0.42, and after that is *number_project* and *last_evaluation*, which is 0.35.Surprisely, *Left* and *satisfaction_level* has a negative correlation which is -0.39.It maybe means person with high *satisfaction_level* left a lot.

Figure 3 Visualization of salary and satisfiction_level



In figure3 ,we have x_axle: *salary*, and y_axle: *satisfaction_level*. In our opinion , the most possible reason maybe the *salary*.But from this graph we`ll know, there are many people who earned less but still have high satisfaction, and

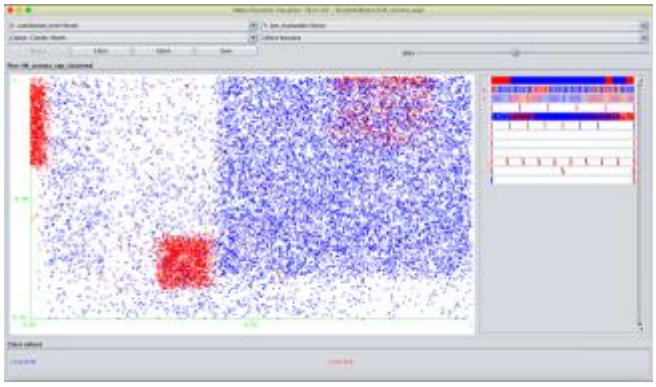
earned a lot but has low satisfiction.It also stresses that there`s no direct correlation between salary and satisfiction level.

3. Data Mining:

3.1 Cluster Analysis

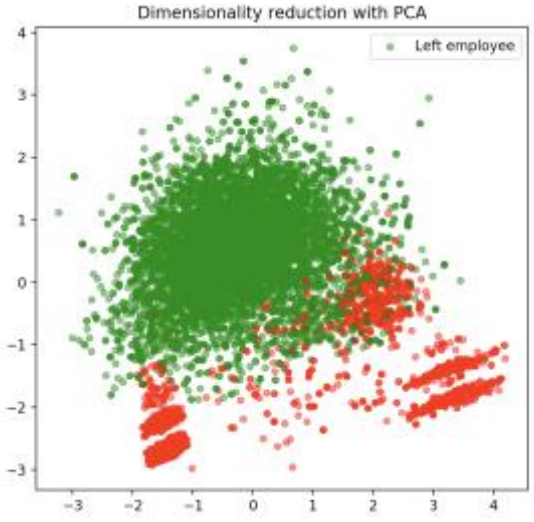
We use weka to cluster the data and got the result below, it`s a little bit hard to tell the clear cluster.

Figure3 A cluster result of satasfiction_level and last_evaluation



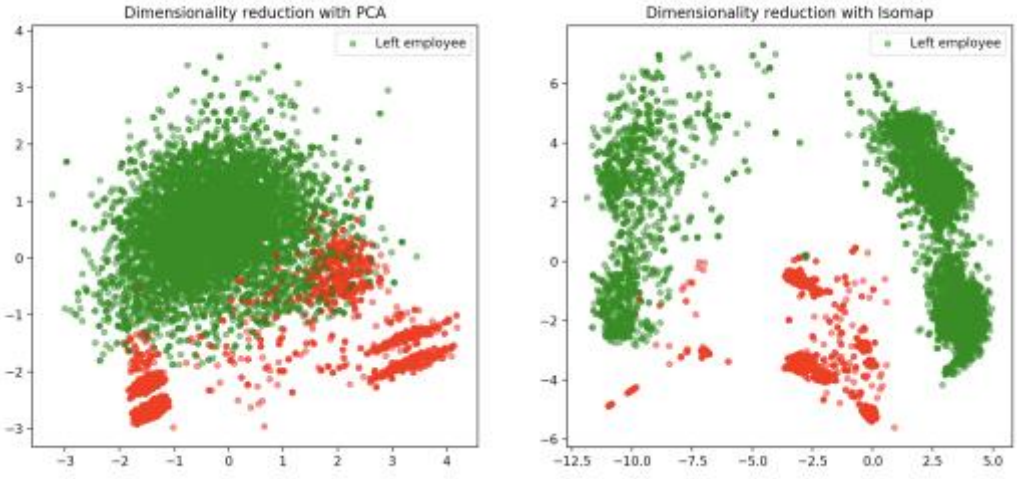
At the beginning of the data exploration, we check features of data, we will perform a dimensionality reduction in order to identify groups. Firstly we use PCA to perform a dimensionality reduction.

Figure4 Dimensionality reduction with PCA



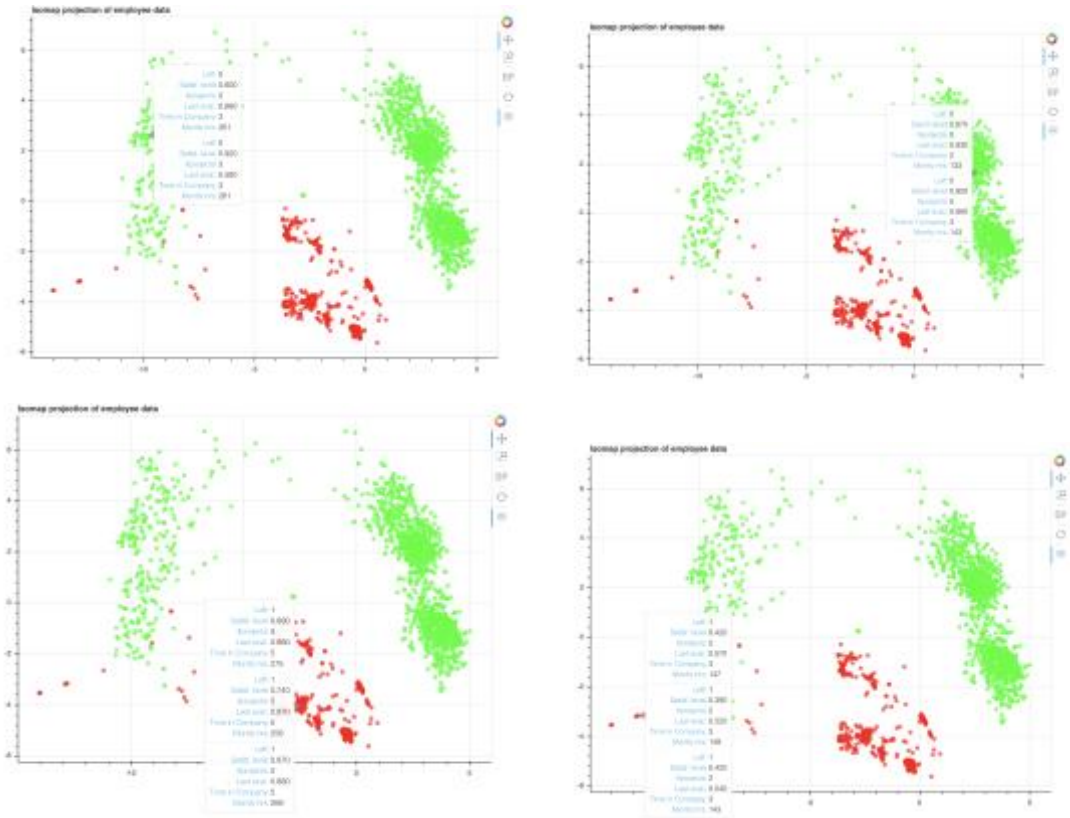
The figure3 show that PCA doesn't show a great separation between the left and stayed employees, so we choose another methods, After learning from others` experience, we chose ISOMAP to perform a dimensionality reduction.

Figure5/6 Result of imensionality reduction with PCA and Isomap



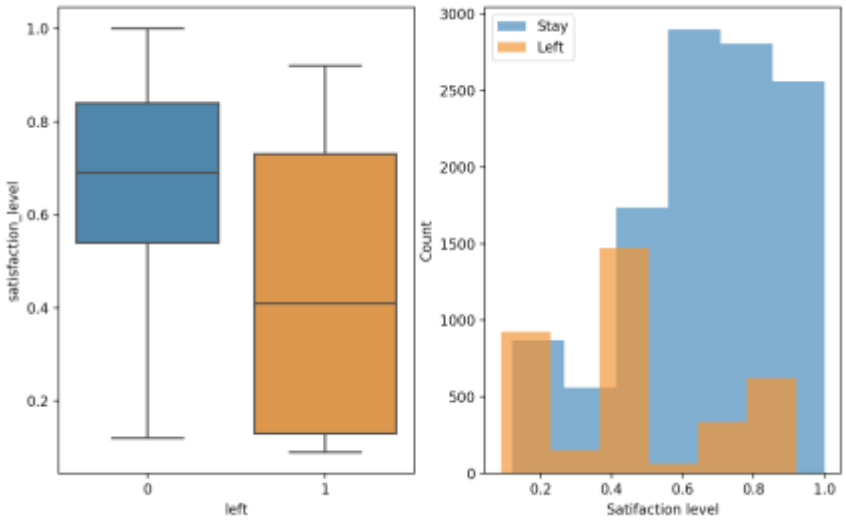
From comparison we saw that ISOMAP is better than PCA in this dataset, The red points correspond to employees who left. The green points represent employees who stayed.

Figure7/8/9/10 Result of isomap projection



After that , we wanted to see the impact of *satisfaction_level* on turnover.

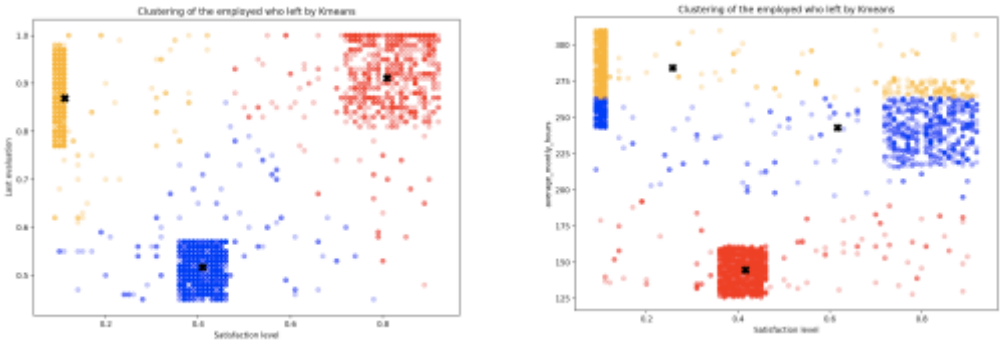
Figure 11 Box_plot of left and satisfaction_level and histograph of satisfaction_level and count



The satisfaction level is the most correlated feature with *left*. Here we can see that employees who left have a lower satisfaction level that those who stayed.

We performed a kmeans clustering by using features *satisfaction_level*、*last_evaluation* and *average_monthly_hours* to isolate several groups：

Figure12/13 Clustering result of satisfaction_level and last_evaluation/Satisfaction_level and average_monthly_hours



We can see that using *satisfaction_level* and *last_evaluation* clustered better, then we got 3 groups between employees who left the company:

Successful and Satisfied employees

Successful and Unsatisfied employees

Unsuccessful and Unsatisfied employees

3.2 Decision Tree

We construct a decision tree to identify the influence of other attributes on the attribute of *left*.

3.2.1 Data preparation

Before we construct the tree,we should import the data first. From these datasets we know the value of *left* attribute is yes or no,we can predict the *left* attribute value by other attributes.

3.2.2 Data preprocessing

Consider the dataset we use,we can see when we use the dataset for predict,we droped the attribute of *sales* and *salary*,because their data type are not numeric type, they can not calculate the value of cross-validation where cross-validation is a part of construct a decision tree algorithm.

3.2.3 Algorithm selection

And we use the scikit-learn algorithm to construct a decision tree.

3.2.4 Construct a decision tree

The core code is as follows:

```
local_path = "/Users/lixuefei/Documents/GitHub/HR_Analysis/HR_comma_sep.csv"
data_name = ['satisfaction_level','last_evaluation','number_project',"average_mont

dataset = pd.read_csv(local_path,header = None,names = data_name)#columns=data_nam

#print(type(dataset))
X_train = dataset.drop(['left', 'sales','salary'],axis = 1) #inplace=True .convert

#print(X_train)

Y_train = dataset["left"]#.convert_objects(convert_numeric=True).dtypes

train_features = X_train.columns

#kfold = StratifiedKFold(Y_train,n_folds=10,random_state=2)
# = StratifiedKFold(Y_train,n_splits=10,random_state=2)
kfold = StratifiedKFold(Y_train,n_folds=10,random_state=2)
DTC = DecisionTreeClassifier(max_depth=3)

cv_results = cross_val_score(DTC,X_train, Y_train, cv=kfold, scoring="accuracy")
#print(type(cv_results))

print(cv_results.mean())
#print(type(DTC))
DTC.fit(X_train,Y_train)
```

3.2.5 The visual display

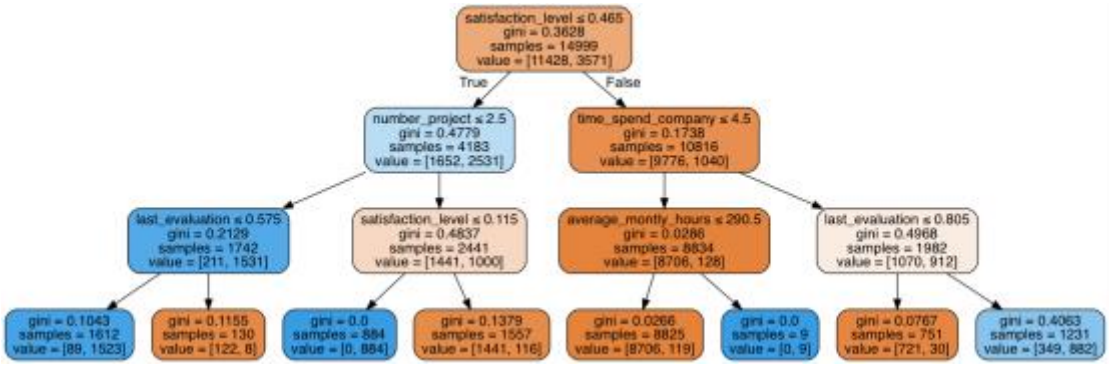
After constructing the decision tree, it can be displayed graphically. The code that transforms the decision tree into a graphical representation is as follows:

```
dot_data = StringIO()
tree.export_graphviz(DTC, out_file=dot_data,feature_names=train_features,

graph = pydotplus.graphviz.graph_from_dot_data(dot_data.getvalue())#[0]
graph.set_lwidth(400)
graph.set_lheight(300)

#img = Image(graph.create_png(prog='dot'))
img = Image(graph.create_png())#prog='dot'
display(img)
graph.write_png("out.png")
```

The last sentence of the code is to output graphics.

Here we import the pydotplus and the Ipython.display for drawing graph.

And the final decision tree graph is as follows



From here we can see the *satisfaction_level* is the most influential attribute on the *left* sttribute,and the second is *num_project* and *time_spend_company*, The third id *last_evaluation* and so on. Each non-leaf node also has its threshold value. From here we can know the size of the influence of each attribute on the class label properties.The class label properties is exactly the *left* attribute.

Conclusion

Clustering: SuccessUnhappy group left because they worked too much. UnsuccessUnhappy group left because they are not really involved in their company. SuccessHappy group is willing to the stay group unless they spent a long time in company.

What did we learn from this project

Teamwork and practice; know more about process of data mining.

Reference:

https://www.kaggle.com/yassineghouzam/don-t-know-why-employees-leave-read-this

<Data Mining Concepts and Techniques 3rd Edition>Jiawei Han, 2012-8, ISBN: 9787111391401