Human Resource Analytics

Exploratory Data Analysis

Dataset : Human Resource Analytics

Source : Kaggle

Description :

Employee attrition analysis is an important process in most companies. It helps the companies to understand how they can develop the work atmosphere to make it more conducive to employees and will help in overall reduction in employee attrition.

The dataset that I have chosen is the Human Resource departmental data which includes different reasons factoring in as the cause of employee attrition. Some of these are:

- Satisfaction Level
- Last evaluation
- Number of projects
- Average monthly hours
- Time spent at the company
- Work accidents
- Promotion in the last 5 years
- Departments (column sales)
- Salary

What are we predicting ?

Using the given factors for the employee attrition, we are predicting the employees who are more probable to leave the company.

Part A

Initial Observations

Data Cleaning :
The data is clean and does not need any imputing.

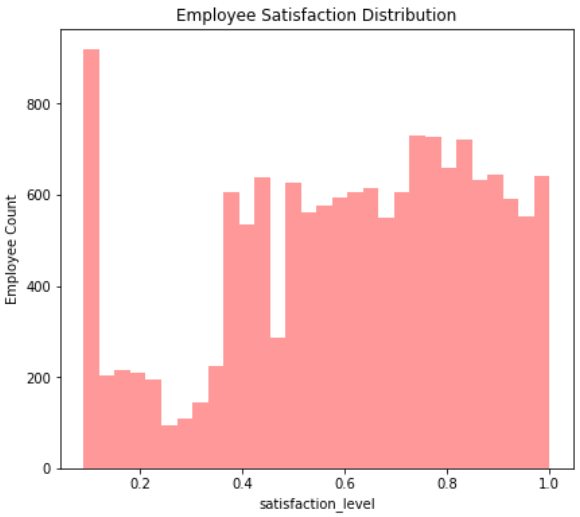
Missing Values :
There are no missing values in the dataset.

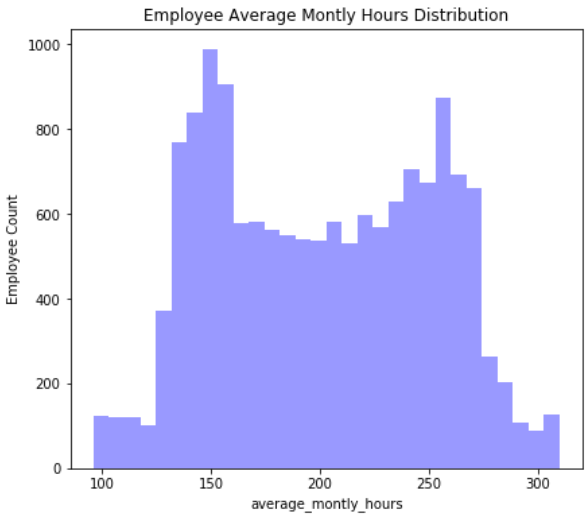
Inappropriate Values :
There are no inappropriate/junk values.

No bad data. Hence nothing to be removed or rectified.

Assess Data distribution

I have mapped two of the independent variables against the count of employees and below are the results of the distribution plot





Initial Conclusion

From the Employee Satisfaction Distribution graph, we understand that there is a rise in dissatisfaction level among the employees.

From the Employee Average Monthly Hours distribution graph, we see that more employees are spending around 160 hours – 260 hours at work. Employees spending lesser and more number than this range is very less.

Statistical Calculations

Data type :

left int64
satisfaction_level float64
last_evaluation float64
number_project int64
average_montly_hours int64
time_spend_company int64
Work_accident int64
promotion_last_5years int64
sales object
salary object

Attrition Ratio

We calculate the attrition ratio for the employees

Attrition Ratio = Employees who left the company/Total number of employees

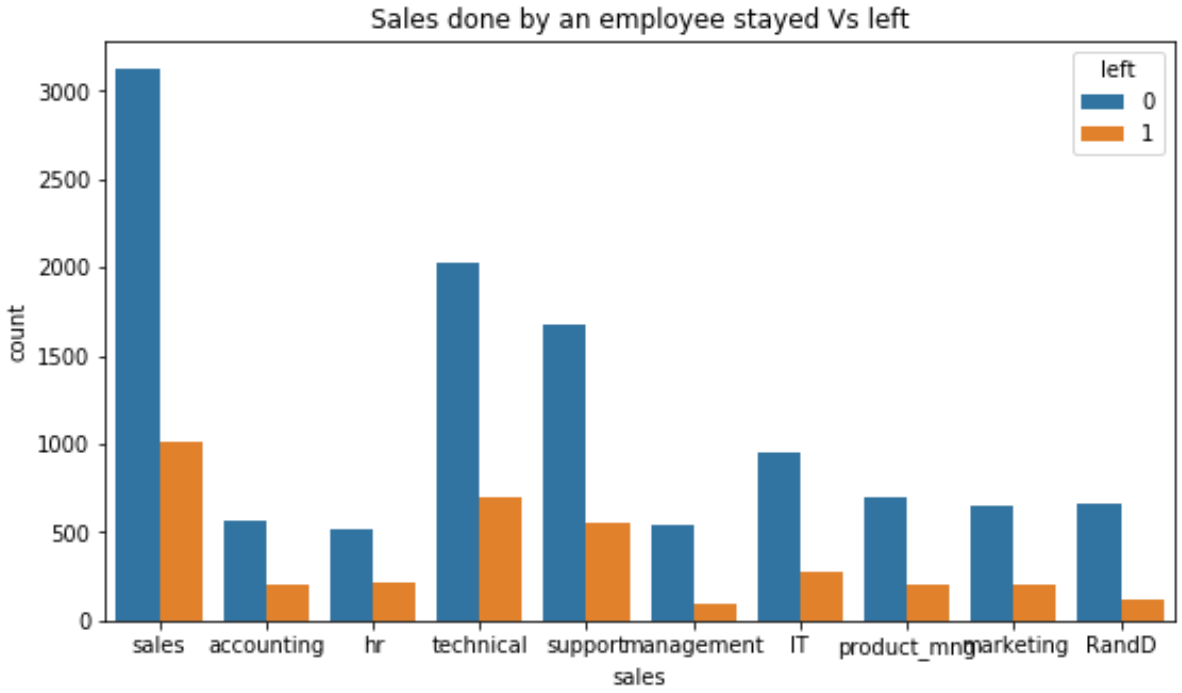
Attrition Ratio

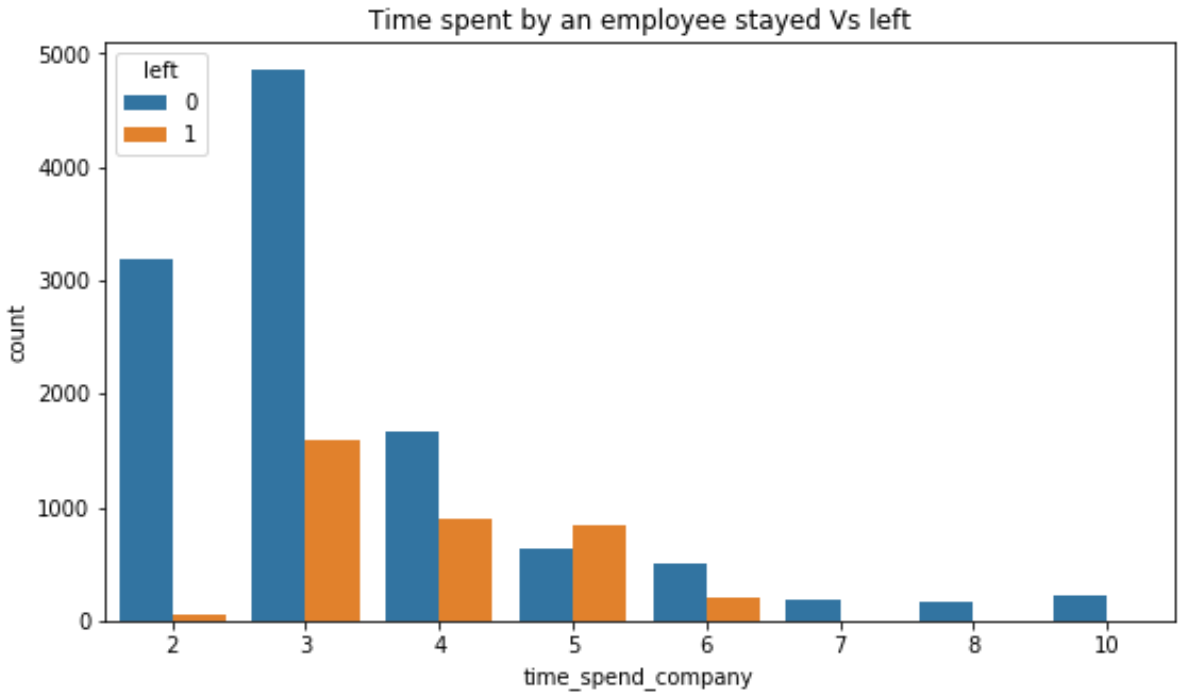
Employees who stayed back	0.761917
Employees who left	0.238083

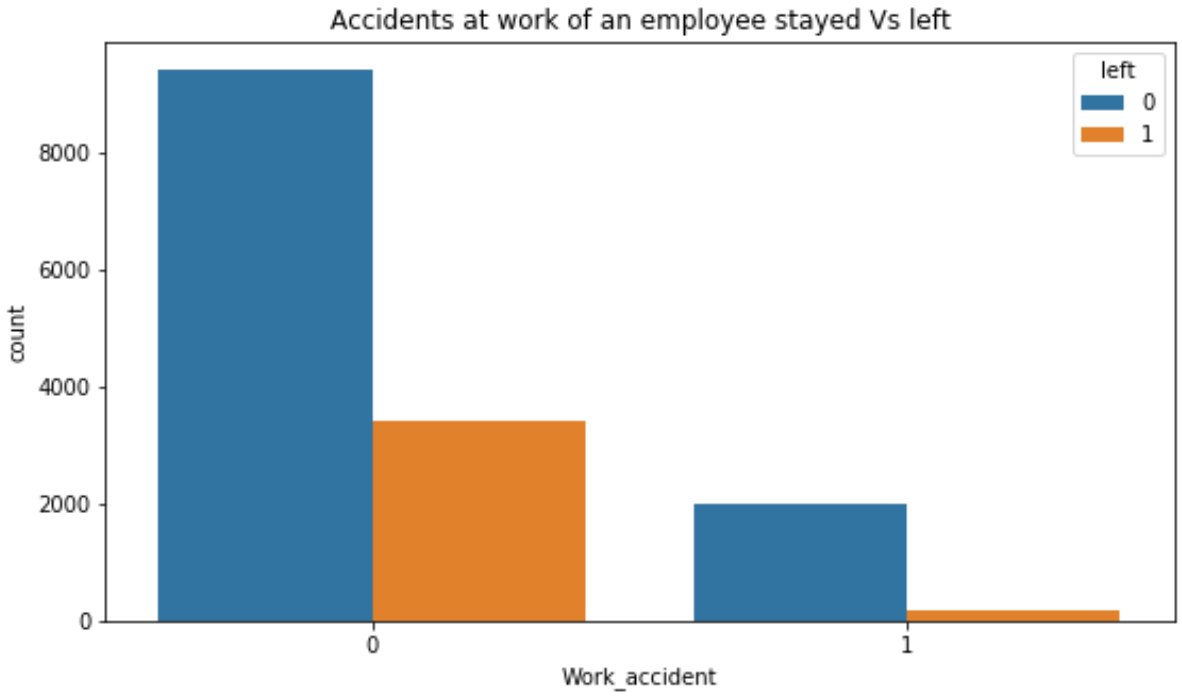
Attrition Summary

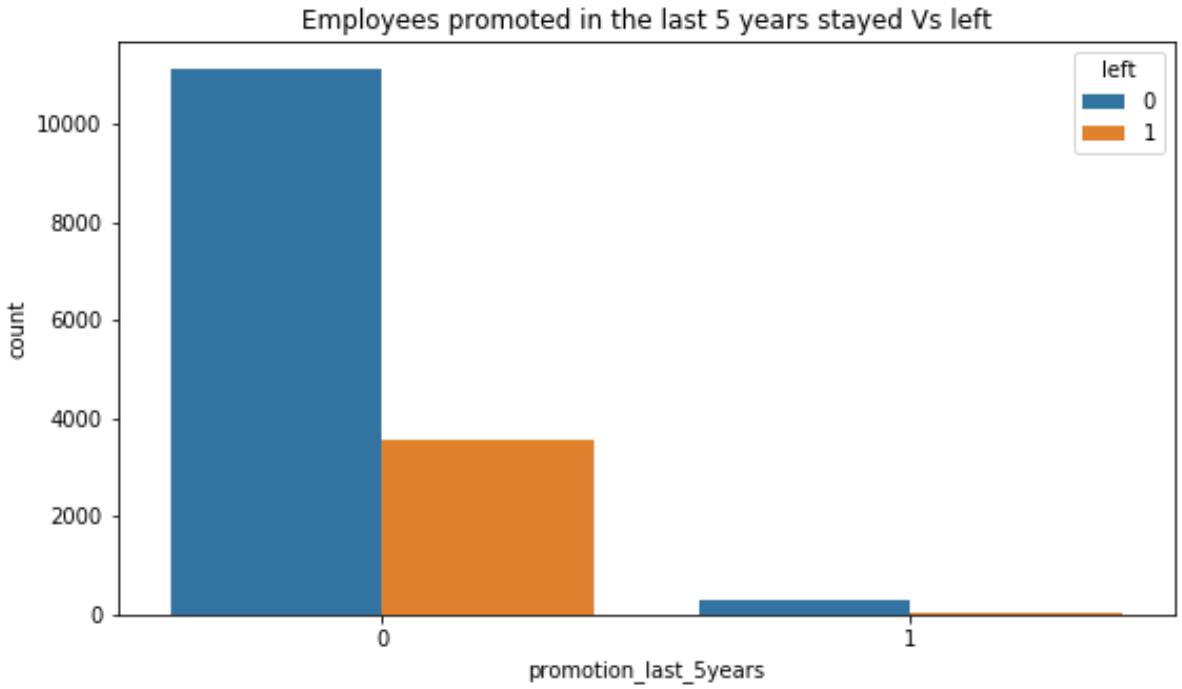
	satisfaction level	Last evaluation	Project Count	Av. Monthly hours	Time spent company	Work accident	Promotion last 5years
left							
0	0.666810	0.715473	3.786664	199.060203	3.380032	0.175009	0.026251
1	0.440098	0.718113	3.855503	207.419210	3.876505	0.047326	0.005321

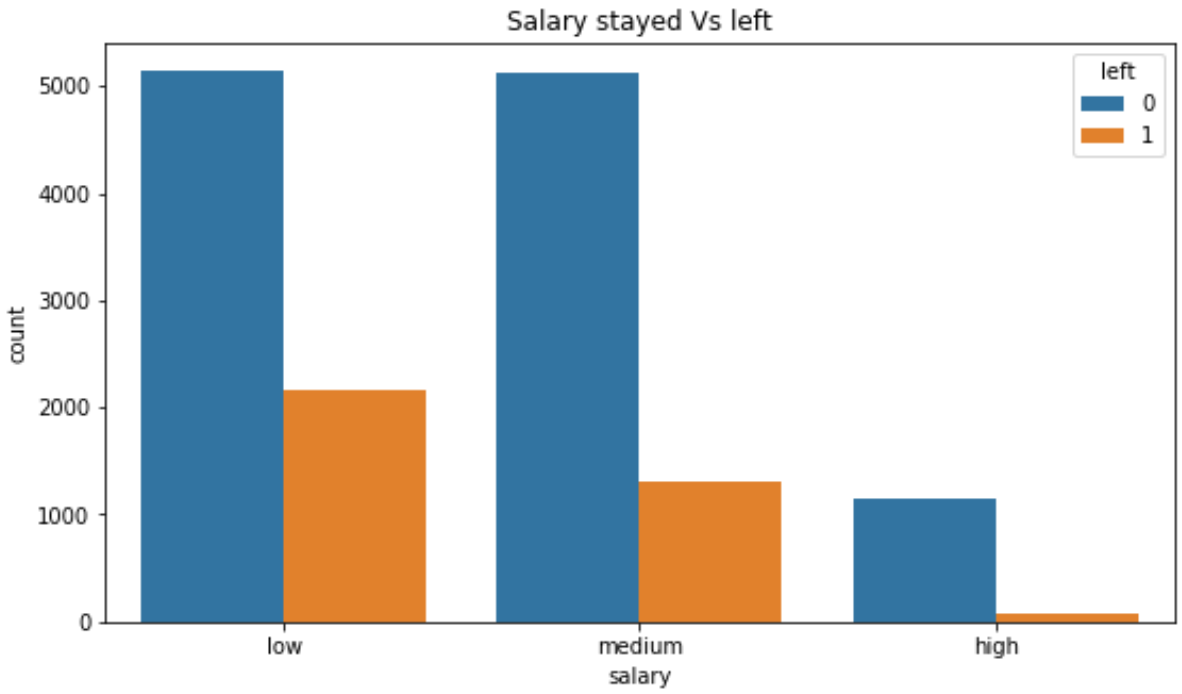
We also view count plot as initial analysis for our case :

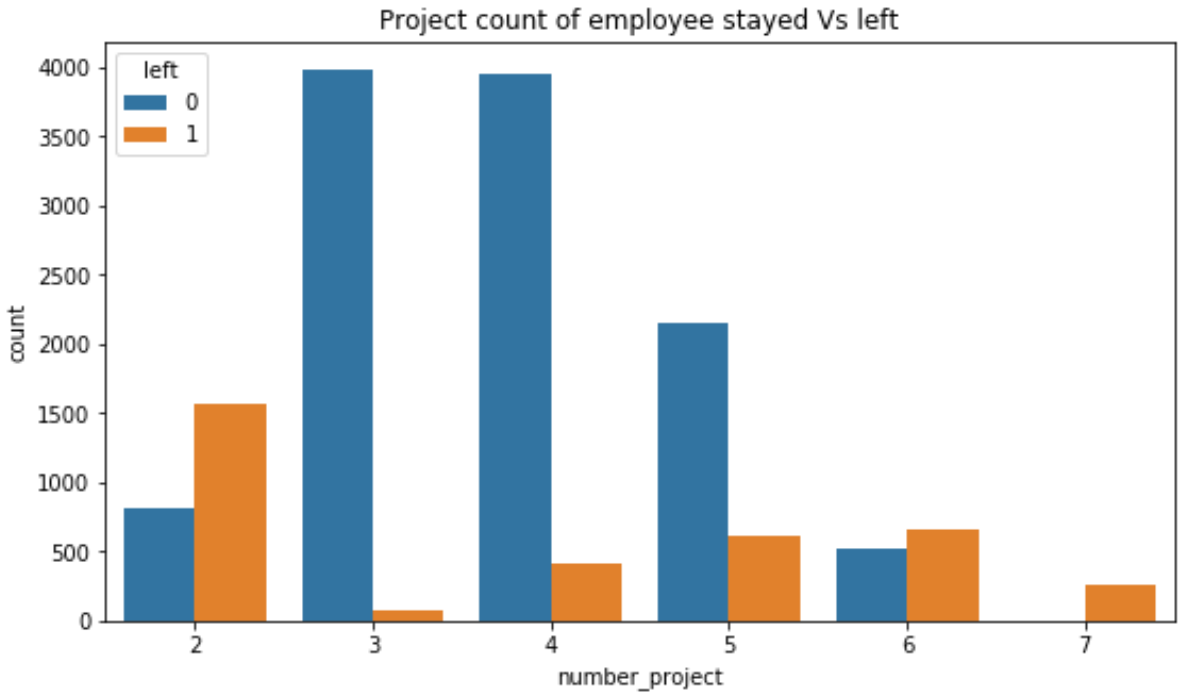



Time spent by an employee stayed Vs left
left
0
1
5000
4000
3000
2000
1000
0
count
2
3
4
5
6
7
8
10
time_spend_company


Accidents at work of an employee stayed Vs left
left
0
1
8000
6000
4000
2000
0
count
0
1
Work_accident


Employees promoted in the last 5 years stayed Vs left
left
0
1
10000
8000
6000
4000
2000
0
count
0
1
promotion_last_5years


Salary stayed Vs left
left
0
1
5000
4000
3000
2000
1000
0
count
low
medium
high
salary



I also draw a correlation matrix to understand how the factors have coupled in the attrition rate :

```
#Drawing a correlation matrix to assess how the columns are related
cor_mat = newdataframe.corr()
cor_mat = (cor_mat)
sns.heatmap(cor_mat,
        xticklabels=cor_mat.columns.values,
        yticklabels=cor_mat.columns.values)
```

cor_mat

Below is my observation,

Positive correlation between evaluation(e), project count(p) and av. Monthly hours(t)

p Vs. t	0.417211
e Vs. p	0.349333
t Vs. e	0.339742

This implies that employees who spent more hours working on more projects were evaluated highly.

On the contrary, the negative correlation for Satisfaction level Vs. the Left implies that people with lower satisfaction level were the ones to leave in higher proportion.

Overall conclusion of Exploratory Data Analysis

- Employees who left are the ones who have spent less time with the company
- Employees who have been promoted in the past 5 years have stayed back
- Employees leaving the company are the ones who have worked on 2, 6 or 7 projects
- Attrition rate is higher in the people with lower income
- The people from Sales tend to leave at a higher rate than the people in management.
- There are lot of people working in the company. But the attrition spikes in the 5th year

Part B

Data Modelling using Machine Learning Algorithm

I have randomly chosen two of the most popular machine learning algorithms and have concluded the better between the two depending on the accuracy in my case. There are many algorithms which can be applied and the data can be modelled other than these. For the case of simplification, I have chosen Decision Tree and Logistic Regression and below are the results of the modeling :

Decision Tree Model

```
from sklearn.metrics import classification_report
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import roc_auc_score

decisionTree = tree.DecisionTreeClassifier(
   #max_depth=3,
   class_weight="balanced",
   min_weight_fraction_leaf=0.01
   )
decisionTree = decisionTree.fit(X_train,Y_train)
print ("Decision Tree Model")
print(classification_report(Y_test, decisionTree.predict(X_test)))
decisionTree_roc_auc = roc_auc_score(Y_test, decisionTree.predict(X_test))
print ("Decision Tree AUC = %2.2f" % decisionTree_roc_auc)
```

	precision	recall	f1-score	support
0	0.97	0.97	0.97	1143
1	0.91	0.90	0.91	357
avg / total	0.96	0.96	0.96	1500

Decision Tree AUC = 0.94

Logistic Regression

```
from sklearn.linear_model import LogisticRegression
logReg = LogisticRegression(class_weight = "balanced")
logReg.fit(X_train, Y_train)
print ("Logistic Regression")
print(classification_report(Y_test, logReg.predict(X_test)))
logReg_roc_auc = roc_auc_score(Y_test, logReg.predict(X_test))
print ("Logistic Regression AUC = %2.2f" % logReg_roc_auc)
```

	precision	recall	f1-score	support
0	0.91	0.74	0.82	1143
1	0.48	0.77	0.59	357
avg / total	0.81	0.75	0.77	1500

Logistic AUC = 0.76

Result:

It is seen that Decision Tree gives us more accuracy and hence is the better algorithm.

Discussion:

The above accuracy is when all the independent features are used for prediction. If selective independent features are used which hold more weight in our prediction, this can help in tuning our model better. This process is called Feature Extraction and can also help in understanding the top factors responsible for attrition.

```
target_name = 'left'

X= newdataframe.drop('left', axis=1)

Y= newdataframe[target_name]


X_train, X_test, Y_train, Y_test= train_test_split(X,Y, test_size=0.10, random_state = 123, stratify=Y)


decisionTree = tree.DecisionTreeClassifier(

class_weight = "balanced",

 min_weight_fraction_leaf=0.01

)

decisionTree = decisionTree.fit(X_train,Y_train)
```

```
### plot the Features ##

feature_extraction = decisionTree.feature_importances_

feat_names = newdataframe.drop(['left'],axis=1).columns


indices = npy.argsort(feature_extraction)[::-1]

plot.figure(figsize=(12,6))

plot.title("Feature extraction")

plot.bar(range(len(indices)), feature_extraction[indices], color =
'darkgreen',align="center")

plot.step(range(len(indices)), npy.cumsum(feature_extraction[indices]),
where='mid', label='Cumulative')

plot.xticks(range(len(indices)), feat_names[indices], rotation='vertical',fontsize=14)

plot.xlim([-1, len(indices)])

plot.show()
```

Upon Feature extraction, we see that the top 5 factors causing spike in attrition is as listed below :

- satisfaction level
- time spent in the company
- last evaluation
- average monthly hours
- project count

This will help the company to take the right measures and help to contain employees from leaving the company.

References :

https://www.datasciencecentral.com/

http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

https://www.youtube.com/user/BCFoltz