Task B

In [1]:

```
import numpy as np
import pandas as pd
import plotly
import plotly.figure_factory as ff
import plotly.graph_objs as go
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import MinMaxScaler
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot

import warnings
warnings.filterwarnings("ignore")
```

In [2]:

```
data = pd.read_csv('task_b.csv')
data=data.iloc[:,1:]
data.head(10)
```

Out[2]:

	f1	f2	f3	y
0	-195.871045	-14843.084171	5.532140	1.0
1	-1217.183964	-4068.124621	4.416082	1.0
2	9.138451	4413.412028	0.425317	0.0
3	363.824242	15474.760647	1.094119	0.0
4	-768.812047	-7963.932192	1.870536	0.0
5	192.093461	-12677.139687	3.456229	0.0
6	-118.048303	-27479.657522	9.407845	1.0
7	-38.987574	3599.957386	2.999579	0.0
8	-31.315888	-15289.241646	3.882981	0.0
9	55.379493	-9020.326833	7.759002	1.0

In [3]:

```
#X = data[['f1', 'f2', 'f3']].values
#print(X)
```

```
X = data.drop(['y'], axis = 1).values
y = data['y'].values

feature_names = np.array(data.drop(['y'], axis = 1).columns.values)
```

In [19]:

```
def get_feature_importance(weights, feature_names):

    abs_weights = np.abs(weights)

    #get the sorting indices
    feature_arg_index = np.argsort(abs_weights)[::-1]

    print('Absolute Weights')
    for i, j in enumerate(feature_arg_index):

        if i == len(feature_arg_index) - 1:
            print(abs_weights[j])
        else:
            print(abs_weights[j], '>>', end =" ")


    #Printing the features importance
    print('Features importance')
    for i, j in enumerate(feature_arg_index):

        if i == len(feature_arg_index) - 1:
            print(feature_names[j])
        else:
            print(feature_names[j], '>>', end =" ")
```

Task 1.1 - Check feature importance using SGDClassifier with log loss

In [23]:

```
from sklearn.linear_model import SGDClassifier

sgd_lr_clf = SGDClassifier(loss='log', n_jobs=1)
sgd_lr_clf.fit(X, y)

get_feature_importance(sgd_lr_clf.coef_[0], feature_names)
```

```
Absolute Weights
62703.613476207865 >> 21919.91499637754 >> 6582.51216742502
Features importance
f2 >> f1 >> f3
```

Observation : Absolute weights/coeficients are proportional to feature importance.
In above task f2 >> f1 >> f3 as per there respective absolute weights vectors values.

Task 1.2 Check feature importance using SGDClassifier with hinge loss

In [26]:

```
sgd_svm_clf = SGDClassifier(loss='hinge', n_jobs=1)
sgd_svm_clf.fit(X, y)

get_feature_importance(sgd_svm_clf.coef_[0], feature_names)
```

```
Absolute Weights
14440.987684835647 >> 5404.760433670973 >> 961.7157488115278
Features importance
f2 >> f3 >> f1
```

Observation : Absolute weights/coeficients are proportional to feature importance.
In above task f2 >> f3 >> f3 as per there respective absolute weights vectors values.

In [27]:

```
def standardize(data):

    #Standardization of data
    scaler = StandardScaler()
    scaler.fit(data)
    return scaler.transform(data)
```

In [28]:

```
standardized_x = standardize(X)
```

Task 2.1 Check feature importance after standardization using SGDClassifier with log loss

In [29]:

```
sgd_lr_clf = SGDClassifier(loss='log', n_jobs=1)
sgd_lr_clf.fit(standardized_x, y)
```

```python
get_feature_importance(sgd_lr_clf.coef_[0], feature_names)
```

```
Absolute Weights
38.73296022323236 >> 9.21878730380539 >> 1.0925554272526696

Features importance
f3 >> f1 >> f2
```

```
Observation : Absolute weights/coeficients are proportional to feature importance.
              After standardizing the columns, now the there is change in feature importance.
              In above task f3 >> f1 >> f2 as per there respective absolute weights vectors v
alues.
```

Task 2.2 Check feature importance after standardization using SGDClassifier with hinge loss

In [30]:

```python
sgd_svm_clf = SGDClassifier(loss='hinge', n_jobs=1)
sgd_svm_clf.fit(standardized_x, y)

get_feature_importance(sgd_svm_clf.coef_[0], feature_names)
```

```
Absolute Weights
38.217762384260716 >> 11.340481233346019 >> 0.440955859256167
Features importance
f3 >> f2 >> f1
```

Observation :
Absolute weights/coeficients are proportional to feature importance.
After standardizing the columns, now the there is change in feature importance.
In above task f3 >> f2 >> f1 as per there respective absolute weights vectors values.