In [2]:
```
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 SGDClassifier
from sklearn.preprocessing import StandardScaler
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
init_notebook_mode(connected=True)
```

In [3]:
```
data = pd.read_csv('task_b.csv')
data=data.iloc[:,1:]
```

In [4]:
```
data.head()
```

Out[4]:

	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

In [5]:
```
data.corr()['y']
```

Out[5]:
```
f1    0.067172
f2   -0.017944
f3    0.839060
y     1.000000
Name: y, dtype: float64
```

In [6]:
```
data.std()
```

Out[6]:
```
f1      488.195035
f2    10403.417325
f3        2.926662
y         0.501255
dtype: float64
```

In [7]:
```
X=data[['f1','f2','f3']].values
Y=data['y'].values
print(X.shape)
print(Y.shape)
```

```
(200, 3)
(200,)
```

What if our features are with different variance

```
* As part of this task you will observe how linear models work in case of data having feautres with different variance
* from the output of the above cells you can observe that var(F2)>>var(F1)>>Var(F3)

> Task1:
    1. Apply Logistic regression(SGDClassifier with logloss) on 'data' and check the feature importance
    2. Apply SVM(SGDClassifier with hinge) on 'data' and check the feature importance

> Task2:
    1. Apply Logistic regression(SGDClassifier with logloss) on 'data' after standardization
       i.e standardization(data, column wise): (column-mean(column))/std(column) and check the feature importance
    2. Apply SVM(SGDClassifier with hinge) on 'data' after standardization
       i.e standardization(data, column wise): (column-mean(column))/std(column) and check the feature importance
```

In [8]:
```python
clfr=SGDClassifier(loss='log',random_state=42)
clfr.fit(X,Y)
```

Out[8]:
```
SGDClassifier(loss='log', random_state=42)
```

In [9]:
```python
clfr1=SGDClassifier(loss='hinge',random_state=42)
clfr1.fit(X,Y)
```

Out[9]:
```
SGDClassifier(random_state=42)
```

In [10]:
```python
##https://machinelearningmastery.com/calculate-feature-importance-with-python/
feature_imp=clfr.coef_[0]
for a,b in enumerate(feature_imp):
    print('Feature: %0d, Score: %.5f' % (a,b))
```

```
Feature: 0, Score: 8252.61713
Feature: 1, Score: -9979.99940
Feature: 2, Score: 10367.64223
```

Make sure you write the observations for each task, why a particular feautre got more importance than others

In [11]:
```
feature_imp1=clfr1.coef_[0]
for a,b in enumerate(feature_imp1):
    print('Feature: %0d, Score: %.5f' % (a,b))
```

```
Feature: 0, Score: -7107.37390
Feature: 1, Score: 9364.07984
Feature: 2, Score: 9088.73594
```

The variance is too high.

The feature 2 has the highest significance of all the three features.

The effect of variance reduces once standardization is done.

F2>F3>F1

Task 2

In [12]:
```
Std = StandardScaler().fit_transform(data[['f1','f2','f3']])
##https://towardsdatascience.com/preprocessing-with-sklearn-a-complete-and-comp
rehensive-guide-670cb98fcfb9
```

In [13]:
```
clfr_SS=SGDClassifier(loss='log',random_state=42)
clfr_SS.fit(Std,Y)
```

Out[13]:
```
SGDClassifier(loss='log', random_state=42)
```

In [14]:
```
clfr_SS1=SGDClassifier(loss='hinge',random_state=42)
clfr_SS1.fit(Std,Y)
```

Out[14]:
```
SGDClassifier(random_state=42)
```

In [15]:
```
feature_imp_SS=clfr_SS.coef_[0]
for a,b in enumerate(feature_imp_SS):
    print('Feature: %0d, Score: %.5f' % (a,b))
```

```
Feature: 0, Score: 2.30695
Feature: 1, Score: 4.39403
Feature: 2, Score: 11.53284
```

In [16]:
```
feature_imp_SS1=clfr_SS1.coef_[0]
for a,b in enumerate(feature_imp_SS1):
    print('Feature: %0d, Score: %.5f' % (a,b))
```

```
Feature: 0, Score: -1.96618
Feature: 1, Score: 2.43060
Feature: 2, Score: 13.54704
```

1.The feature 3 has the highest impact of all the three features after standardization.

2.The standardization reduces the effect of variance among the features

3.f3>f2>f1 this is the order of features importance according to their significance.