1. Show the predictive framework you designed.

Hint: What features do you extract? What algorithms do you use in the framework?

Features: Month、DayofMonth、DayofWeek、CRSDepTime、CRSArrTime、UniqueCarrier、FlightNum、CRSElapsedTime、Origin、Dest

在特徵的部分，因為這次作業要預測2005年航班取消情形，所以能使用的特徵必須是在航班取消之前就能知道的資訊，而諸如DepTime、ArrTime、TaxiIn、TaxiOut等特徵是若航班被取消則這些欄位會固定顯示零或是NA，也就是要等到航班被取消才能知道這些特徵的數值，所以不能被使用；還有因為在訓練用資料(2000至2004年的航班資料)TailNum這個欄位當它為NA時，有很大的機率那次航班會被取消，然後在測試用資料(2005年航班資料)則當航班被取消時，TailNum欄位會顯示零，所以我認為此欄位跟目標欄位有不合理的過高相依性，所以我也沒加入這個欄位資料；而我也沒加入Year的特徵，以及因為Distance這個欄位的數值是Origin以及Dest的距離所以我也沒加入此特徵；而後面幾個欄位，例如：CarrierDelay是跟飛機延遲有關的欄位，所以我也沒加入這些特徵，所以最後所使用的特徵分別是：Month、DayofMonth、DayofWeek、CRSDepTime、CRSArrTime、UniqueCarrier、FlightNum、CRSElapsedTime、Origin、Dest這10個欄位。

Algorithms:

Preprocess: UnderSampling

Model: Logistic Regression、SVM、Decision Tree、Random Forest

演算法的部分，因為有發現到有被取消的航班及未被取消的航班比例是1:41，所以在訓練模型前，會先將訓練資料做undersampling，讓取消的航班及未取消的航班比是1:1。接著我會將資料分別丟入Logistic Regression、SVM、Decision Tree以及Random Forest這四種模型中分別觀察它們對測試資料(2005年的航班資料)的各自對取消及未被取消的航班精確度。

2. Explain the validation method you use.

Hint: Leave-one-out, Holdout, k-fold, or other methods?

Holdout

在這次的驗證資料方式中，我選用了holdout這個方式，將2000至2004年的訓練用航班資料分成8:2，再將這20%資料作為驗證資料，最後把這些資料分別丟入4個模型中(Logistic Regression、SVM、Decision Tree及Random Forest)以檢測結果。

3. Explain the evaluation metric you use.

Hint: Don't just show the prediction results, you should show the effectiveness of your framework using like confusion matrix.

Confusion matrix

Logistic Regression(maxIter=5)

Accuracy: 0.6683

	Precision	Recall	F-score
Uncancelled	0.6651	0.6891	0.6769

Cancelled	0.6717	0.6471	0.6592

```
Validation
[[74984. 33822.]
 [37750. 69214.]]
TP: 0.68915317170009    FP: 0.6470775214090723
Precision(0): 0.6651409512658115        Precision(1): 0.67174579758531
Recall(0): 0.68915317170009     Recall(1): 0.6470775214090723
F-score(0): 0.676934187957028   F-score(1): 0.6591809523809524
Accuracy: 0.6682949436900403
```

SVM(maxIter=15)

Accuracy: 0.6759

	Precision	Recall	F-score
Uncancelled	0.6749	0.6889	0.6818
Cancelled	0.6769	0.6626	0.6697

```
Validation
[[74903. 33832.]
 [36085. 70879.]]
TP: 0.6888582333195383  FP: 0.6626435062263939
Precision(0): 0.6748747612354489        Precision(1): 0.6769011851667924
Recall(0): 0.6888582333195383   Recall(1): 0.6626435062263939
F-score(0): 0.681794805277554   F-score(1): 0.6696964686429668
Accuracy: 0.6758584879855725
```

Decision Tree

Accuracy: 0.5981

	Precision	Recall	F-score
Uncancelled	0.5692	0.8361	0.6773
Cancelled	0.6808	0.3559	0.4674

```
Validation
[[91036. 17848.]
 [68894. 38070.]]
TP: 0.8360824363542853  FP: 0.3559141393365992
Precision(0): 0.569224035515538 Precision(1): 0.680818341142387
Recall(0): 0.8360824363542853   Recall(1): 0.3559141393365992
F-score(0): 0.6773159136056902  F-score(1): 0.4674549673997127
Accuracy: 0.5981338719839887
```

Random Forest(numTrees=3)
Accuracy: 0.5404

	Precision	Recall	F-score
Uncancelled	0.5307	0.7750	0.6300
Cancelled	0.5677	0.3013	0.3937

```
Validation
[[84518. 24542.]
 [74737. 32227.]]
TP: 0.7749679075738126  FP: 0.30128828390860474
Precision(0): 0.5307086119745063        Precision(1): 0.5676865895118814
Recall(0): 0.7749679075738126   Recall(1): 0.30128828390860474
F-score(0): 0.6299908689413563  F-score(1): 0.39365308154128975
Accuracy: 0.5404260637706921
```

4. Show the validation results and give a summary of results.

Result

Logistic Regression(maxIter=5)
Accuracy: 0.6769

	Precision	Recall	F-score
Uncancelled	0.9864	0.6801	0.8051
Cancelled	0.0298	0.5128	0.0564

```
Testting Data
[[4719069. 2220093.]
 [  64898.   68299.]]
TP: 0.6800632410657079  FP: 0.5127668040571485
Precision(0): 0.9864342709721869        Precision(1): 0.02984584808896378
Recall(0): 0.6800632410657079   Recall(1): 0.5127668040571485
F-score(0): 0.8050869354077738  F-score(1): 0.05640841612676636
Accuracy: 0.6769124700824718
```

SVM(maxIter=15)
Accuracy: 0.6793

	Precision	Recall	F-score
Uncancelled	0.9866	0.6825	0.8068
Cancelled	0.0302	0.5160	0.0571

```
Testting Data
[[4736458. 2203596.]
 [  64468.   68736.]]
TP: 0.6824814331415865  FP: 0.516020539923726
Precision(0): 0.9865717571985071        Precision(1): 0.030249100923632637
Recall(0): 0.6824814331415865   Recall(1): 0.516020539923726
F-score(0): 0.8068249839451221  F-score(1): 0.05714817820228007
Accuracy: 0.6793466320612086
```

Decision Tree

Accuracy: 0.8262

	Precision	Recall	F-score
Uncancelled(predicted)	0.9827	0.8376	0.9044
Cancelled(predicted)	0.0268	0.2330	0.0481

```
Testting Data
[[5812176. 1126805.]
 [ 102163.   31029.]]
TP: 0.8376123237691528  FP: 0.23296444230884739
Precision(0): 0.9827262184328629        Precision(1): 0.026799178465997716
Recall(0): 0.8376123237691528   Recall(1): 0.23296444230884739
F-score(0): 0.9043851705240358  F-score(1): 0.0480687453234869
Accuracy: 0.826224839239651
```

Random Forest(numTrees=3)

Accuracy: 0.7854

	Precision	Recall	F-score
Uncancelled(predicted)	0.9818	0.7960	0.8792
Cancelled(predicted)	0.0213	0.2316	0.0391

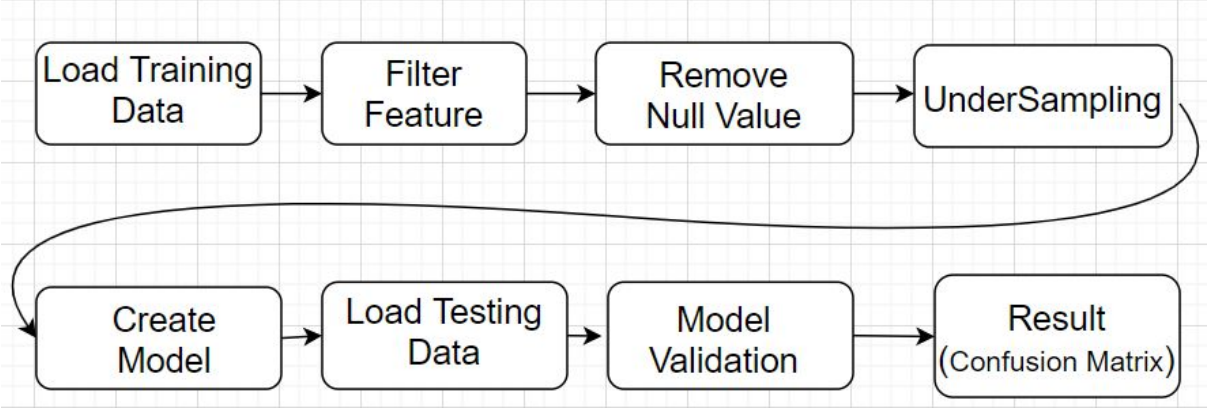
```
Testting Data
[[5524044. 1415445.]
 [ 102361.   30853.]]
TP: 0.7960303705359285  FP: 0.23160478628372394
Precision(0): 0.9818070330877354        Precision(1): 0.021332394845322334
Recall(0): 0.7960303705359285   Recall(1): 0.23160478628372394
F-score(0): 0.8792122550134515  F-score(1): 0.03906649648752273
Accuracy: 0.7853994434659564
```

Summary

在這次的結果中可以發現因為資料不平衡的特性很嚴重，所以造成取消的航班部分精準度都偏低，又這次的航班資料因為能用的特徵資料很有限，且裡面的欄位資料跟航班是否被取消並無過大的相關性，所以即使換了不同的模型預測取消的航班精確度都差不多差，而未被取消的航班則因為數量上的優勢，所以不同的模型出來的結果也都有九成以上的精準度。

在驗證資料中，表現最好的模型是SVM，有67%的精準度；而在預測測試資料中，表現最好的模型是Decision Tree，有82%的精準度

Program workflow:



Execution commands:

```
hadoop fs -put /home/node/03/Downloads/2000.csv
hadoop fs -put /home/node/03/Downloads/2001.csv
hadoop fs -put /home/node/03/Downloads/2002.csv
hadoop fs -put /home/node/03/Downloads/2003.csv
hadoop fs -put /home/node/03/Downloads/2004.csv
hadoop fs -put /home/node/03/Downloads/2005.csv
hadoop fs -ls hdfs:///user/ubuntu
python3 /home/node03/Desktop/hw4.py
```