DATA SCIENCE CASE STUDY **Rishabh Samra**
Jai Kisan Final Year MS Scholar, IIT-M

Given the data related to patient characteristics and reading of various values captured at different timestamps, we need to reduce mortality of patients by providing high risk patients, the limited available ventilators. So this is the Binary Classification problem

DATA CLEANING, PREPARATION AND FEATURE ENGINEERING

Here we have data of patient characteristics in baselines.csv.

Biometric information of patients were available in the form of time series in labs and vitals.csv where each subject is having values of following measures : 's_bp_noninvasive (d)', 'vs_bp_noninvasive (s)', 'vs_hr_hr', 'xp_resp_rate_pt', 'xp_resp_spo2'.

There are 27 columns in the file out of which the **event** column is the label of the data and the 'mrn' column in baselines.csv is the key to join the data with labs and vitals.csv(subject column is common).

There are **no null values** in baselines.csv but out of 708106 rows in labs and vitals.csv, there were **85815 null rows** which were removed and for each measure(5) of each subject we get its minimum, maximum, average, 25th quartile and 75th quartile over all time instants. Thus, for each subject we have 25 values which was merged with baselines.csv

After initially fitting the data, from *baselines.csv* and the mean values from the *labels and trains.csv* into a classifier, we observe the variables from time series data, age and BMI were having higher feature importance. So we tried to introduce more features that can be derived using these variables. Other variables like all symptoms, chest x-ray results etc. were having very less importance so we didn't introduced derived features from it.

We introduced the feature of **weight_status** using BMI where a person is categorized as follows:
Underweight : 0 -18.5
Normal : 18.5-24.9
Overweight : 25-29.9
Obese : 30 or above
Source: https://www.cdc.gov/healthyweight/assessing/

Another feature of subjects' **Age_category** was introduced similarly by dividing given age values into bins and the value depends on his age.

Following are the data of the features like average, minimum, maximum, quantiles(0.25 &0.75) of values of time series data from *labels and trains.csv* :

name	subject	s_bp_noninvasive(d)Avg	vs_bp_noninvasive(s)Avg	vs_hr_hrAvg	xp_resp_rate_ptAvg	xp_resp_spo2Avg
0	655528	61.162459	129.902067	72.801656	30.400445	91.445496
1	729545	58.777900	128.472046	72.903227	29.859476	92.746338
2	805568	60.488792	129.666909	74.767961	29.835982	91.200127
3	895876	59.276564	131.727024	76.640413	30.124052	92.662587
4	905164	59.334065	129.495319	77.028180	28.720727	91.492369

name	subject	s_bp_noninvasive(d)Max	vs_bp_noninvasive(s)Max	vs_hr_hrMax	xp_resp_rate_ptMax	xp_resp_spo2Max
0	655528	65.358869	134.885183	77.408236	33.976935	99.261517
1	729545	62.797679	135.156700	77.364256	31.598743	96.216565
2	805568	64.848642	135.640160	81.120618	34.848911	94.663084
3	895876	62.823134	137.934264	79.806152	33.890384	97.485132
4	905164	62.964323	133.997336	80.893087	34.020580	94.581302

name	subject	s_bp_noninvasive(d)Min	vs_bp_noninvasive(s)Min	vs_hr_hrMin	xp_resp_rate_ptMin	xp_resp_spo2Min
0	655528	55.864589	124.909075	66.680792	27.394691	85.851285
1	729545	54.336709	125.390764	66.978891	28.576244	86.892212
2	805568	53.678706	124.513776	71.122778	23.515832	87.472743
3	895876	55.895045	126.102565	72.864779	25.249702	87.810329
4	905164	57.022347	125.715506	72.893187	24.693358	86.366743

name	subject	s_bp_noninvasive(d)q25	vs_Labp_noninvasive(s)q25	vs_hr_hrq25	xp_resp_rate_ptq25	xp_resp_spo2q25
0	655528	60.126181	127.583502	70.541252	29.451060	89.279046
1	729545	57.523701	127.173381	71.349564	28.860239	91.270237
2	805568	58.543108	127.523197	72.899532	28.268134	90.374399
3	895876	58.194272	130.062418	75.323073	27.880565	90.798072
4	905164	57.610357	128.170852	76.011808	26.965862	90.886343

Thus, we get the combined data initially with 1345 rows and 52 columns which in a transposed way is shown as below :

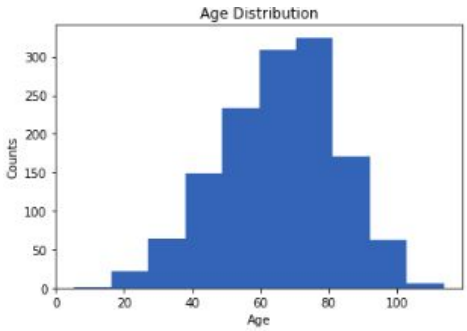
	0	1	2	3	4	5	6	7	8	9	...
Age	62.3217	78.6256	70.4607	59.0431	90.4772	70.6145	72.2871	47.8176	62.987	76.2436	...
sex.factor	Male	Female	Female	Male	Male	Female	Male	Female	Male	Male	...
bmi	20.3886	27.5546	34.1417	19.8333	19.858	28.1329	34.9791	26.8428	33.6863	21.4582	...
hypoxia_ed.factor	No	Yes	Yes	Yes	No	Yes	No	Yes	Yes	No	...
smoke_vape	No	No	No	Yes	No	Yes	No	No	No	No	...
dm.factor	No	Yes	Yes	Yes	Yes	Yes	No	No	No	No	...
htn.factor	Yes	Yes	Yes	No	Yes	No	No	Yes	No	Yes	...
pulm___1.factor	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	...
renal___1.factor	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	...
renal___2.factor	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	...
cad.factor	No	No	Yes	Yes	No	No	No	No	No	No	...
cancer	No	No	No	No	No	No	No	No	No	No	...
any_immunosuppression	unknown/No	unknown/No	unknown/No	unknown/No	unknown/No	unknown/No	unknown/No	unknown/No	unknown/No	unknown/No	...
symptoms___1.factor	Checked	Checked	Checked	Checked	Unchecked	Checked	Checked	Unchecked	Checked	Checked	...
symptoms___2.factor	Checked	Checked	Checked	Unchecked	Checked	Checked	Checked	Checked	Unchecked	Checked	...
symptoms___10.factor	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Checked	Unchecked	Checked	Unchecked	...
symptoms___9.factor	Unchecked	Checked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	...
symptoms___8.factor	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Checked	Unchecked	Unchecked	Checked	...
symptoms___3.factor	Checked	Checked	Unchecked	Unchecked	Checked	Checked	Checked	Checked	Unchecked	Checked	...
first_cxr_results___0.factor	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	...
first_cxr_results___1.factor	Unchecked	Unchecked	Unchecked	Checked	Unchecked	Unchecked	Checked	Unchecked	Unchecked	Unchecked	...
first_cxr_results___2.factor	Checked	Checked	Checked	Unchecked	Checked	Unchecked	Checked	Checked	Checked	Checked	...
first_cxr_results___3.factor	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	Unchecked	...
duration_symptoms	8	20	9	7	25	6	7	9	13	8	...
Ed_before_order_set	No	Yes	No	No	Yes	Yes	Yes	No	No	Yes	...
s_bp_noninvasive(d)Avg	61.1625	58.7779	60.4888	59.2766	59.3341	60.9842	59.8975	61.3979	61.8403	62.031	...
vs_bp_noninvasive(s)Avg	129.902	128.472	129.667	131.727	129.495	129.841	130.217	128.044	131.397	131.425	...
vs_hr_hrAvg	72.8017	72.9032	74.768	76.6404	77.0282	74.0878	74.163	74.796	75.4427	74.6609	...
xp_resp_rate_ptAvg	30.4004	29.8595	29.836	30.1241	28.7207	30.5852	29.6106	29.2002	30.4569	30.0094	...
xp_resp_spo2Avg	91.4455	92.7463	91.2001	92.6626	91.4924	92.469	93.3567	91.8043	91.4818	92.2574	...
s_bp_noninvasive(d)Min	55.8646	54.3367	53.6787	55.895	57.0223	55.1337	54.6553	56.1079	58.3557	56.6959	...
vs_bp_noninvasive(s)Min	124.909	125.391	124.514	126.103	125.716	126.146	125.596	122.366	126.471	126.197	...
vs_hr_hrMin	66.6808	66.9789	71.1228	72.8648	72.8932	71.7302	70.0911	70.9159	70.8906	70.297	...
xp_resp_rate_ptMin	27.3947	28.5762	23.5158	25.2497	24.6934	26.4464	24.3467	25.5078	22.4794	27.1362	...
xp_resp_spo2Min	85.8513	86.8922	87.4727	87.8103	86.3667	86.1681	88.7255	88.059	86.9478	87.4297	...
s_bp_noninvasive(d)Max	65.3589	62.7977	64.8486	62.8231	62.9643	65.962	64.3555	65.2192	64.9114	66.733	...
vs_bp_noninvasive(s)Max	134.885	135.157	135.64	137.934	133.997	133.048	133.014	134.984	136.69	135.552	...
vs_hr_hrMax	77.4082	77.3643	81.1206	79.8062	80.8931	76.6721	77.6708	79.2974	81.8486	77.8882	...
xp_resp_rate_ptMax	33.9769	31.5987	34.8489	33.8904	34.0206	36.5576	34.5253	31.7391	37.1341	32.9618	...
xp_resp_spo2Max	99.2615	96.2166	94.6631	97.4851	94.5813	97.1836	98.0858	95.9751	95.0193	97.7139	...
s_bp_noninvasive(d)q25	60.1262	57.5237	58.5431	58.1943	57.6104	59.4003	58.5657	59.8503	61.0056	60.6132	...
vs_Labp_noninvasive(s)q25	127.584	127.173	127.523	130.062	128.171	128.508	129.035	125.286	129.709	130.427	...
vs_hr_hrq25	70.5413	71.3496	72.8995	75.3231	76.0118	72.9591	72.8506	73.6034	72.794	72.7481	...
xp_resp_rate_ptq25	29.4511	28.8602	28.2681	27.8806	26.9659	28.7886	28.1312	28.2886	27.2191	28.8402	...
xp_resp_spo2q25	89.279	91.2702	90.3744	90.7981	90.8863	90.0367	90.872	89.8985	90.2719	90.0815	...
s_bp_noninvasive(d)q75	62.2828	60.0527	62.2774	60.5047	59.7591	62.7205	61.5788	62.7767	62.7384	63.5034	...
vs_Labp_noninvasive(s)q75	132.472	129.279	131.558	134.194	130.332	131.369	131.567	129.261	132.773	132.955	...
vs_hr_hrq75	74.7129	74.6546	77.1719	77.7612	77.9889	75.0361	75.4466	75.899	77.7014	76.6353	...
xp_resp_rate_ptq75	31.5046	30.6448	31.3946	32.2995	30.4311	31.5316	31.2637	30.0706	33.8851	31.2925	...
xp_resp_spo2q75	92.4888	94.4443	91.8506	94.3505	92.8647	95.1799	95.7457	93.497	92.8303	94.9155	...
weight_status	1	2	3	1	1	2	3	2	3	1	...
Age_category	2	2	2	1	3	2	2	1	2	2	...

EXPLORATORY DATA ANALYSIS

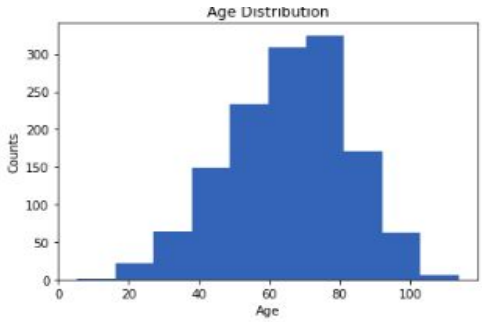
	subject	Age	bmi	duration_symptoms	event
count	1.345000e+03	1345.000000	1345.000000	1345.000000	1345.000000
mean	5.518217e+07	65.474174	27.844956	8.896654	0.479554
std	3.899901e+07	17.223304	6.513850	5.154361	0.499768
min	6.555280e+05	5.408467	9.861328	1.000000	0.000000
25%	6.467320e+06	54.153658	23.596934	5.000000	0.000000
50%	7.670870e+07	66.963905	27.237496	9.000000	0.000000
75%	9.007959e+07	77.862961	30.961962	11.000000	1.000000
max	9.028996e+07	113.674338	58.904689	35.000000	1.000000

These are the variables in baselines.csv which are continuous(Except subject which is categorical). Here we observed one data of Age having negative value which was replaced with the mean value so we have minimum value of age = 5 **(Outlier Removal)**

Before Conversion

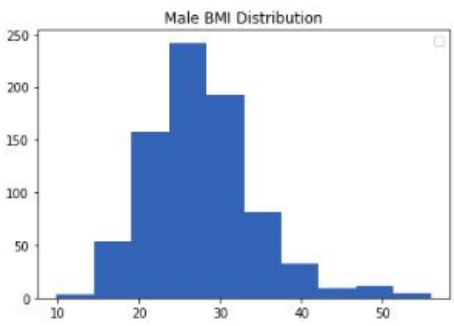


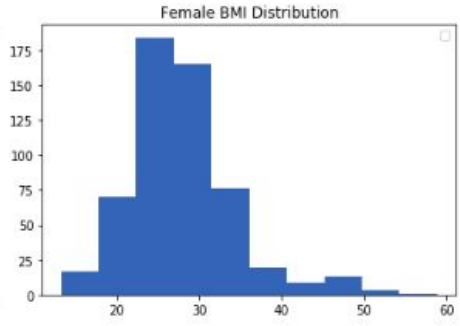
After Conversion

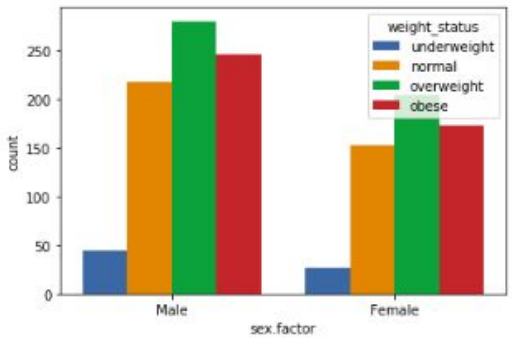


	sex.factor	event	count
0	Female	No	296
1	Female	Yes	261
2	Male	No	404
3	Male	Yes	384

% of Males requiring Ventilators : 48.8%
% of Females requiring Ventilators : 46.8%

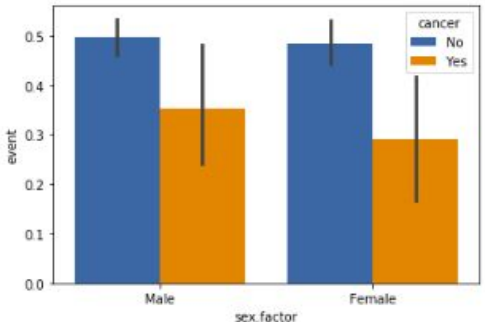






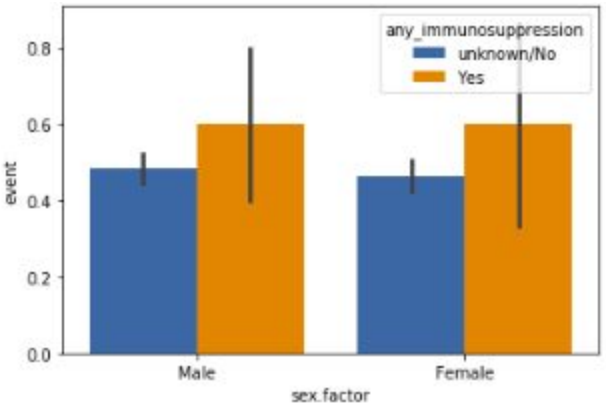
INFLUENCE OF DIFFERENT FEATURES ON VENTILATOR REQUIREMENTS OF PEOPLE

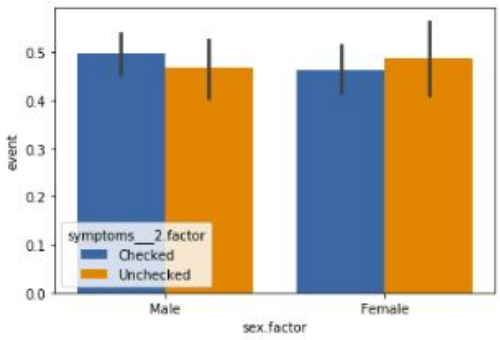
In the barplots, the Y-axis denotes the fraction of people who will require a ventilator
We can infer that there is more chance of assigning ventilators to patients not having cancer.
Also males are having a higher chance of using ventilators as compared to females.

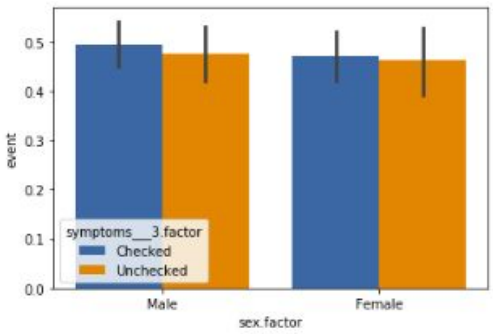


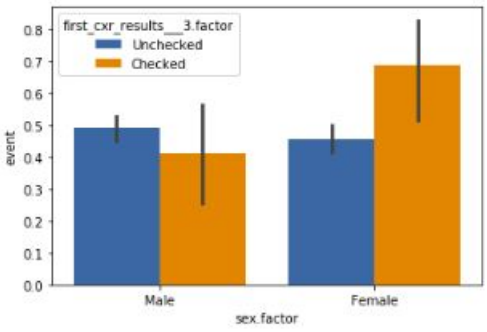
	cancer	event	count
0	No	No	631
1	No	Yes	612
2	Yes	No	69
3	Yes	Yes	33

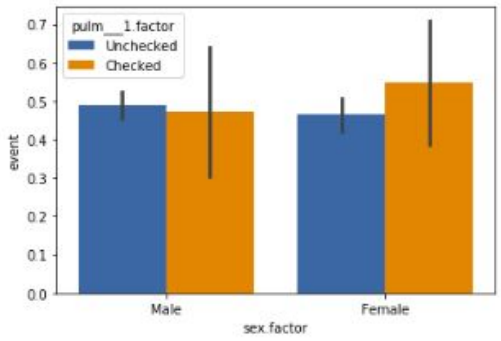
	any_immunosuppression	event	count
0	Yes	0	16
1	Yes	1	24
2	unknown/No	0	684
3	unknown/No	1	621











	symptoms___2.factor	pulm___1.factor	event	count
0	Checked	Checked	0	24
1	Checked	Checked	1	24
2	Checked	Unchecked	0	459
3	Checked	Unchecked	1	425
4	Unchecked	Checked	0	9
5	Unchecked	Checked	1	10
6	Unchecked	Unchecked	0	208
7	Unchecked	Unchecked	1	186

USING STATISTICAL METHODS TO MAKE INFERENCES

As explained before using the given base data which was merged and trained using a Random Forest Classifier to get the important features that affect the prediction. Some of the features are shown below in descending order of their importance:

It is clearly visible that some features like Age, BMI and other features derived from the time-series data of *labs and vitals.csv* has more feature importance than others. So we increase more features related to them as described above. Also from the feature importance data as well as from discussion with a **Pathologist**, it can be inferred that some features like Cough, fever, nausea, myalgias, renal_factor etc. does not affect the ventilator allotment much so I did not proceed further to derive more features from them.

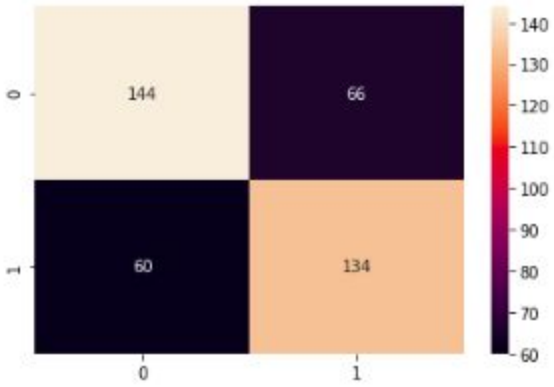
```
Age = 0.14005595972902446
vs_hr_hrAvg = 0.10721001079334053
bmi = 0.08301666013317895
s_bp_noninvasive(d)Avg = 0.06087274892042242
vs_hr_hrMax = 0.0517958638109836
s_bp_noninvasive(d)Max = 0.05156391235536019
vs_hr_hrMin = 0.0504472053768748
vs_bp_noninvasive(s)Avg = 0.041655270702802574
s_bp_noninvasive(d)Min = 0.0344236808317227
vs_bp_noninvasive(s)Max = 0.033618595907344936
xp_resp_spo2Max = 0.03326605848565366
xp_resp_rate_ptMin = 0.03256188458659361
xp_resp_spo2Min = 0.0322572000957547
vs_bp_noninvasive(s)Min = 0.031679695339232775
xp_resp_rate_ptAvg = 0.031291664206053685
xp_resp_rate_ptMax = 0.030728959223710453
xp_resp_spo2Avg = 0.030708171359965387
duration_symptoms = 0.02863942838660108
symptoms___1.factor_Unchecked = 0.006314873604798945
smoke_vape_Yes = 0.006212325805061753
dm.factor_Yes = 0.005740763536415627
htn.factor_Yes = 0.005381144651218526
symptoms___3.factor_Unchecked = 0.005087031708128262
symptoms___2.factor_Unchecked = 0.004722779891086653
```

SELECTING ML CLASSIFIERS FOR PREDICTING ***EVENT***

Given our dataset has a large number of categorical variables, so the use of logistic regression is not favoured.
The important evaluation metric is False Negative(FN) which is a measure of the number of patients who require a ventilator in reality, but wasn't predicted. However, it is also desirable to decrease the number of False Positives(FP) as it is a wastage of resources to allocate a ventilator to a patient who didn't require it.

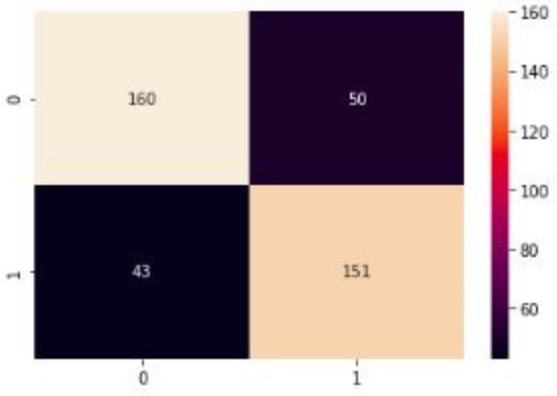
So initially I use the data(test_size=0.3) with only given features to train a **Decision Tree** and got the following results for the test data:

	precision	recall	f1-score	support
0	0.71	0.69	0.70	210
1	0.67	0.69	0.68	194
accuracy			0.69	404
macro avg	0.69	0.69	0.69	404
weighted avg	0.69	0.69	0.69	404



TRAINING WITH A RANDOM FOREST CLASSIFIER

	precision	recall	f1-score	support
0	0.79	0.76	0.77	210
1	0.75	0.78	0.76	194
accuracy			0.77	404
macro avg	0.77	0.77	0.77	404
weighted avg	0.77	0.77	0.77	404



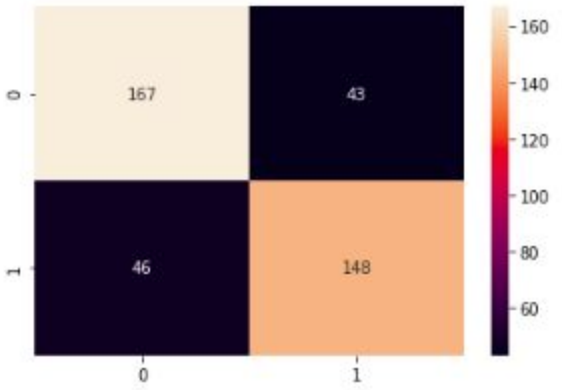
Random Forest Classifier is a type of ensemble method where results from different decision trees trained parallely on the subset of datasets are combined resulting in a stronger classifier.

We can see from the results of the classifier that there is an improvement in the accuracy and most importantly in the False Negative metric of the classifier(from 66 to 50).

I saw the important features as mentioned above and thus added a few features like Weight_Status, Age_Category, Quantile(0.25) and Quantile(0.75) of the values of time_series data, for improving results.

TRAINING A RANDOM FOREST CLASSIFIER AFTER ADDING FEATURES:

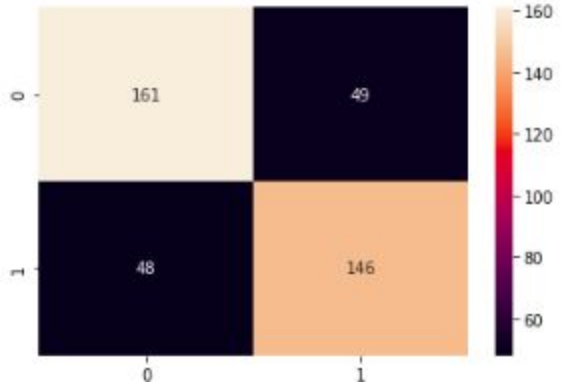
	precision	recall	f1-score	support
0	0.78	0.80	0.79	210
1	0.77	0.76	0.77	194
accuracy			0.78	404
macro avg	0.78	0.78	0.78	404
weighted avg	0.78	0.78	0.78	404



We can observe there is a little increase in the accuracy and also the False Negatives decrease from 50 to 43. Thus there is a bit of improvement in the results.

REMOVAL OF A FEW FEATURES LIKE Symptoms_1,8,9,10, initial Cxr_Results WHICH ARE HAVING LESS IMPORTANCE AS WELL AS SUGGESTED BY A PATHOLOGIST

	precision	recall	f1-score	support
0	0.77	0.77	0.77	210
1	0.75	0.75	0.75	194
accuracy			0.76	404
macro avg	0.76	0.76	0.76	404
weighted avg	0.76	0.76	0.76	404

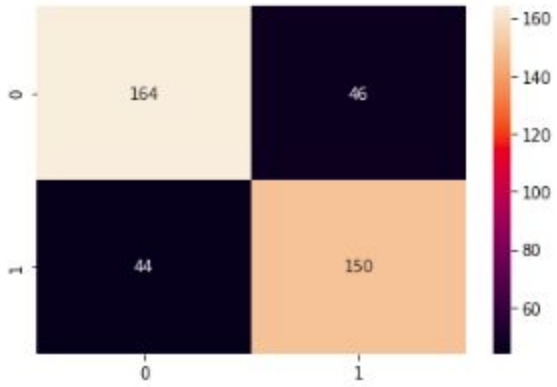


HYPERPARAMETER TUNING OF THIS DATA(Using GridSearch method)

Grid parameter values: [**'n_estimators'**: [100,300,500,600,700,900], **'min_samples_split'**: [2,5,15,50], **'max_depth'**: [None, 3,5,7,9], **'min_samples_leaf'**:[1,2,5,10]]

We get the Best estimator to have 700, 5, None, 1 values respectively and performance improves a bit.

	precision	recall	f1-score	support
0	0.79	0.78	0.78	210
1	0.77	0.77	0.77	194
accuracy			0.78	404
macro avg	0.78	0.78	0.78	404
weighted avg	0.78	0.78	0.78	404



CONCLUSION :
There is a bit of increase in False Negatives and a bit decrease in accuracy as compared to the previous case when features were not removed. But Hyperparameter tuning gives better results**.**

GRADIENT BOOSTING ALGORITHM: CATBOOST

Gradient Boosting is a type of boosting algorithm which is another ensemble method of training a series of models sequentially instead of parallely as in the case of Bagging.
Catboost is one of the algorithm that does gradient boosting and has following advantages:

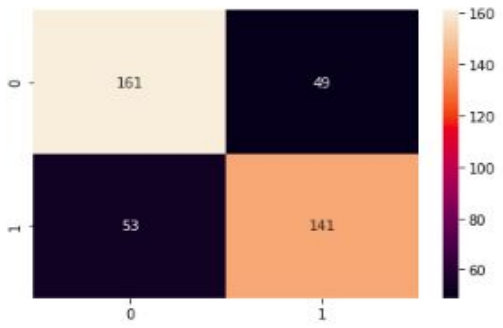
1. Fast inference as it uses symmetric trees.
2. Supports sophisticated categorical features.
3. Boosting scheme helps to reduce overfitting.

TRAINING A CATBOOST CLASSIFIER

Test_Size = 0.3

	precision	recall	f1-score	support
0	0.75	0.77	0.76	210
1	0.74	0.73	0.73	194
accuracy			0.75	404
macro avg	0.75	0.75	0.75	404
weighted avg	0.75	0.75	0.75	404

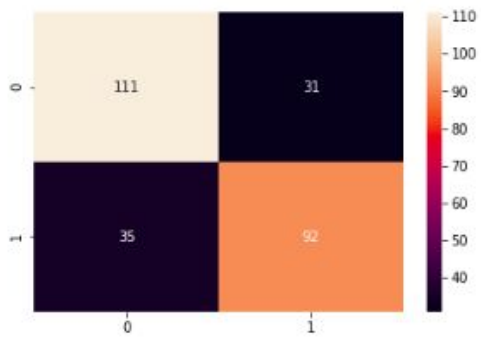
<matplotlib.axes._subplots.AxesSubplot at 0x7f2344522590>



Test_Size = 0.2

	precision	recall	f1-score	support
0	0.76	0.78	0.77	142
1	0.75	0.72	0.74	127
accuracy			0.75	269
macro avg	0.75	0.75	0.75	269
weighted avg	0.75	0.75	0.75	269

<matplotlib.axes._subplots.AxesSubplot at 0x7f2344452896>



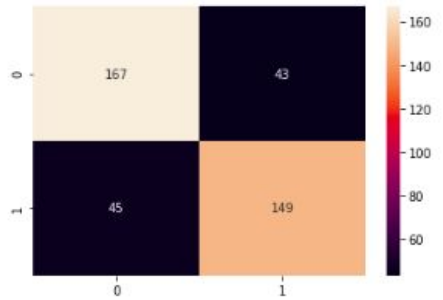
HYPERPARAMETER TUNING

Grid = {'iterations': [10,100,500],'learning_rate': [0.03, 0.1],'depth': [4, 6, 9],'l2_leaf_reg': [1, 3, 5, 7, 9]}

Best Parameter Values: 500, 0.03, 6, 3 respectively.

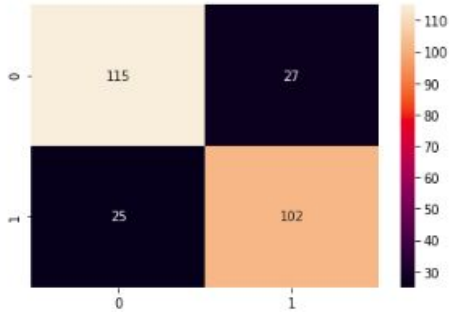
	precision	recall	f1-score	support
0	0.79	0.80	0.79	210
1	0.78	0.77	0.77	194
accuracy			0.78	404
macro avg	0.78	0.78	0.78	404
weighted avg	0.78	0.78	0.78	404

<matplotlib.axes._subplots.AxesSubplot at 0x7f23444aecd0>



	precision	recall	f1-score	support
0	0.82	0.81	0.82	142
1	0.79	0.80	0.80	127
accuracy			0.81	269
macro avg	0.81	0.81	0.81	269
weighted avg	0.81	0.81	0.81	269

<matplotlib.axes._subplots.AxesSubplot at 0x7f234436f19



Here, we got better results for test_size = 0.20 as compared to that of 0.30

MODELS COMPARISON RESULTS

Test_Size: 0.3	PRECISION	RECALL	F1-SCORE	ACCURACY
Decision Tree	0.69	0.69	0.69	0.69
Random Forest	0.78	0.78	0.78	0.78
CatBoost	0.78	0.78	0.78	0.78

Test_Size: 0.2	PRECISION	RECALL	F1-SCORE	ACCURACY
Decision Tree	0.73	0.73	0.73	0.73
Random Forest	0.80	0.80	0.80	0.80
CatBoost	0.81	0.81	0.81	0.81

DISCUSSION:

After visualizing the data and calculating the features' importance, it was found that adding more features connecting the important features increases the evaluation metric of the model. Removing less important features can result in a bit decrease of accuracy. We also found that a Gradient Boosting method like Catboost performs better than Random Forests because of the advantages of it as discussed above. Also random forests performed better than decision trees because it takes the combination of several trees, hence is a strong classifier. We also observed that we get far better results when we decrease the test size to 0.2 from 0.3 while doing train_test split.