## JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR) An International Scholarly Open Access, Peer-reviewed, Refereed Journal

# **Airline Flight Delay Prediction System**

Krushna Mudigonda<sup>1</sup>, Sonal Jagtap<sup>2</sup>, Ravina Asole<sup>3</sup>, Kalyani Sisodiya<sup>4</sup>, Kartik Mirkhale<sup>5</sup>

Department of E&TC, SKNCOE, SPPU, Pune

<sup>1</sup>krushnamudigonda99@gmail.com, <sup>2</sup>skjagtap.skncoe@sinhgad.edu,

<sup>3</sup>ravina.asole.skncoe@sinhgad.edu, <sup>4</sup>kalyamisisodiya829@gmail.com, <sup>5</sup>kartikmirkhale2000@gmail.com

*Abstract* — It is essential for both passengers and airlines to anticipate flight delays because they not only result in significant financial losses but also damage the reputation that has been developed over many years and cost customers valuable time. By utilising the data that is currently available, this dataset seeks to estimate the arrival delay of a scheduled individual flight at the destination airport. The prediction model proposed in this paper uses supervised machine learning methods to forecast airline arrival delays. We use a total of five methods to forecast the precise flight delay. Each algorithm's performance is examined. The model is provided flight data as well as meteorological data. With this information, each algorithm makes a specific prediction.

Keywords-Flight delay prediction, Airline Transport, Machine learning, Classification Algorithms, Data Analytics

## I. INTRODUCTION

Because air travel is so important to the industry's economy, airlines and airports must raise the caliber of their offerings. A major problem for airports and airline companies nowadays is flight delays. Also, passengers are concerned when a flight is delayed, and the agency and the airport experience additional costs as a result. Flight delays cost the U.S. government between 31 and 40 billion dollars in 2007. 76% of the flights landed on time in 2017.

The proportion of on-time flights dropped by 8.5% from 2016 to 2017. The following can be listed as some of the causes of flight delays: safety, weather, part shortages, technical difficulties with aircraft equipment, and delays in the flight crew. Unavoidable flight delays have significant financial repercussions for travelers, airlines, and airports. Moreover, delays can damage the environment by causing higher fuel costs and harmful gas emissions.

## **II. LITERATURE SURVEY**

N. K. Singh and K. R. K. Rao. provides a comprehensive overview of the existing research in airline flight delay prediction using ML techniques. The authors review various ML models, feature selection techniques, and evaluation metrics used in the literature. They also discuss the challenges and future directions of this research area. A. H. Alazzawi et al. focuses on the application of ML techniques in predicting flight delays. The authors review the various ML models, data sources, and features used in the literature. They also discuss the limitations and challenges of the current research and suggest future research directions. *M. Ghasemi et al.* provides a systematic analysis of the existing research in airline flight delay prediction using ML techniques. The authors review the various ML models, data sources, and features and limitations of the current research and provide suggestions for future research. *A. Kumar et al.* focuses on the application of ML techniques in predicting flight delays in the airline industry. The authors review the various ML models, data sources, and features used in the literature. They also discuss the challenges and future directions for future research. *A. Kumar et al.* focuses on the application of ML techniques in predicting flight delays in the airline industry. The authors review the various ML models, data sources, and features used in the literature. They also discuss the challenges and future directions of this research area.

## III. METHEDOLOGY

This project could be relevant in field applications of machine learning such as surveys, monitoring and studies in maintaining a good ecosystem for the Airline field. It is also applicable in Train , bus & other transportation system for Delay prediction. The aim of this research is to predict flight delays, which are the highest economy-producing field for many countries, and among many transportation modes, this one is the fastest and most comfortable, so identifying and reducing flight delays can dramatically reduce flight delays and save huge amounts of money by using machine-learning algorithms.



Fig.1 Data for prediction process

#### **IV. METHODOLOGY DESCRIPTION**

All the data manipulation or dropping together happens before it is used, so the dataset is cleaned up in the pre-processing Process. All the representation of the dataset is in the form of graphics. So, it is efficient to communicate in this process. Parameter Selection for getting proper configuration model information from the given data. The parameters are mean (mu) and standard deviation (sigma). All the processed data is usable for building the project model. With the help of algorithms, our model can learn, test, and predict the features of the data. The prediction process depends on the model. So, when the output of an algorithm comes from a dataset and is applied to new data to forecast the likelihood of a particular outcome, The user sends input to the modelling dataset for prediction. A total of five algorithms are going to be used for the prediction purpose. Random forest, logistic regression, decision tree, naive bayes, and K-nearest neighbors are going to be implemented. All algorithms perform the operation and give their particular output. It predicts the accuracy of a flight delay with its proper classification report and final status. The user sends input to the modelling dataset for prediction, inputs like flight numbers, timings, etc. If the flight is delayed, then it shows a "Delay prediction." And if the flight is on time, then it shows "Flight on Time" to the application users.

We will upload our dataset into application. The quality of the data should be checked before applying our algorithms. Transforming raw data into numerical features that can be processed while preserving the information in the original data set. After extractingfeatures from the dataset we will generate our algorithms on that dataset. We will plot the accuracies comparison graph between all the algorithms.

## V. EXPERIMENTATION

In Simulation,

The performance of the classifier can be calculated from the confusion matrix. After being compared to the actual result, the classifier results can generate four values namely (TP), (TN), (FP), (FN). Four measures were used to measure the performance of selected algorithms: accuracy, precision, recall, and F1 score. These measures are all positively related to the quality of algorithms. Consequently, for a specific algorithm, the higher values of these measures are, the better their performances are. The value of the four measures can be obtained by calculations using these parameters:

Where,

Accuracy = TP+TN / TP+TN+FP+FN Precision= TP / TP+FP Recall= TP / TP+FN F1-Score = 2 × (Recall × Precision) / (Recall + Precision)

True Positive (TP) True Negative (TN) False Positive (FP) False Negative (FN)

All algorithms perform the operation and give their particular output. It predicts the accuracy of a flight delay with its proper classification report and final status. The user sends input to the modelling dataset for prediction. inputs like flight numbers, timings, etc. If the flight is delayed, then it shows a "Delay prediction." And if the flight is on time, then it shows "Flight on Time" to the application users.

A. Flowchart

We will upload our dataset into application. The quality of the data should be checked before applying our algorithms. Transforming raw data into numerical features that can be processed while preserving the information in the original data set. After extractingfeatures from the dataset we will generate our algorithms on that dataset. We will plot the accuracies comparison graph between all the algorithms.



Fig.3 Accuracy Graph

Accuracy is calculated using comparison of all the algorithms based on graph. The final prediction of every algorithm is shown in fig 3. It determines that which algorithm gives the highest accurate prediction.

## V. EXPERIMENTAL RESULTS

## A. Random Forest :

Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset.

💭 Jup									the tay and	10 CM
	yter Flight_D	elay_Prec	diction Last	Checkpoint	Last Tuesd	ay at 1.52 PM	(autosaved)		n Logout	
File	Edit View Ins	ert Cell	Kernel	Widgets	Help			Trusted	Python 3 (pykernel) O	
n +	3× € € ↑		■ C #	Code	Y	602				
	Machin	e Learn	ing Mod	el						
	Random for	rest Classifie	er							
I/	[29]: H from s	klearn.ense	mble impor	t RandomFo	orestClas	sifier				
	model1 model1	<pre>#RandomFore .fit(X_trail</pre>	estClassifi in,y_train)	er(criter)	ion="gini	",max_depth	=20)			
	Out[29]: Random	ForestClass	sifier(max_	depth=20)						
D	[30]: M y_pred	ict1=model1	.predict(X	_test)						
D	[31]: H # Calc from s accura	ulate Accur klearn.metr cyscorel=ac	nicy Score	confusion re(y_predi	n_matrix, ict1,y_te	accuracy_sc st)	ore,classification_report			
	priot(	"Accuracy 5	score 1s :-	", accuracy	yscore1) 932					
	fast. H select	televeldi.		- T closed						
D	classi	fication Re	eport	e ',class	precisio	n recall	f1-score support			
		0	0.83	0.96	0.89	202519				
		1	0.89	0.62	0.73	103386				
	ac Bac	ro avg	0.86	0.79	0.85	305905				
	and TELLC	en avg	0.03	0.03	0.04	303903				
7	(aa) H mecon	fusion wate	iv/v test	v predicti	1)				_	-
			Q Sear	chi	1		) 🚍 💼 🗮 🔺 🍭	🧿 🚈 💽	^	IN 9
	_					_				
Select or crea	e a r 🗙 🥚 Flight [	Delay_Prediction	n - Aupyter X	+					· · ·	a (
	wter Flight D	eney_rredicts	tiction	Checken	Last Tur - 1	au at 1 53 P**	laderandi		- Ui 16 1	
Ele	Edit View in	ert Cor	Komel	Wirknots	Halo	ay at 1.52 PM	(annound)	Trusted	Potton 3 (indemni)	
8 +	× 2 15 +		C H	Code	~				( . ) - an a (b) manual O	
1	weight	ed avg	0.85	0.85	0.84	305905				
1	[33]: M cm+con	fusion_matr	rix(y_test.	y_predict:	1)					
	sns.he	atmap(cm,ar	nnot=True, f	mt='d')						
	-Axess	wohrof ( >		_						
							- 175000			
							150000			
	0 -	19	9382		813		720000			
							- 125000			
	_		_				- 100000			
							- 75000			
		39	1020				- 75000			
	н.						- 75000 - 50000			
							- 75000 - 50000 - 25000			
	r.	39	020		6436	6	- 75000 - 50000 - 25000			
	-	39	020		6436	6	- 75000 - 50000 - 25000			
	ri - Logistic Re	39 egression	0		6436 1	6	- 75000 - 50000 - 25000			
I	rd - Logistic Re [34]: Hi from s	39 gression klearn.line	020 0 ear_model s	mport Log	6436 1 isticRegr	6 ression	- 7500 - 50000 - 25000			
р	Logistic Re [34]: H from s	39 egression klearn.line	o o car_model s	mport Log	6436 1 isticRegr	ession	- 75000 - 50000 - 25000	<u>후</u> 핏 <u>C</u>	^	ENG 👳
In	Logistic Re [34]: H from s	39 gression kleam.line	020 0 ear_model_s	mport Log: rch	6436 1 isticRegr	ession	- 75000 - 50000 - 25000	9 v C	^	ENG IN
п	Logistic Re [34]: H free S	39 gression klearn.line	0 0 ear_model s	mport Log tch	6436 i isticRegr	ession	- 3000 - 3000 - 2500	<u> 9 5 6</u>		ENG 🗢

Expected outcome - 1) Random forest -

Random Forest output -0.8458 = 84.58% Accuracy with classification Report

In this Fig.3 we observed that we got 84.58% accuracy by the Random Forest algorithm. And it shows the classification report of the output.

## B. Logistic regression :

It is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables. Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.Logistic Regression is much similar to the Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.

## www.jetir.org (ISSN-2349-5162)

G	(i) localhost:8888/notebool	s/Flight_Delay_Prediction.ipynb	AN CG 5	6 E		syncing 🎧			
	💭 jupyter 🛛	light_Delay_Prediction Last Checkpoint: Last Tuesday at 1:52 PM (autosaved)	n Cogout						
	File Edit Ve	w Insert Cell Kernel Widgets Help Tru	usted Python 3 (ipyke	ernei) O					
	B + × 2	B ↑ ↓ ► Run ■ C ≫ Markdown ∨ □							
	6	gistic Regression							
	In [34]:	from sklearn.linear_model import LogisticRegression model2.rlogisticRegression() model2.rlfLtk.train.y_train)							
		C:\Users\Krushna\anaconda3\llb\site-packages\sklearn\linear_model\_logistic.py:814: ConvergenceWarnin erge (Status=1): 5700: TOTAL NO. of TIERATIONS REACHED LIMIT.	ng: lbfgs failed to co	onv					
		<pre>Increase the number of iterations (max.iter) or scale the data as shown in: https://scikitlemon.org/stable/modules/preprocessing.html Please also refer to the documentation for alternative solver options: https://scikitlemon.org/stable/modules/linear_model.htmlalogistic-regression iter_i = _cdek_optimize_result(</pre>							
	Out[34]	LogisticRegression()							
	In [35]: 1	y_predict2=model2.predict(X_test)							
	In [36]: 1	<ul> <li>doindist Accuracy score</li> <li>fram Allarm wattrix import confusion matrix, accuracy_score, classification_report accuracyscore3/accuracy_score(y_predict2,y_test)</li> <li>print(faccuracy score is i= securacyscore3)</li> </ul>							
		Accuracy Score is :- 0.8379987250943921							
	In [37]: 1	print("Classification Report ", classification_report(y_test,y_predict1))							
		Classification Report precision recall f1-score support							
		0 0.83 0.96 0.89 202519							
		1 0.89 0.62 0.73 103386							



Fig.5 Logistic Regression output

Expected outcome - 2) Logistic Regression

Logistic Regression output -0.8379 = 83.79% Accuracy with Classification Report

In this Fig.5 we observed that we got 83.79% accuracy by the Logistic Regression algorithm. And it shows the classification report of the output.

## C. Decision tree :

is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

	localhost.8888/notebooks/Flight_Delay_Prediction.ipynb	A <sup>a</sup> CD	THE THE LINE SHO
	Upyter Flight Delay Prediction Last Checkment Last Tuesday at 152 PM (autosaved)	2	loont
	Die Die Mars Instand der Marst Mittale Lieb		
	File Call View listen Cell Kemel Widgets nep	Python 3 (g	pykerner) O
	Dicision Tree Classifier		
	In [39]: N from sklearn.tree import DecisionTreeClassifier model3-model3-model3-model3-model3-model3-filt(X train, ~ yrain) model3-filt(X train, ~ yrain)		
	Out[39]: DecisionTreeClassifier(criterion='entropy', max_depth=2, random_state=0)		
	In [40]: W v predict3=model3.predict(X test)		
	in [ai]: n & cuccute cucrety score from slavem.metrics import confusion_matrix,accuracy_score,classification_report accuracyscore):accuracy_score(j_repd(cl.3,_clest) print(Acuracy_score is :-',accuracyscore))		
	Accuracy Score is :- 0.8362465471306452		
	<pre>In [42]: M print("Classification Report ",classification_report(y_test,y_predict3))</pre>		
	Classification Report precision recall f1-score support		
	0 0.82 0.97 0.89 202519 1 0.90 0.58 0.71 103386		
	accuracy 0.84 305005		
	macro avg 0.86 0.77 0.80 305905 weighted avg 0.85 0.84 0.83 305905		
	<pre>In [43]: M cmwconfusion_matrix(y_test,y_predict3)</pre>		
	Out[d3]: <axessubnlot:></axessubnlot:>		
34°C Sunny	🚦 Q Search 🛛 📶 📾 🛤 🚔 🚣 🏘 🌻 🏹 🛛	0	^ ENG @3 41 ➡ 2
34°C Sunny	🚦 Q Seach 🛛 🙇 🖻 🖻 💼 🗮 🛓 🌻 🦻 🏹	<u>e</u>	^ <sup>ENG</sup> 10 03 01 ■ 2
34°C Sunny	🚆 Q. Sanch 🛛 🎉 🔎 🖻 🖻 💼 🚔 🎍 🧐 💱	Q	∧ <sup>ING</sup> Ø3 Ø1 ∎ <sub>2</sub>
34°C Sunny	🏭 Q South 🥂 🎉 📾 🖻 🖻 💼 📥 🏚 🌞 🤴 🤴 🟹	ē	^ ING 623 ⊄I∎⇒ ₂
34°C Sanny	📲 Q. Sewich 🥣 🎉 🔎 📾 📼 📥 🗛 👰 👰 🏹 🛛	<u>e</u>	^ <sup>ENG</sup>
34°C Sunny	Inger Steector mana et x 🖉 Right, Deng, Andreson - Appel x 🕂	<b>2</b>	NG Q QI ■ 2
34°C Sutray	Inter Sector context = x Sector context = x	2 * G	∧ ING @ di ● 2 IN @ di ● 2
34°C Suriny I I ⊂ Home - C	Classest	2 * 0 •	∧ ING Q di ● 2 IN Q di ● 2
34°C Suriny		C A <sup>A</sup> L C A A C A A A A A A A A A A A A A A A	<ul> <li>► CHAG (Q) 41 €0 2</li> <li>► Content</li> <li< td=""></li<></ul>
34°C Surrey	Control	e A Ca dia I Python 3 (python I I Python 3 (python I Python 3 (python 3 (py	Out () () () () () () () () () () () () ()
3rC Suny I ◯ Home C	Imple:         Op. Search         Imple:         Imp	e A Ca e ind   Python 3 (python 3 (p	CHO (3 01 80 2     ))))))))))))))))))))))))))))))))
3rc Sunny I ○ Home C (	Physic Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X         Physic/Looky/Prediction : Apple: X         Image: Selector means at X <th< td=""><td>A The second sec</td><td>∧ <sup>(1)</sup><sub>N</sub> (2) of an <sub>2</sub> ↓ (1) → (1)</td></th<>	A The second sec	∧ <sup>(1)</sup> <sub>N</sub> (2) of an <sub>2</sub> ↓ (1) → (1)
34°C Sunny - C (	Image:         Image:<	C	A DAG (C) OI BO 2
14°C Sourny Home C	Progre         Sector         Sector<	A <sup>A</sup> Li · Č And  Python 3 (python	<ul> <li>Model and American America</li></ul>
JIC Surry I □ Home - C (	They Select rough 1         Fight Select rough 2         Image 3         Image 3 <t< td=""><td>A The second sec</td><td>∧ DG (0) cli ● , ∴ (0) cli ● , ∴</td></t<>	A The second sec	∧ DG (0) cli ● , ∴
JIC Surry I □ Home - C (	Image: Security         Image: Sec	A La A	∧ <sup>DG</sup> <sub>R</sub> (0) cl ■ 2
1 C Home	Image: Second	A CL C I Pythen 3 appr	<ul> <li>Model and American Ameri American American A</li></ul>

Q Searc 🚮 💷 💿 📟 💼 🚞 ENG 🚓 🕫 📼 0634 PM 🥥

Fig.6 Decision Tree output

Expected outcome – 2) Decision Tree

Decision Tree output -0.8362 = 83.62% Accuracy with Classification Report

In this Fig.6 we observed that we got 83.62% accuracy by the Decision Tree algorithm. And it shows the classification report of the output.

## D. K-Nearest neighbour :

K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

	cs/Flight_Delay_Predict	tion.ipynb						AN CQ 1	\$ £ 6	۵ (N
🔵 jupyter 🛛	light_Delay_Pre	diction Last Cher	ckpoint: a minute a	go (autosaved)				<b>.</b>	Logout	
File Edit Vi	ew Insert Cell	Kernel Widg	gets Help				Trusted	Python 3 (ipyk	ternel) O	
8 + × Ø	🚯 🛧 🔸 🕨 Rur	n 🔳 C 🇯 😋	ode 👻	63						
к	NN Classifier									
In [49]:	<pre>from sklearn.nei model5 = KNeighb model5.fit(X tra</pre>	ighbors <b>import</b> K xorsClassifier(n xin, y train)	NeighborsClas: _neighbors = !	sifier 5)						
Out[49]	KNeighborsClassi	lfier()								
In [50]:	y_predict5=model	15.predict(X_tes	it)							
	C:\Users\Krushna functions (e.g. 1.0, this behavi ken will be elim g. mode. = stat	<pre>&gt;\anaconda3\lib\ `skew`, `kurtos ior will change: ninated, and the ts.mode( v[neigh</pre>	site-packages iis`), the def the default value None w ind, k], axi	\sklearn\neig ault behavior value of `kee ill no longer s=1)	ghbors\_classifica r of `mode` typica epdims` will becom r be accepted. Set	tion.py:228: Futur lly preserves the e False, the `axis `keepdims` to Tru	eWarning: Unlike axis it acts alor <sup>-</sup> over which the e or False to avo	other reduct ng. In SciPy statistic is pid this warn	ion 1.1 :ta in	
To (Tab. )	a colorista terr		,,,	,						
10 [21]:	from sklearn.met accuracyscore5#a print("Accuracy	rics import con accuracy_score(y score is :-",ac	fusion_matrix _predict5,y_t curacyscore5)	,accuracy_sco est)	ore,classification	_report				
	Accuracy Score i	is :- 0.81418414	121356384							
In [52]:	print("Classific	ation Report ",	classificatio	n_report(y_te	est,y_predict5))					
	Classification F	Report	precisi	on recall	f1-score suppo	rt				
	0 1	0.82 0. 0.81 0.	.93 0.87	282519						
	accuracy	0.	0.81	305905						
	macro avg weighted avg	0.81 0. 0.81 0.	.76 0.78 .81 0.81	385985 385985						
Select or create a r X	Flight_Delay_Predictic	on-Jupyter x +								
Select or create a r x	Flight_Delay_Predictic ss/Flight_Delay_Predictic	an - Jupyte: x + tion.ipynb	kpoint a minute e	(autosaverti	-			A* 0., 1	ia) g≞ (	@ (
Select or create a r x localhost 8888/notebool	Flight_Delay_Predictions ss/Flight_Delay_Predictions Flight_Delay_Predictions	an - Jupyter x + tion.ipymb	ckpoint: a minute a	igo (autosaved)			Teacher	At Dig 1	ia) tje t Logost	@ (
Select or create a = x localhost 8888/notebool j jupyter F File Edit Vi R + ac P	Flight_Delay_Predictic ss/Flight_Delay_Predictic Flight_Delay_Pre rw Insert Cell	an - Jupyter X + tion.ipymb diction Last Chec Kernel Widg	tkpoint: a minute a gets Help	go (autosaved)			Trusted	A <sup>th</sup> CQ 1 Python 3 (pyk	Cia t∕a t Logout termel) O	@ (
Select or create a = x localhost 8388/notebool Jupyter F File Edt Va B + 3c (2)	Flight_Delay_Predictic      Flight_Delay_Predict      Flight_Delay_Predict      Flight_Delay_Pre      Insert Cell      •• • • • Rur      1	an - Aupyte x + tion.jpymb diction Last Chec Kernel Widg 0.81 0.	tkpoint: a minute a pets Help cde V pp 0.68	go (autosaved)			Trusted	A <sup>t</sup> Ca 1 Python 3 (ipyk	Go tra ( Logout termel) O	@ (
Select or create a X localhost 5988/notebool Upypter F File Edt Vi S + sc 2g	■ Flight_Delay_Predictic cx/Flight_Delay_Predictic Flight_Delay_Flight_Delay_Flight Flight_Delay_Flight_Delay_Flight Flight_Delay_Flight_Delay_Flight Flight_Delay_Flight_Flight Flight_Delay_Flight Flight_Delay_Flight Flight_Delay_Flight Flight_Delay_Flight Flight_Delay_Flight Flight_Flight Flight_Flight Flight_Flight Flight_Flight Fli	an - Jupyte: x + ion.lpymb diction Last Chec Kernel Widg a C + 0 0.81 0. 0.81 0.	ckpoint a minute a pets Help ode 59 0.68 0.81 0.76 0.78 81 0.81	go (autosaved) III 103386 305905 305905			Trusted	A <sup>th</sup> Ch 1 Python 3 (ipyk	<mark>∆₀ t₂ (</mark> Logout iemel) O	⊕ (
Select or owned a 1 X localhost 5988/notebool Jupyter F File Edit V/ 20 + 3c (2)	Flight_Delay_Prediction     Flight_Delay_Predict     Flight_Delay_Predict     Flight_Delay_Pre     Insert Cell     t    t	an - Jupyle: x + ion.ipymb diction Last Chec Kemel Widg 0.81 0. 0.81 0. 0.81 0. 0.81 0. 0.81 0.	ckpont a minute a pets Help de V 59 0.68 76 0.78 81 0.81 edict5)	go (autosaved) 103386 305905 305905			Trusted	A <sup>th</sup> CR 1	Coout Logout ermel) O	@ (
Select or onsets at X localhout BBBR/notebool Jupyter F File Edt V/ B + Sc (2) In [53]: I	Hight_Delay_Predict     Hight_Delay_Predict     Hight_Delay_Predict     Delay_Predict     mer Cell     the Parameter Cell     the Parameter Predict     the Parameter Pre	m - Apple x + ion/pyhb diction Last Chee Kernel Widg a C P control of the 0.81 0. 0.81	ckpoint a minute a           pets         Help           ode         ~	go (autosaved) 103386 305905 305905			Trusted	A <sup>th</sup> CR 1 Python 3 (pyk	Cogout Logout ermel) O	⊕ (
Select or ormels at X localhout BBBR/notebool Jupyter F File Edt V/ B + Sc (2) In [53]: I out[53]: I	<ul> <li>Flight_Delay_Predicts</li> <li>Flight_Delay_Predict</li> <li>Flight_Delay_Predicts</li> <li>Flight_Delay_Predicts</li> <li>Inset</li> <li>Cell</li> <li>Cel</li></ul>	m - Apyle X + diction Last Chec diction Last Chec X = C + C - C - C - C 0.81 0. 0.81 0. 0.81 0. 0.81 0. 0.81 0. 0.81 0.	ckpont a minute s           pets         Help           ode         V           3.2         0.681           76         0.78           81         0.81           edict5)         d*)	go (autosaved) 103386 305905 305905	,		Trated	A <sup>h</sup> CR 1	G C C	⊕ (
Select or create a 1 x koalhout 5550 / notebool C Jupyter F File Edt V B + 3c 20 In [53]: 1 out[53]	<ul> <li>Rejk_Dieg, Pedick</li> <li>Right_Delay_Predict</li> <li>Right_Delay_Predict</li> <li>Right_Delay_Predict</li> <li>Right_Delay_Predict</li> <li>Right_Delay_Predict</li> <li>Right_Delay_Predict</li> <li>Right_Delay_Predict</li> <li>Right_Delay_Predict</li> <li>Right_Delay_Predict</li> </ul>	m - Apyle: X + diction Last Chec Micro Last Chec Micro Last Chec 0.81 0. 0.81 0. 0.81 0. 0.81 0. 0.81 0.	ckpoint a minute a pots Help de V 59 0.68 76 0.78 81 0.81 redict5) d')	go (autosaved) 103386 305905 305905	- 18000		Tusted	A <sup>th</sup> Ch 1	G C C	₪ (
Silect constant: x boothord 1000/nonebood y jupyter F Fle Est W B + k ⊗ In [53]: 1 out(53)	<ul> <li>Tight, Delay, Prediction</li> <li>Tight, Delay, Predi</li></ul>	m - Apple x + ion/pynb diction Last Check Kernel Widg C 0 0.81 0. 0.81 0. 0.81 0. rrik(y test,y pr emotoTrue, fat.*	ckpoint a minute e pots Help de V 59 0.68 76 0.78 81 0.81 redict5) d')	go (autosaved) 103386 305905 305905	- 180000		Tusted	A <sup>ta</sup> Chi 1 Python 3 (ky)k	G C (	@ (
Since to create at X branched SINSH-total Conduction JUDDYCLE File Exet Vic B + IK © In [ss]: I cot[ss]	Tight, Delay, Predicts     WiFight, Delay, Predict     WiFight, Delay, Predict     WiFight, Delay, Predict     WiFight, Delay, Pre     wiFight, Delay, Pre     wiFight, Delay, Predict     wiFigh	m - Jappin X + ion/pynb diction Last Chene Kernel Widg C 0 0.81 0. 0.81 0. 0.81 0. rrix(y test,y pr mont=True, fat="	ckport a minute at test Help de view 59 0.68 76 0.81 76 0.81 redict5) d')	go (autosaved) 103386 305905 305905 305905	- 180000 - 160000 - 140000		Trated	A <sup>th</sup> CR 1	G C (	₲ (
Since to create at X branhood XXXXX-relations y Uppyter F File Eat V. B + K @ In [53]: 1 out[53]	Pigle, Delay, Predicts     WFight, Delay, Predict     Piglet, Delay, P	m - Apptr X + tion(pynb) diction Last Chee Kernel Widg a C P O 0.81 0. 0.81 0.	ckport a minufe at the Help de V 12 0.007 0.81 0.	go (autosaved) 103386 105905 105905 105905 105905	- 18000 - 16000 - 14000 12000		Tranted ]	At CR 1	G C C	₪ (
Sidet ocenit # X braheat NUMPortection JUpyter F File Est Vi B + 1x 0 In [53]: I out[53]	Pigle, Delay, Predicts     WFight, Delay, Predict     Ilight, Delay,	In - Apple X + ton-lopeb diction Last Che- Mem Widg 0.81 0. 0.81 0.00 0.00 0. 0.81 0.00 0.00 0.00 0.00 0.00 0.00 0.00	ckpoint a minute a pots Help de V 59 0.68 61 0.78 61 0.61 rdi(t5) 1430	go (autosaved) 103386 105905 105905	- 180000 - 160000 - 140000 - 120000		[Tested]	A <sup>th</sup> Cii 1 Python 3 (pyk	G C C	@ (
Silect or context : X localized SIME/Antibook Display SIME B + K 0 In (33): I out(53)	Fight, Data, Predicts     Fight, Data, Predicts     Urlight, Data, Predicts     Tight, Data	m - Apple X + tonipytb diction Last Check Kernel Wold 0.81 0. 0.81 0. 0.81 0. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1	ckpoint a minule a pots Help de V 59 0.68 81 0.81 76 0.78 81 0.81 redict5) 1432	go (autosaved) 103386 305905 305905 305905	- 18000 - 16000 - 14000 - 12000 - 12000		Texted	A <sup>th</sup> CG 1 Python 3 (pyk	G D C	⊕ (
Silect or conter at localhood SIME/rootfood Office Jupyter F File Eat W B + k @ In [3]: I oct[53]	■ Fight, Delay, Predicts verFight, Delay, Predicts Fight, Delay, Prefice inst. Cell	m - Apple X + tonippeb diction Lac Check 0.81 0 0.81 0 0.81 0 0.83 0 0.83 0 0.83 0 0.83 0 0.83 0 0.84 0 0.85 0 0.	ckpoint a misule a           pets         Help           ode         V           59         0.68           76         0.78           0.81         0.81           edi(rt5)         1438	go (autosaved) 103386 103585 103585 103585 105585	- 180000 - 160000 - 140000 - 120000 - 100000 - 80000		Trated	A CG 1	G C (Lopot errel) O	⊕ (
Siend or conte at banknot Stiller vertex of a Signature of the second state of the	Fight, Delay, Predicts     WiFight, Delay, Predicts     WiFight, Delay, Predicts     WiFight, Delay, Pre     memt Cell     WiFight, Delay, Pre     wiFight, Delay,	m - Apple X + tonipyeb diction Last Check 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 - 0 -	ckpoint a minute a pets Help 55 0.668 75 0.78 81 edict5) 1430	go (autosaved) 103386 103386 305905 305905 10	- 180000 - 160000 - 140000 - 120000 - 100000 - 80000 - 60000		Tested	A <sup>th</sup> Chi 1 Python 3 (qyk	G to ( Logout annel) O	æ (
Since to create at X branchest XXXXV-restricted and XXXVV-restricted and XXXVVV-restricted and XXXVVV-restricted and XXXVV-restricted a	Fight, Delay, Predicts     Weight, Delay, Predicts     Weight, Delay, Predicts     Inight, Delay, Dela	sn - Apple X + toor.ipyeb dicition Last Check Kernel Weig 0.43 0.4 0.43 0.4 0.44 0.44 0.44 0.44 0.44 0.44 0.44 0.44	Ckport a minute of the second	990 (udsawed) 103360 305945 305945 10 10 10 10 10 10 10 10 10 10	- 180000 - 160000 - 140000 - 120000 - 100000 - 80000 - 80000 - 60000 - 40000		Trated	A" G 1	ig φ (	<b>(b)</b>
Since to create at X brainfood XXXXX outshood Uppyter F File Eat V. B + K @ In [53]: 1 out[53]	Fight, Delay, Predicts     WFight, Delay, Predict     Ilight, Delay, Delay	III - Apple X + toor.jopeb diction Last check Neme Weids 0.81 0. 0.81 0. 0.810	ckpont a minute a parts Help and Kinet 59 0.6.8 76 0.81 76 0.	92 (autoswed) 10356 1	- 180000 - 160000 - 140000 - 120000 - 100000 - 60000 - 60000 - 40000 - 20000		Tranted	A <sup>n</sup> CL 1	) q (j) Lugar Q (imma	· (
Silect or conter at X localized SIME/Antibook Display Sime Sime Sime B + k 0 In (S3): 1 Out(S3)	Fight, Delay, Predicts     UriFight, Delay, Predicts     UriFight, Delay, Predicts     Torman Cell     To	m - Apple         X         +           tion/publ         Uidtion         tasched           Kemel         Vidg         0           0         0         0           0         0         0           0         0         0           0         0         0           0         0         0           0         0         0           0         0         0           0         0         0           0         0         0           0         0         0	ckpont a minute a pots Help de	90 (autoswol) 20336 20336 305965 30596 3059	- 180000 - 160000 - 140000 - 120000 - 120000 - 80000 - 60000 - 40000 - 40000 - 20000		Texted	A <sup>n</sup> Ci 1 () Phina 3 (ayt	(a) (b) (t) (topot) ammel) (b)	æ (

Fig.7 K-Nearest Neighbor output

Expected outcome – 5) K-Nearest Neighbor

K-Nearest Neighbor output -0.8141 = 81.41% Accuracy with Classification Report

In this Fig.7 we observed that we got 81.41% accuracy by the K-NN algorithm. And it shows the classification report of the output.

## E. Naive bayes :

This algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.

<pre>vertextenderstandingstanding vertextenderstanding vertextenderstand</pre>	
<pre>pupy ter regit(_Della_prediction tand back get (2 M ( addamed)</pre>	
File       East Verse line of an line of Wingh Hulp       The and the analysis of the analysi	
<pre>Note by: Note by: 1: [4]: Monitory: 1: [4]:</pre>	
Name bayes       1m [st]: *     *	
<pre>in [41]: # (res silenc.naive hyper toonside#)</pre>	
<pre>bit(4); danslarM() 1 (5) # _predictModel.predict(_test) 1 (5) # _predictModel.predict(_test) 1 (4) # finiteAccuracy_sore(_predictAc_test)_sore(_sore(_lassification_report correspondencestary_sore(); predictModel.predict(); correspondencestary_sore(); predictAccuracy_sore(); correspondencestary_sore(); co</pre>	
<pre>in (4): # yredictAmodel.predict(X_text) in (4): # r circledra Excerney sore</pre>	
<pre>10 [41]: W relations accounty toor correspondences accounty toors print(Accounty toors) (print(Lt, tet)) print(Accounty toor is 1: 0.0112/200401007 Accounty toors is 1: 0.0112/200401007 In [47]: W print(Classification Report "relating (report(g.test,g_redicta)) Classification Report "printing over the state of the state over the</pre>	
Accuracy Score 1s := 0.831842887480427         Im [cr]: W print("Lassification mapper", classification recall f1-score support         Classification mapper "precision recall f1-score support         0       0.81       0.77       0.68       202510         accuracy does not be an experiment of the score support       0.68       0.68       0.68       0.69         accuracy mapped accuracy	
<pre>In [47]: W print('Classification Report ",classification rescall f1-score support</pre>	
Classification Report procision real fit-score support 0 0.61 0.077 0.82 202519 1 0.73 0.55 0.66 123366 accuracy macro mg 0.86 0.73 0.73 305965 weighted mg 0.86 0.81 0.82 30595 1n [48]: M (meconfusion matrix(y, test, y, predictA) ms. heatmap(on, amont True, fatt; 4) 0xf[48]: <a href="https://www.setsup.org">weighted: 1n [48]: M (meconfusion matrix(y, test, y, predictA) ms. heatmap(on, amont True, fatt; 4) 0xf[48]: <a href="https://www.setsup.org">weighted: 1n [48]: M (meconfusion matrix(y, test, y, predictA) ms. heatmap(on, amont True, fatt; 4) 0xf[48]: <a href="https://www.setsup.org">weighted: 1n [48]: M (meconfusion matrix(y, test, y, predictA) ms. heatmap(on, amont True, fatt; 4) 0xf[48]: <a href="https://www.setsup.org">weighted: 1n [48]: M (meconfusion matrix(y, test, y, predictA) ms. heatmap(on, amont True, fatt; 4) 0xf[48]: <a href="https://www.setsup.org">weighted: 1n [48]: M (meconfusion matrix(y, test, y, predictA) ms. heatmap(on, amont True, fatt; 4) 0xf[48]: <a href="https://www.setsup.org">weighted: 1n [48]: <a href="https://www.setsup.org"></a>weighted: 1n [48]: <a 10.1001="" doi.org="" href="https://www.se&lt;/td&gt;&lt;td&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;0 0.61 0.07 0.62 20519&lt;br&gt;xccuracy 0.66 0.53 0.65 0.65 103360&lt;br&gt;weighted ang 0.64 0.73 0.62 205005&lt;br&gt;weighted ang 0.64 0.73 0.62 205005&lt;br&gt;20 (40): 4 (xccurations and right yest, y product of the&lt;br&gt;inst-hetracy(cn, xmo):Tree, fmt: (d) )&lt;br&gt;Cot(40): 4 (xccurations and right yest, y product of the&lt;br&gt;inst-hetracy(cn, xmo):Tree, fmt: (d) )&lt;/td&gt;&lt;td&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;4:Stars and&lt;br&gt;weighted and&lt;br&gt;0.84         0.83         35995&lt;br&gt;0.87           1n         (4):         M         cenconfusion_matrix(y_test,y_predict4)&lt;br&gt;(ms.headsaw(u_monthree,test-0'))           Ont[4]:         (decconfusion_matrix(y_test,y_predict4)&lt;br&gt;(ms.headsaw(u_monthree,test-0'))&lt;/td&gt;&lt;td&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;Might and         0.48         0.70         0.77         200005           Might and         0.48         0.47         200005         200005           In [48]:         M cencenfusion_matrix/y_testy_great(ct4)&lt;br&gt;into-headbad(ct_memorthme_matrix/dt)         0.01(48):         0.01(48):         0.01(48):           Ont[48]:         dawsdaplot_memorthme_matrix/dt)         0.01(48):         0.01(48):         0.01(48):&lt;/td&gt;&lt;td&gt;&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;In [48]: M (executivision matrix(y, test, y, predict4)&lt;br&gt;ins.hestabg(ca, amot.True, int: 'd')&lt;br&gt;Ont[48]: &lt;a href=" https:="" j.jest.org="" j.jest.org<="" td=""><td></td></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a></a>	
on.hetdag(or,woot*ree,ret:"d") Ont[40]:  <b< td=""><td></td></b<>	
C O localhost 8888/notebooks/Flight_Delay_Prediction.lpynb A <sup>A</sup>	Ce 😪 🕫 🕞 Not syncing
JUDYter Flight_Delay_Prediction Last Checkpoint: a few seconds ago (autosaved)	Logout
File Edit View Insert Cell Kernel Widgets Help Trusted Python	(gykernel) O
Image: Image	
accuracy 0, 85 0, 76 0, 79 30595 macro ang 0, 86 0, 76 0, 79 30595 weighted ang 0, 84 0, 83 0, 82 30595	
In [48]: H cm=confusion_matrix(y_test,y_predict4)	
<pre>sns.heatmap(cm,annot=True,fmt='d') Out[48]: <axessubplot:></axessubplot:></pre>	
- 175000	
o - 196821 5698 - 150000	
- 125000	
- 10000	
- 75000	
- 45870 57516 - 50000	
- 25000	
o i	

Fig.8 Naïve Bayes output

Expected outcome - 6) Naive bayes

Naïve Bayes output -0.8314 = 83.14% Accuracy with Classification Report

In this Fig.8 we observed that we got 83.14% accuracy by the naïve bayes algorithm. And it shows the classification report of the output.

It can be seen that during the prediction, the one with the best performance among the Five algorithms is the Random Forest model. For example, the accuracy value for the Random Forest is 0.8458. This value is significantly higher than that of Logistic Regression, with the second-greatest value of 0.8379 accuracy. Similar patterns of noticeable differences for performance scores of Random Forest can also be seen in the other three measures. Besides the Random Forest, the two tree-based ensemble classifiers Logistic Regression and Decision Tree, are also better performed than others. The values of measures of these two algorithms are relatively similar. The difference between their performance scores and the other Two algorithms is also significant. All algorithms perform the operation and give their particular output. It predicts the accuracy of a flight delay with its proper classification report and final status. The user sends input to the modelling dataset for prediction. inputs like flight numbers, timings, etc. If the flight is delayed, then it shows a "Delay prediction." And if the flight is on time, then it shows "Flight on Time" to the application users.



	localhost 8888/noteboo	oks/Flight Deli	ay Prediction.jpynb									A <sup>A</sup> G	s s= 16	g (Not syncing 💲	
	C jupyter F	-light Dela	av Prediction La	st Checkpoint: (	2/24/2023	(autosaved)						2	Logout		
	Eie Edit Vi	en Incert	Cel Kernel	Wideats	Help						Trusted	Defense 1	Vinkemal) O		
	B + 3: 9	n + +	► Run ■ C	P Cole								I. Justice	(a)acito) o		
	C	Delayed I	Flights												
	In [18]:	for i in	df2.ARR DELAY:												
		if(i)	0): STATUS.append(1)												
		else: S	: STATUS.append(0)												
		df2.DEP_D df2['STAT	DELAY.fillna(0,in TUS`]=STATUS	olace=True)											
	Out[18]	:													
			OP_CARRIER OP_CA	RRIER_FL_NUM	ORIGIN 104	B5 DEST DEP_	DELAY ARR	-23.0	ELAPSED_TIME 268.0	DAY N	IONTH STAT	0			
		4	10	242	7 174	291	-0.0	-24.0	99.0	1	1	0			
		2	13	742	307	85	-5.0	-13.0	134.0	1	1	0			
		4	13	242	233	12	20.0	14.0	190.0	1	1	1			
									-			-			
		1022657 1022658	4	2393	8 18 8 112	112	-3.0 -8.0	-14.0	116.0	26 26	2	0			
		1022659	4	209-	1 18	245	0.0	3.0	123.0	26	2	0			
		1022660	4	2395	241	185	-6.0	-31.0	1/8.0	26	2	0			
		1022001		239	0 10	211	-3.0	-10.0	101.0	20	2	0			
		101000010	ows - to columns												
	) localhost 8888/noteboo	oks/Flight_Dela	ay.Prediction.ipynb ay_Prediction La	st Checkpoint: (	12/24/2023 Help	(sutosaved)					Tasted	A <sup>a</sup> Sa	5 5 6	g Nez grang 🕻	-
	Discalhost 8888/noteboo Discultor F Jupyter F File Edit Vi Di + 3: (2)	oks/Flight Dela Flight_Dela ew Insert	ay Prediction.jpynb ay_Prediction La Cei Kernei F Run E C	st Checkpoint: 0 Widgets Code	12/24/2023 Help	(autosaved)					Trusted	A <sup>h</sup> Si	5 S= 16 Logaut 8 (bykernel) O	g (Nersyncing 🕯	-
	i localhost 8888/noteboo j jupyter F File Edit Vi In [19]: I	ew Insert IS + + M df2=df2,d df2=df2,d df2=df2,d	ay Prediction lpynb ay_Prediction La Cel Kernel Run C drop(['ARR_DELAY' ()	st Checkpoint: ( Widgets ) Code ],axis=1)	Help	(autosaved)					Trusted	A <sup>t</sup> G	5 S= 6 Logait 8 (isykemel) O	g (Nerginning	-
	Iocalhost 8888/noteboo           Jupyter F           File Edt Vi           B         +           In [19]:           Out[19]	eks/Flight_Dela Flight_Dela ew Insett ID + + H df2=df2.d df2.head( : 0P_CA	ay Prediction lpynb ay_Prediction Ls Cel Kernel F Run E C trop(['ARR_DELAY'] O RRIER OP_CARRIER	st Checkpoint: ( Widgets # [Code ],axis=1) _FL_NUM_ORM	I2/24/2023 Help	(sutosaved)	CR5_ELAPSI	ID_TIME DAY	MONTH STAT	1.05	Trusted	A <sup>h</sup> 12 e Python 1	s ⊄= fe Logout 8 (ipykernel) O	g (Norsprang 🛔	-
	Diocalhost 8888/noteboo Jupyter F File Eat Vi Bi   +   sc   q] In [19]: I out[10]	ew Insert Elight_Dela ew Insert ID + + H df2xdf2xd df2xhead( c 0P_CA 0 1	ay Prediction layer ay_Prediction Ls Cel Kenel   Hun = C drop(['ARR_DLLAY') 0) RRHER OP_CARRIER 13 13	st Checkpoint: ( Widgets ) Code ],axis=1) pt_num ORM 2429 2477	12/24/2023 Help	(sutosaved)	CR5_ELAPSI	ю_тиме рау 208.0 1 93.0 1	MONTH STAT	0 0	Trusted	A <sup>th</sup> tử 🥐	5 S= Fe	g (Norgnang g	-
	Iocalhost3555/noteboo           jupyter           File           Edit           Im           +           in           (29):           Out[10]	oks/Flight_Dela Flight_Dela ew Insett ID + + H df2=df2,d df2=df2,d df2=df2,d 1 2	ay Prediction.lpynb ay_Prediction Ls Cel Kernel F Run C C drop(['AGR_DLLAY') O GRIER OP_CARRIER 13 13 13	st Checkpoint: ( Widgets Code ], axLs=1) PL_NUM ORM 2429 2427 2428	12/24/2023 Help V = AIN DEST 104 05 174 291 102 05	(sutosaved) DEP_DELAY -5.0 -8.0 -8.0	CR5_ELAPSI	ID_TIME DAY 288.0 1 93.0 1 134.0 1	MONTH STAT	105 0 0	Trusted	A <sup>th</sup> 😧 Python :	5 S= 6	g Nerspring 🖁	-
	Idealhost8558/noteboo Jupyter F File East Vi In [19]: 1 out[19]	oks/Flight_Dela Flight_Dela ew Insett IS	ay Prediction.lpynb ay_Prediction Ls Cel Kernel F Run C C drop(['AGR_DLLAY') O RRILER OP_CARRIER 13 13 13 13	st Checkpoint: C Widgets Code ],axis=1) 2429 2427 2428 2429 2427 2428 2429 2427	12/24/2023 Help W EST 104 05 174 281 1002 05 277 23 292 11	(sutosaved) DEP_DELAY 5.0 8.0 0.0 0.0 0.0 0.0 0.0 0.0 0	CR5_ELAPSI	LD_TIME DAY 268.0 1 134.0 1 134.0 1 1920 1	MONTH STAT 1 1 1	0 0 0	Trusted	A <sup>th</sup> 12 Python 1	5 S= Fe Logaut 8 (tpykernel) O	g Nerspring 🥞	-
	Doalhost SSSM/hotebook Dipupyter F File Eat Vi B +   B +   B +   B D +   B +   B +   B D +   B +   B +   B D +   B +	oks/Flight_Dela Flight_Dela ew Inset ID	ay Prediction.jpynb ay_Prediction Ls Cel Kernel Blan C drop(['Acce_DLLAY') OREER OP_CARRIER 13 13 13 13	st Checkpoint. ( Widgets Code J, axtis=1) PL_NUM ORIH 2429 2429 2426 2426 2426 2426 2424 2424	12/24/2023 Help IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	(sutosaved) DEP_DELAY -5.0 -8.0 -8.0 -8.0 -8.0 -8.0 -8.0 -8.0 -8	CR5_ELAPSI	D_TIME DAY 208.0 1 93.0 1 134.0 1 193.0 1 112.0 1	NONTH STAT 1 1 1 1	0 0 0 0 1	Trusted	A <sup>th</sup> 12 C	5 St Regard	g Norsprong	-
	<ul> <li>balhotSSS/rotebot</li> <li>jupyter F</li> <li>Fie Eat W</li> <li>Fie Star W<!--</td--><td>sks/Flight_Dela ew inset ID + + H df2=df2.d df2=head( 1 2 3 4 M df2=info( <class 's<="" td=""><td>ay Prediction.lpynb ay_Prediction La Cel Kernel FRM C Cel Kernel FRM C Cel Kernel Cel Kernel Cel</td><td>st Checkpoint, C Widgets Code , ax1s=1) 2429 2427 2429 2427 2426 2424</td><td>AN DEST 10/2 05 10/2 05 10/</td><td>(sutosaved) DEP_DELAY 5.0 -8.0 6.0 20.0</td><td>CRS_ELAPS</td><td>D_TIME DAY 288.0 1 99.0 1 134.0 1 192.0 1 112.0 1</td><td>MONTH STAT 1 1 1 1</td><td>0 0 0 0 1</td><td>Trusted ]</td><td>A* 12 Python :</td><td>5 🗲 🖗</td><td>g (Nergnong g</td><td>-</td></class></td></li></ul>	sks/Flight_Dela ew inset ID + + H df2=df2.d df2=head( 1 2 3 4 M df2=info( <class 's<="" td=""><td>ay Prediction.lpynb ay_Prediction La Cel Kernel FRM C Cel Kernel FRM C Cel Kernel Cel Kernel Cel</td><td>st Checkpoint, C Widgets Code , ax1s=1) 2429 2427 2429 2427 2426 2424</td><td>AN DEST 10/2 05 10/2 05 10/</td><td>(sutosaved) DEP_DELAY 5.0 -8.0 6.0 20.0</td><td>CRS_ELAPS</td><td>D_TIME DAY 288.0 1 99.0 1 134.0 1 192.0 1 112.0 1</td><td>MONTH STAT 1 1 1 1</td><td>0 0 0 0 1</td><td>Trusted ]</td><td>A* 12 Python :</td><td>5 🗲 🖗</td><td>g (Nergnong g</td><td>-</td></class>	ay Prediction.lpynb ay_Prediction La Cel Kernel FRM C Cel Kernel FRM C Cel Kernel Cel	st Checkpoint, C Widgets Code , ax1s=1) 2429 2427 2429 2427 2426 2424	AN DEST 10/2 05 10/2 05 10/	(sutosaved) DEP_DELAY 5.0 -8.0 6.0 20.0	CRS_ELAPS	D_TIME DAY 288.0 1 99.0 1 134.0 1 192.0 1 112.0 1	MONTH STAT 1 1 1 1	0 0 0 0 1	Trusted ]	A* 12 Python :	5 🗲 🖗	g (Nergnong g	-
	bahat 3888/voteboo jupyter F Fie Eat W (B) +   sc ep in [29]; I out[10] In [20]; I	ks:/Flight_Dela ew inset ID ← ↓ ID	ay Prediction lays ay_Prediction is Cel Kernal Hun C Cel Kernal Hun C Cel Kernal Cel	st Checkpoint ( Widgets Code ,axis=1) 2429 2427 2429 2427 2428 2424 2424 2424 2424 2424 2424 2424 2424 2424 2424 2424 2424 2429 2427 2429 2424 242 242	12/24/2023 Help NIN DEST 104 05 174 281 1002 05 2/7 223 233 12 2661	(sutosaved) DEP_DELAY -5.0 -8.0 -8.0 -8.0 -8.0 -8.0 -8.0 -8.0 -8	CRS_ELAPSI	ID_TIME DAY 208.0 1 99.0 1 134.0 1 199.0 1 112.0 1	MONTH STAT 1 1 1 1 1	US 0 0 0 1	Trusted	A <sup>a</sup> 12 Python 3	5 🗲 🖗	g (Nersprong d	-
	<ul> <li>boahest 3888/voeboo</li> <li>jupyter F</li> <li>File Eat W</li> <li>Ri (*) (a: (b)</li> <li>in (20): 1</li> <li>out(10)</li> <li>in (20): 1</li> </ul>	sks/Flight_Dela ew Inset IIght_Dela ew Inset IIght_Dela ew Inset IIght_Dela ft dt2.head( 1 2 3 4 W df2.info( cclass 'p Intseted at cols bata cols bata cols bata cols	ay hedicionisymb ay Prediction La Cel Kenel La Cel Kenel La Cel Kenel Cel Kenel Ce	st Checkpoint. ( Widgets Code (),ax(s=1) (),ax(s=1	12/24/2023 Help V = 31N DEST 104 05 174 291 1002 05 277 223 233 12 233 12 2661 t Dtyp	(sutosaved) DEP_DELAY -5.0 -8.0 -20.0 - -	CRS_ELAPSI	ID_TIME DAY 208.0 1 99.0 1 134.0 1 112.0 1	MONTH STAT	<b>30</b> 0 0 0 1	Trusted	A <sup>th</sup> Si	5 🕸 Τορατ i Topart δ (gykernei) Ο	g (Nersprong d	-
	<ul> <li>boahest 3880/voeboo</li> <li>jupyter F</li> <li>File Eat W</li> <li>B + a e</li> <li>a e</li></ul>	<pre>sks/Flight_Dels ew Inset Elight_Dels ew Inset # b   df2.info(     class 'p     intset     idd2.info(     i</pre>	ay frediction Levies ay Crediction Levies Cel Kenal Fina Cel Kenal Fina Ce	st Checkpoint. ( Widgets Code Code ( ,ax1s=1) ( ,a	12/24/2023 Help V = 1004 05 1774 291 1002 05 1774 291 1002 05 1774 293 1002 05 1774 293 1002 05 1774 293 1004 05 1774 293 172 1735 174 1745 175 1745 175 1745 175 1745 175 1745 175 1755 1755 175 1755	(sutosaved) DEP_DELAY -5.0 -8.0	CRS_ELAPSI	ID_TIME DAY 208.0 1 304.0 1 134.0 1 195.0 1 112.0 1	MONTH STAT	US 0 0 0 1	Trusted	A <sup>th</sup> tử Python :	5 🕸 Τορατ i topart δ (gykernei) Ο	g (no docad g	-
	Double 3888/notebook Double 3888/notebook Double 3888/notebook Press Car In [20]: 1 Out[10] In [20]: 1	sks/Flight_Dels Flight_Dels ev inset ID + + ID + +	ay hedictoniayab ay Predictoniayab ay Predictoniayab Cel Kenel Film C C asses or_cases to case or_cases to case or_cases to case or cases to cases	st Checkpoint: 0 Widgets Code (cod	12/24/2023 Help I I I I I I I I I I I I I I I I I I I	(sutosaved) DEP_DELAY 	CRS_ELAPSI	ID_TIME DAY 208.0 1 394.0 1 194.0 1 195.0 1 112.0 1	MONTH STAT	US 0 0 0 1	Trusted	A <sup>th</sup> 12 Python 1	5 \$ <sup>±</sup> € i copart 3 (pykemer) O	i (wayang f	-
	bolant 3888/rosebook	sks/Flight_Dels Flight_Dels w Inset ID + + D df2.df2.df2.d df2.hf2.df df2.hf2.df df2.hf2.df df2.hf2.df df2.hf2.df df2.hf2.df df2.hf2.df df2.hf2.df df2.hf2.df df2.hf2.df df2.df2.df df2.hf2.df df2.df2.df2.df df2.df2.df2.df df2.df2.df2.df2.df df2.df2.df2.df2.df2.df2.df2.df2.df2.df2.	ay frediction used ay Prediction used Cel Kernel Film C C drep[//weipclaw/ 12 13 13 13 13 10 00 00 00 00 00 00 00 00 00	st Checkpoint: 0 Widgets Code (0,000) (0,00)	an DEST           an DEST <td< td=""><td>(sutosaved)</td><td>CRS_ELAPSI</td><td>10_TIME DAV 208.0 1 93.0 1 134.0 1 193.0 1 112.0 1</td><td>MONTH 51A1 1 1 1 1 1 1</td><td>00 0 0 0 1</td><td>Trusted</td><td>A<sup>th</sup> 🗘</td><td>5 \$<sup>±</sup> € i topat ((pykeme) O</td><td>ij (versprang dj</td><td>-</td></td<>	(sutosaved)	CRS_ELAPSI	10_TIME DAV 208.0 1 93.0 1 134.0 1 193.0 1 112.0 1	MONTH 51A1 1 1 1 1 1 1	00 0 0 0 1	Trusted	A <sup>th</sup> 🗘	5 \$ <sup>±</sup> € i topat ((pykeme) O	ij (versprang dj	-
	bookerstassinverseout	Alternation         Alternation           Start         Insett           ID         Image: Start           ID         <	ay PredictionLayed ayPredictionLayed Cell Kennel Layer Cell Kennel Layer Cell Kennel Cell	at Checkpont: 0 Widgets by Code code (action), axis=1) 2427 2429	an DEST	(sutosaved) DEP_DELAY -5.0 -8.1 -5.0 -8.1 -2.0	CRS_ELAPSI	208.0 1 208.0 1 134.0 1 134.0 1 193.0 1 112.0 1	MONTH STAT	0 0 0 0 1	Tuated	A <sup>th</sup> 12 Protoco 2	S S= Fe Logar S (pykernet) O	a (Koropenny (	-
	Doublet 3888/rotebook Dipyter 1 Free Cat M (n + ) sc (n) an [20]: 1 out[10] In [20]: 1	AstarFlight Dela Flight_Dela w Inset ID + + df2:df2.df2.d df2.thead( df2.thead) i i df2.thead( df2.thead) i i df2.thead( df2.thead) i i df2.thead( df2.thead) i i df2.thead( df2.thead) i i i i i i i i i i i i i i i i i i i	ay Prediction Layed ay _ Prediction Layed Cel Kenel i hun C ay _ Cel Kenel i a a a a a a a a a a a a a a	st Checkpont. C V/dons Code (,axis=3) (,	BN DEST           W	(subserved) DEP_DELAY 4 0 2 0 2 2 4 4 2 2 2 2 4 4 4 4 4 4 4	CR5_ELAPSI	ID_TIME DAY 208.0 1 99.0 1 134.0 1 139.0 1 112.0 1	NONTH STAT 1 1 1 1	0 0 0 1	Tastel	A <sup>th</sup> 12 Proton 5	s S= R togan (gykenne) O	a (marahada)	-
	Double 3888/rotebook Double 38888/rotebook Double 38887/rotebook Double 3888/rotebook Doub	sku/Flight_Dela Flight_Dela ev Inset ID + + dt2.head( 2 1 2 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4	ay fredictionlayed ay_Prediction is Cell Kernel Arrow (1 Arrow Content Cell Kernel Arrow (1 Arrow Content Cell Kernel Arrow (1 Arrow Content Cell Kernel Cell Kernel	st Checkpont: C V/dpots Code (v/dpots Code (code (code)	BN DEST           SN DEST <td< td=""><td>(subseved) DIP_DELAY </td><td>CRS_RLAPS</td><td>10,1106 0A/ 20100 1 101.0 1 102.0 1 102.0 1 102.0 1</td><td>- MONTH STAT</td><td>0 0 0 0 1</td><td>Tualet</td><td>A<sup>1</sup> Q</td><td>S S= R logar S (gykerne) O</td><td>i (koropenny (</td><td></td></td<>	(subseved) DIP_DELAY 	CRS_RLAPS	10,1106 0A/ 20100 1 101.0 1 102.0 1 102.0 1 102.0 1	- MONTH STAT	0 0 0 0 1	Tualet	A <sup>1</sup> Q	S S= R logar S (gykerne) O	i (koropenny (	
	bolderstässinversecker	$\begin{array}{ccc} & & & & & \\ & & & & \\ & & & & \\ & & & \\ & & & \\ & & & \\ & & & & \\ & & & & \\ & & & \\ & & & & \\ & & & & \\ & & & & \\ & & & & \\ & & & & \\$	ay Predictionupub ay Prediction Le Cell Kennel hear a Cell argentization arg	at Checkpont. ( Widgets Cole J, axtis=3) Cole 2429 2427 2428 2427 2428 2427 2428 2429 2428 2429 2428 2429 249	2224/2023 Help V = 2007 200 2017 201 2017 2017 201 2017 2017 201 2017	(subserved) DOP_DELAY 	CRS_ELAPS	00_1146 047 2010 0 1340 1 1340 1 1120 1	1 MONTH 5101	<b>us</b> 0 0 0 1	Tusted	A <sup>1</sup> (g	s 🛠 🤅	i (tersprong )	
*C .	Dealbet 3888/rotebook	Bit         Control         Co	ay Predictionupueb ay Prediction Le Cell Kernel han 0 C the Kernel han 0 C han 0 C	st Chaocont ( Whipe) Whipe) Code ( ( ) ) ( ) ( ) ) ( ) ) ( ) ) ( ) ) ( ) ) ( ) ) ( ) ) ( ) ) ( ) ) ( ) ) ) ( ) ) ) ( ) ) ( ) ) ( ) ) ( ) ) () ) ) ( ) ) ( ) ) () ) ) ( ) ) ) () ) ) ) () ) ) () ) ) ) () ) ) ) () ) ) ) ) () ) ) ) ) () ) ) ) ) ) () ) ) ) ) ) ) ) ) ) ) ) ) ) ) ) ) ) ) )	2224/2023 Help v = 2004 05 2774 201 1004 05 2774 201 1004 05 2774 201 1004 05 2774 201 1004 05 2774 201 2860 t Dfygg 2860 t Dfygg 11 inter 11 inter	(subsaved)	CRI_ILANS	00,100,100,000 11 000 1 100 1 1120 1 1120 1	MONTH STAT		Tustel	A <sup>b</sup> Co		i (krynny j	
PC STATE	boldet 3888/rotebold	skarfight Dele light_Dele we have to be the first of the second seco	ay Predictionupub ay _Prediction Le	st Checkont C Widputs Widputs Widputs Content	2224/2023 Help N DEST AN DE	(subsaved)	CR. JLWS	D_1146 DAV 2010 1 99a 1 1120 1 1120 1	HONTH TRA 1 1 1 1 1 1 1 1 1 1 1 1 1		Trated	A <sup>b</sup> Cr	s s s s s s s s s s s s s s s s s s s	1 (ber sprong )	
re sumy	bolant 3888/rotebol	HarRight Dele Hight Dele W finat D ← 0 M df2-df2, head 0 1 2 3 4 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4 1 5 1 1 1 1 5 1 5 1 1 1 1 1 1 1 1 1 1 1 1 1	ay /hedictionlayeb ay_Prediction is Cell Kernel bins Cell Kernel bins Cell Kernel bins Cell Cell Cell and Cell Cell Cell and Cell and Cell Cell and Cell	st Chedgooff. (2) Wolgois )) Code ), active1) 200 200 200 200 200 200 200 20	22242023 22242023 22400 224000 22400 22400 22400 224000 224000 224000	(subsaved)	CRI LLAPS	2010 100 100 100 100 100 100 100 100 100	80000 920 1 1 1 1 1 1 1		Tratel		5 S S Guard	1 (του τριτουρ 🧍	
e La surriv	bolderstässikvesebode	skurfight Dela Bight Dela W Inset W	ay Predictionued ay Prediction Le Cell Kernel han 0 C the point of the constraints and t	4 Chadquet C Widgels ⇒ [Code ],atis1) 269 269 269 269 269 269 269 269	322442023           HHe           Image: State Stat	(subsaved)	CR.,LAM	000 11 11 11 11 11 11 11 11 11 11 11 11	BOIN 51		Thated		5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	3 ( 160 genorg ) 10 ( 10 genorg )	

In this fig.9 It shows the delayed flight data, so from that data only arrival delay data should be finalized for the prediction process. All data should be put inside the web page for prediction process.



Fig.10 Login page

In this login page, user should put their login id and password so they can log into their page.



Fig.11 Flight input Information

After login it shows the input methods for prediction process, so user need to put all the data from arrival delay for prediction process.

	x u - v
← → C (© 127.0.0.1.8000/customer/	(2) 20 20 20 20 20 20 20 20 20 20 20 20 20
Res	sult
Wo	nt be Delay
Fig.12 F	Final Prediction

After entering all input user need to press predict button. Then it shows the final prediction of the flight if it is delayed or not.

## VIII. CONCLUSION AND FUTURE SCOPE

Machine learning algorithms have been implemented to predict individual flight delays. The experimental results show that the random forest-based method can obtain good performance for the prediction task. Every algorithm performs the operations and predicts the accurate values. After getting the output result, it should be classified and shown on the accuracy graph.

Which algorithm predicts the highest and best accuracy that is going to be used as the final output for delay prediction? Users can see the flight status on the web page. Based on the analysis of their results, it is evident that the integration of multidimensional heterogeneous data, combined with the application of different techniques for feature selection and regression, can provide promising tools for inference in the domain. Machine learning algorithms that are used predict delays with high accuracy, which gives a proper result. This project can be used in other services like train, metro, and bus transportation systems.

#### REFERENCES

- [1] JJ Rebollo and H. Balakrishnan, "Characterization and prediction of air traffic delays", *Transportation Res Part C Emerg Technol.*, vol. 44, pp. 231-41, Aug 2014.
- [2] AJ Reynolds-Feighan and KJ Button, "An assessment of the capacity and congestion levels at European airports", *J Air Transp Manag.*, vol. 5, no. 3, pp. 113-34, Nov 1999.
- [3] S AhmadBeygi et al., "Analysis of the potential for delay propagation in passenger airline networks", *J Air Transp Manag.*, vol. 14, no. 5, pp. 221-36, June 2008.
- [4] Y Tu, MO Ball and WS. Jank, "Estimating flight departure delay distributions-a statistical approach with long-term trend and short-term pattern", J Am Stat Assoc., vol. 103, no. 481, pp. 112-25, Aug 2008.
- [5] S Oza et al., "Flight delay prediction system using weighted multiple linear regression", Int J Eng Comp Sci., vol. 4, no. 05, pp. 11765, Feb 2015
- [6] C-Y Hsiao and M Hansen, "Air transportation network flows: equilibrium model", Transp Res Rec., vol. 1915, no. 1, pp. 12-9, Feb 2005.
- [7] R Britto, M Dresner and A Voltes, "The impact of flight delays on passenger demand and societal welfare", Transp Res Part E Logist Transp Rev., vol. 48, no. 2, pp. 460-9, Nov 2012.
- [8] S AhmadBeygi et al., "Analysis of the potential for delay propagation in passenger airline networks", J Air Transp Manag., vol. 14, no. 5, pp. 221-36, Oct 2008.
- [9] Bin Yu et al., "Flight delay prediction for commercial air transport: A deep learning approach", *Transportation Research Part E: Logistics and Transportation Review*, vol. 125, pp. 203-221, Jan 2019.
- [10] Ding Yi, "Predicting flight delay based on multiple linear regression", IOP Conference Series: Earth and Environmental Science, vol. 81, no. 1, Jan 2017.
- [11] Bojia Ye et al., "A Methodology for Predicting Aggregate Flight Departure Delays in Airports Based on Supervised Learning", *Sustainability*, vol. 12.7, pp. 2749, Aug 2020.
- [12] Sina Khanmohammadi, Salih Tutun and Yunus Kucuk, "A new multilevel input layer artificial neural network for predicting flight delays at JFK airport", *Procedia Computer Science*, vol. 95, pp. 237-244, June 2016.
- [13] Jennifer S. Raj and J. Vijitha Ananthi, "Recurrent neural networks and nonlinear prediction in support vector machines", *Journal of Soft Computing Paradigm (JSCP)*, vol. 1.01, pp. 33-40, June 2019.
- [14] Vignesh Muthukumar and N. Bhalaji, "MOOCVERSITY-Deep Learning Based Dropout Prediction in MOOCs over Weeks", Journal of Soft Computing Paradigm (JSCP), vol. 2.03, pp. 140-152, Apr 2020.

