**ENS 491-492 – Graduation Project**

**Draft Final Report**

**Project Title:Prediction of Remaining Life of the Aircraft Engine**

**Group Members:Berfu Aydın – 23520**

**Ahmet Rauf Yavuz - 23887**

**Supervisor(s): Ahmet Demirelli**

**Date:** 15.05.2021



· **ENS 491-492 Final Report must contain the following sections (leave section titles unchanged).**

· **You are required to write clearly and concisely: The main body of the report (not including the Title Page, Appendix, and References) must be between 20-25 pages.**

· **Project advisor may have additional requirements for the report.**

**1.** **EXECUTIVE SUMMARY**

- *The issue or problem that was solved/researched.*

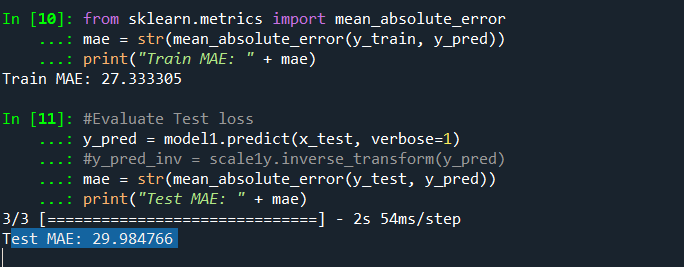
Aviation sector is getting more and more important today. In addition to building aircraft, their maintenance is quite difficult and costly. Aircraft that are not maintained or whose condition is uncertain cause irreparable casualties. Our starting point in this project: It is to prevent high costs and loss of life by ensuring that the aircraft that are close to the end of their engine life are detected in advance and the engine supply is made as quickly as possible.

Our project aims to create machine learning algorithms to predict the remaining life of the aircraft engine based on approximately 20 sensor data obtained from jet aircraft engines. While doing this, it will be determined which model will work more effectively and with high accuracy by using different RNN models available in the literature. In this process, synthetic data we have obtained from NASA will be used.

There are some completed works that we can access and see their results for comparing with our work.

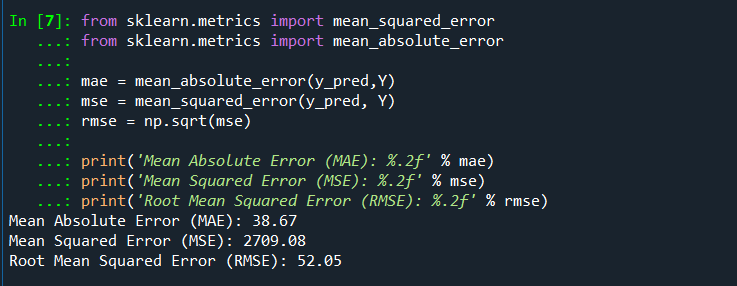
One of them is a Project that was held in 2008 with PHM data, they got an 84.19% of accuracy. There is another project that uses stacked sparse Autoencoder. They obtained 83.82% accuracy which is close to the first project that we mentioned. One of the best projects that related to our project has quite high accuracy which is 95% regression accuracy. They used PSO-SVM based algorithm in this project4.

- *Results and Findings*



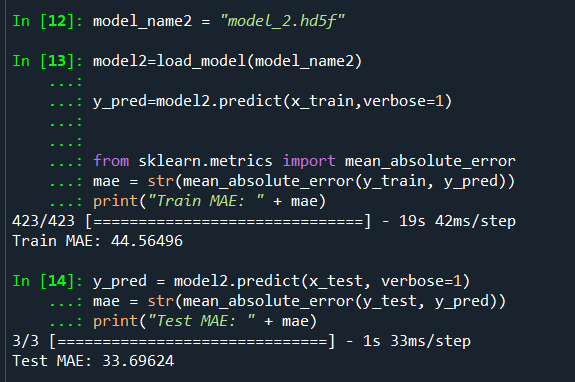
Result of BLSTM

In our BLSTM model we got the best score which is 29.98 Mean Absolute error. In statistics, mean absolute error (MAE) is a measure of errors between paired observations expressing the same phenomenon.



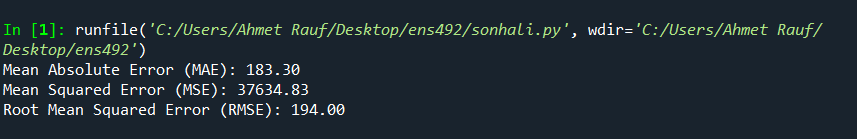
Result of Random Forest

As you can see, by means of Random Forest, we find 38.67 mean absolute error score which is pretty good for our model.



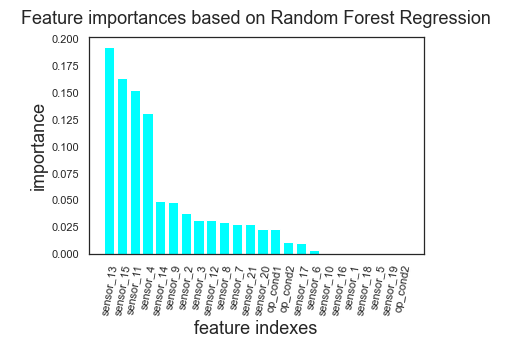
Result of LSTM

Using LSTM, we achieved a good result of 33.69. Although it was relatively low compared to the BILSTM result, we still got a good result.

**

Result of Naive Bayes

When we used Naive Bayes, we got a score like 183.30. Obviously it was a bad result compared to other models. It was a model that we did not prefer, but still we wanted to put it in our visuals for us to see and compare.



We have determined which feature is more important using Random Forest, which is usually applied here. As can be seen in the figure, sensor 13 data is the most important feature of our data set.

- *Methodology or techniques that were utilized in the project.*

First method that we agreed to work with in this project is LSTM(Long Short-Term Memory). LSTM is a type of recurrent neural network that can learn order dependence in sequence prediction problems.LSTM is one of the Recurrent Neural Network methods. Recurrent Neural Networks (RNN) are more useful when working with sequential data. They basically derived from feed forward networks but RNNs perform similar behaviours to human thinking. RNNs don't make evaluations by not only looking at the actual input parameters, they also consider the previous inputs. That means that in RNNs we have memory which allows them to memorize what happened in the past in processing. LSTM is actually an upgrade for RNNs. Because RNNs have some problems about learning and LSTMs solved vanishing gradients and exploding gradients problems. These problems are related to the training part of the network .

The second model that we utilize in our project is BLSTM(Bidirectional LSTM). BLSTM is actually an upgrade for LSTM. The basic idea behind BLSTM is that we divide the forward and the backward training sequences into two different recurrent neural networks but they are connected to the same output layer.

Another algorithm that we used is the Gaussian Naïve Bayes algorithm. Naïve Bayes is a machine learning model that assigns class labels to problem occurrences. It is a simple classification technique but it is working well. However, we need to use Gaussian Naive Bayes because our data contains continuous values and it is a regression problem. Gaussian Naive Bayes model which works well with continuous values. In Gaussian Naive Bayes, we are estimating only mean and the standard deviation of our training dataset and that makes it slightly easier to work with it.

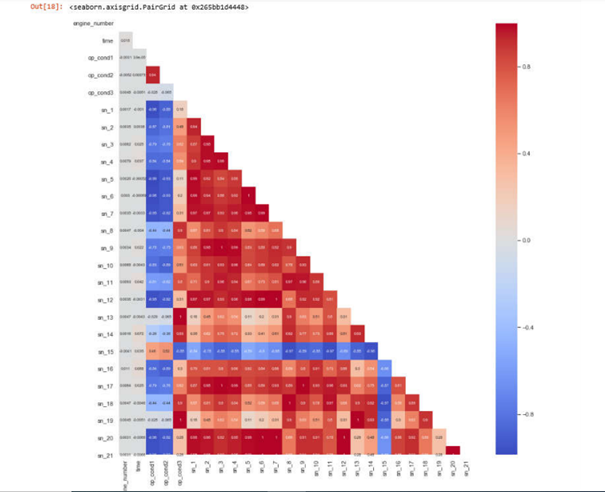
Another method that we decided to use in this project is the Random Forest algorithm which is a widely - known ensemble algorithm both used for classification and regression problems. Random Forest is a combination of multiple decision trees.

Later steps on the project we decided to enhance the results from our random forest algorithm with K-Fold Cross Validation. Cross- validation is a resampling process used to evaluate machine learning models. Cross validation is mainly used in machine learning to estimate the ability (how good a model is) of the model with unseen data.

**2.** **PROBLEM STATEMENT**

**2.1.** **Objectives/Tasks**

The aim so far is to recognize the available data and to find correlations between them by visualizing these data. This part, called data processing, is one of the most important parts before the machine learning model is created. In this way, the model is purified from unnecessary data and creates a measure for the result to be more consistent and for possible problems to be encountered in the future.

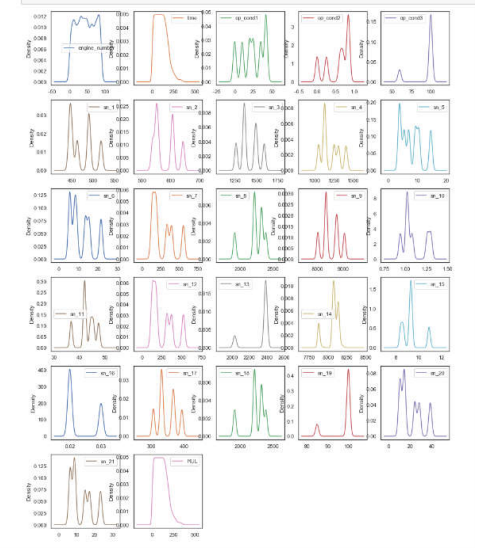
**

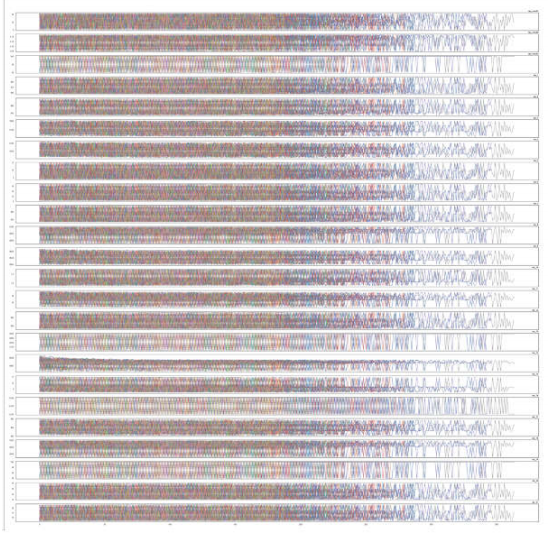
Heatmap is formed which is a very useful visualization tool, generally shows the correlation between

features. The red colored boxes indicate that the correlation is intense. It can be seen that there are

many features that have positive correlation in general. Light colored (eg white colored) boxes represent low correlation. For example, the value between sn\_9 and sn\_3 sensors is 1. This represents a complete

positive correlation between these sensors. Which features will be used will be determined by considering all these correlations. Unnecessary features will not be included in the model.

**

**

This chart is one of the most efficient. Some data contains too much noisy data. This data does not need to be included in the model because it both deflects the result and creates unnecessary load for the model.

When looking at the general distribution of the feature values, it can be seen that the change depending on the specific motors is very intense in each feature. It seems difficult to find a general trend. However, it seems a little more likely that it can be found a specific trend for sn\_8, sn\_11, sn\_14, sn\_15 and sn\_18 sensors.

While creating the RNN (Recurrent Neural Network) model, different machine learning algorithms available in the market will be used. In this way, it will be observed which model gives better results.

**2.2.** **Realistic Constraints**

In this project since we only deal with data and computers the realistic constraints are about the economic and manufacturability in a sense. Since we are dealing with relatively big data our computers may not have enough computational power to handle the models that we are trying to develop. In the early steps of the project one of the team members' computer crashed and we lost all the progress. Later we got a capable computer and continued our work. This is an economical and manufacturability issue that we encounter while working on the project.

In our project we need to estimate the remaining life of an aircraft engine which is carrying people so that we need to be as precise as possible. One wrong calculation or wrong step may cause unwanted casualties. For this reason we need to satisfy that engineering standard of our speciality field.

3. **METHODOLOGY**

First method that we agreed to work with in this project is LSTM(Long Short-Term Memory). LSTM is a type of recurrent neural network that can learn order dependence in sequence prediction problems. Traditional feed-forward neural networks and recurrent neural networks are different in the sense of processing data. In traditional feed forward neural networks we have connections between nodes from starting point(input layer) to the end(output layer). The information that we feed our model flows in only one direction, first in input nodes then hidden nodes and lastly Output nodes. We can not form cycles in that network model. On the other hand Recurrent Neural Networks (RNN) are more useful when working with sequential data. They basically derived from feed forward networks but RNNs perform similar behaviours to human thinking. RNNs don't make evaluations by not only looking at the actual input parameters, they also consider the previous inputs. That means that in RNNs we have memory which allows them to memorize what happened in the past in processing. That means that we are creating “cycles” in our network flow which is from output nodes to hidden layers we can feed information that we already learn to make stronger predictions. This is the main reason RNNs work well with sequential data as well. There are 3 basic rule if we want the work with RNNs

(Yoshua Bengio, et al.1994.)5

1. We need enough memory while RNNs are running because it has a memorization feature and uses past inputs that we need to store.
2. System need to be resistant to noise
3. System parameters need to be trainable in a reasonable time.

LSTM is actually an upgrade for RNNs. Because RNNs have some problems about learning and LSTMs solved vanishing gradients and exploding gradients problems. These problems are related to the training part of the network .

*“Hence standard RNNs fail to learn in the presence of time lags greater than 5 – 10 discrete time steps between relevant input events and target signals. The vanishing error problem casts doubt on whether standard RNNs can indeed exhibit significant practical advantages over time window-based feedforward networks. A recent model, “Long Short-Term Memory” (LSTM), is not affected by this problem. LSTM can learn to bridge minimal time lags in excess of 1000 discrete time steps by enforcing constant error flow through “constant error carrousels” (CECs) within special units, called cells” (*Felix A. Gers, et al).7

LSTM networks consist of some recurrently connected memory blocks. Each LSTM unit consists of an input, an output and a forget gate that enables memory blocks to write, read and reset operations for the networks.

The second model that we utilize in our project is BLSTM(Bidirectional LSTM). BLSTM is actually an upgrade for LSTM. The basic idea behind BLSTM is we divide the forward and the backward training sequences into two different recurrent neural networks but they are connected to the same output layer.This means that for every point in a given sequence, the BLSTM has complete, sequential information about all points before and after it are considered and in the light of that informations predictions are made.

Another algorithm that we used is the Gaussian Naïve Bayes algorithm. Naïve Bayes is a machine learning model that assigns class labels to problem occurrences. It is a simple classification technique but it is working well. Naïve Bayes assumes that the value of features is independent of each other and contribution of each feature incorporated to the model is equal. It calculates the probability of a sample for deciding which category that sample belongs to based on previous information algorithm learned during training. For example, an animal can be considered to be a dog if it has ears and is a large animal. Naïve Bayes works in a way that each of these constraints contribute independently to the probability that this animal is a dog, without looking for any possible correlations between other features. However, our problem is not a classification problem, for that reason we used Gaussian Naive Bayes model which works well with continuous values. In Gaussian Naive Bayes, we are estimating only mean and the standard deviation of our training dataset and that makes it slightly easier to work with it.

The final method that we decided to use in this project is the Random Forest algorithm which is a widely - known ensemble algorithm both used for classification and regression problems. The Random Forest algorithm performs well in large data and it gives parameter estimates such that each individual decision tree in the algorithm executes fast and precisely. The trick behind this algorithm is that Random Forest is a combination of multiple decision trees. For understanding The Random Forest algorithm we need to understand the Decision trees first. Decision trees can work both with discrete and continuous variables (univariable). We can use decision trees for both classification and regression problems. Learning in Decision trees is greedy. That means that in each step we need to pick best at the moment to move on and recursively iterate over each node in the tree until the last parent node. For a particular problem we can build many different decision trees. If we combine and look at every one’s output, we have a single output as a combination of all trees in the decision tree. This combination can be made by taking their average or we can give them weight and then take their average as well as a relatively more simple voting system for evaluation. This whole system is called the Random Forest Algorithm.

Later steps on the project we decided to enhance the results from our random forest algorithm with K-Fold Cross Validation. Cross- validation is a resampling process used to evaluate machine learning models. Cross validation is mainly used in machine learning to estimate the ability (how good a model is) of the model with unseen data. In Cross Validation parameter k refers to the number of folds (how many parts will be created when data is divided). Cross Validation first shuffles the data randomly and after that according to the parameter k it splits data into groups. Separates 1 group for validation and starts training with remaining groups. When training is done it tests the model with validation set and records the score. It repeats for every set in the data. and after that we can evaluate the models attribute with our data

The two most likely problems to be encountered are underfitting and overfitting problems. These are the problems that are generally encountered and cause the result to be inefficient.

In short, the overfitting problem is that the model sticks to existing data too much. Another saying is that it starts to memorize rather than learning. In other words, the algorithm moves away from solving the main problem and focuses on attaching the values given to it. it focuses so much that it fits the values perfectly, but therefore it avoids solving the problem it has to solve.

Let me try to explain this with a concrete example from daily life. Imagine a student going to take the exam. But this student memorized the answers of previous exam questions, rather than learning the insight of topics a few days before the exam. When we put new questions in front of this student, this student will answer incorrectly because s/he memorizes the past questions. Likewise, if the model we created is very sticked to data in the training set, it will give wrong results when given the test data.

Under fitting, on the other hand, means that if a model has under-learning, as opposed to over-learning, the model does not fit the training data and therefore misses trends in the data.

There are some ways to eliminate the overfitting problem.

**1.** **Reducing the network’s capacity**

Our first model has a large number of trainable parameters. As this number increases, the model memorizes the target class so easily for each training example. As a result, this is not ideal for generalizing on new data. By lowering the capacity of the network, you force it to learn the patterns that matter or that minimize the loss.

**2.** **Applying regularization**

Regularization techniques are generally used to increase performance by preventing overfitting in the designed model. In addition, there are cases where it is used to reduce the complexity of the model without degrading performance.

**3.** **Train with more data**

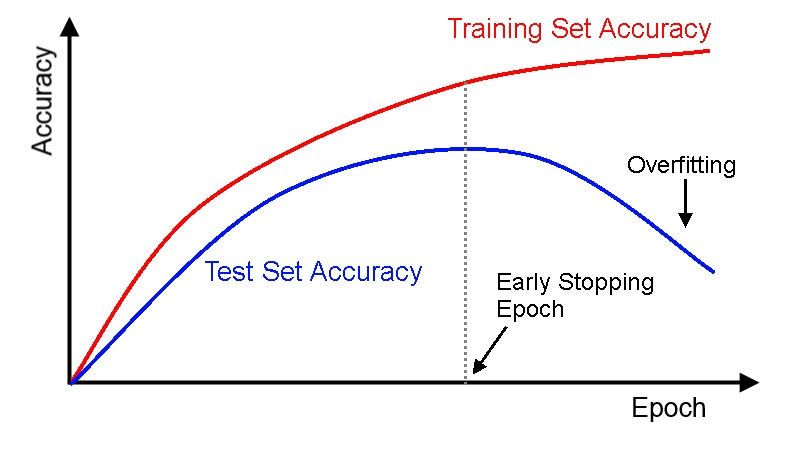
Even, it won’t work every time, but training with more data can help algorithms detect the signal better. With more data, our model can obtain a more stable result by obtaining more signal data.

**4.** **Remove Features**

Some unnecessary, meaningless and dirty data prevents our model from finding a more stable and consistent result. Our model can be simplified by removing unnecessary features. For this, algorithms such as random forest can be used.

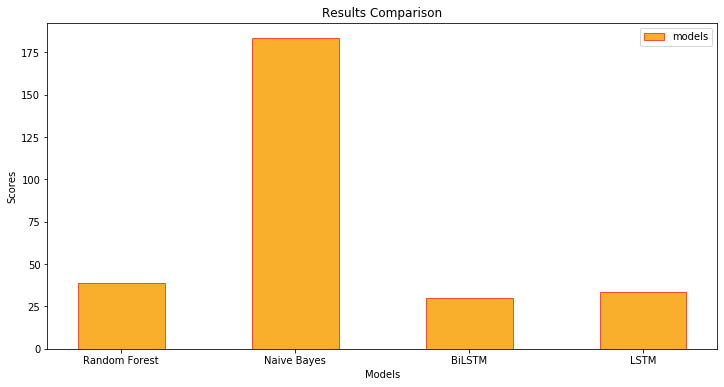
**5.** **Early Stopping**

In order to prevent unnecessary over-learning, the algorithm stops learning as soon as the model starts to memorize. This prevents too much reliance on data.In other words, our model stops working when it starts overfitting.

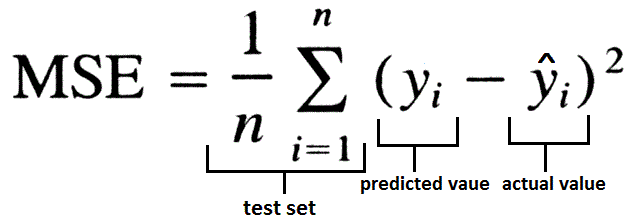


4. **RESULTS & DISCUSSION**

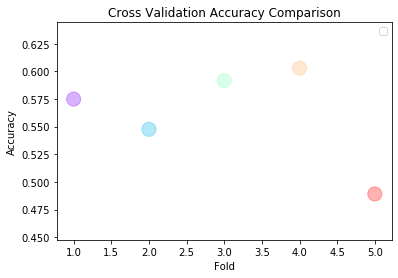
The purpose of our project was to compare the results we obtained using different neural network models available on the market. For this, we used 4 main methods that are frequently used in the market. Finally, we created a visual to find the differences between the results we got.



We used mean square error scores to evaluate the results. To briefly mention mean square error, it measures the average of the **squares** of the **errors**. In other words, it gives the average **squared** difference between the predicted values and the actual value. This applies to all results and gives the sum. The reason for squaring is to prevent the error from taking a negative value.



When we look at our result table, we can easily see that we got the best result from Bidirectional LSTM. As we mentioned above, Bidirectional LSTM is a special case of the LSTM type. Therefore, it generally gives better results. Later, we obtained the best results from LSTM, Random Forest and Naive Bayes, respectively.



We also displayed the data we obtained as a result of cross validation on a visual.

5. **IMPACT**

The biggest impact of our project will be on the aviation and defense industry sectors. These kinds of sectors are very costly and they are the areas where international agreements gain importance. The ability to manufacture and supply a part is both difficult and very costly. Thanks to this project, aircraft engines can be ordered and supplied without delay, as there will be prior notification. Thus, economic gain can be achieved.

Again, if we think in the technological field, we can compare the 5th generation warplanes, which we call the new generation, to a flying software. They are aircraft with many technological equipment and software. When we think about it, we can think of this project not only in terms of engine but also in terms of safety. This problem can be easily detected from the sensor data when any defect or deterioration occurs in the motor. Thus, problems that may occur can be prevented by early notification.

Finally, it would be wrong to reduce this project to jet aircraft engines only. This could be an important project for the automotive industry in the future. Furthermore, we can imagine that from the white goods to the computer we use at home, an early warning system can be created.

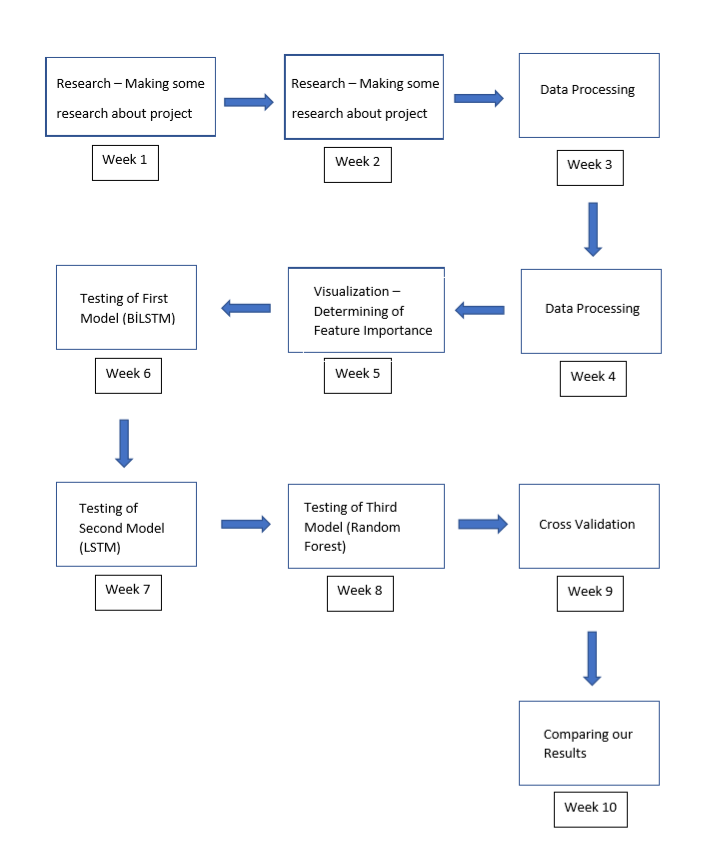
There are no restrictions or regulations against this project.

6. **ETHICAL ISSUES**

Our project has no adverse effects on health and the environment. There may be only one security problem. Engine data can be easily accessed if the system is exposed to cyberattacks. This means access to the jet aircraft's confidential information. Possibly it can cause trouble in engine supply.

Although it is very unlikely, access to the knowledge of jet aircraft engines can be misused by some states or terrorist organizations.

**7.** **PROJECT MANAGEMENT**

**

**8.** **CONCLUSION AND FUTURE WORK**

*As for the challenges, obstacles, it can be difficult to obtain real-time data. We used artificial data to do this, but real-time data is not easy to obtain.*

We have carried out a particularly economically efficient project that gives good accuracy in predicting the engine life of aircraft. Of course, our project can be developed to get better scores. First, our model can be further optimized. Thus, the accuracy rate can be increased. In addition, it can be used not only for jet planes but also for transportation vehicles such as passenger planes and helicopters.

This system can be developed from the automotive sector to the technologies we use at home (such as television). In this way, as we will have data on the remaining engine life of the car we will buy in automotive purchases, car purchases and sales can become more transparent. In short, we can develop our project for almost any product.

**9.** **APPENDIX**

**10.** **REFERENCES**

**1.**[**https://medium.com/@gulcanogundur/overfitting-a%C5%9F%C4%B1r%C4%B1-%C3%B6%C4%9Frenme-underfitting-eksik-%C3%B6%C4%9Frenme-ve-bias-variance-%C3%A7eli%C5%9Fkisi-b92bef2f770d**](https://medium.com/@gulcanogundur/overfitting-a%C5%9F%C4%B1r%C4%B1-%C3%B6%C4%9Frenme-underfitting-eksik-%C3%B6%C4%9Frenme-ve-bias-variance-%C3%A7eli%C5%9Fkisi-b92bef2f770d)

**2.**

[**https://medium.com/@gulcanogundur/overfitting-a%C5%9F%C4%B1r%C4%B1-%C3%B6%C4%9Frenme-underfitting-eksik-%C3%B6%C4%9Frenme-ve-bias-variance-%C3%A7eli%C5%9Fkisi-b92bef2f770d#:~:text=A%C5%9F%C4%B1r%C4%B1%20%C3%B6%C4%9Frenmenin%20aksine%2C%20bir%20model,verilerdeki%20trendleri%20ka%C3%A7%C4%B1rd%C4%B1%C4%9F%C4%B1%20anlam%C4%B1na%20gelir.&text=Underfitting%20sorunu%20olan%20modellerde%20hem,ve%20y%C3%BCksek%20bias'a%20sahiptir**](https://medium.com/@gulcanogundur/overfitting-a%C5%9F%C4%B1r%C4%B1-%C3%B6%C4%9Frenme-underfitting-eksik-%C3%B6%C4%9Frenme-ve-bias-variance-%C3%A7eli%C5%9Fkisi-b92bef2f770d#:~:text=A%C5%9F%C4%B1r%C4%B1%20%C3%B6%C4%9Frenmenin%20aksine%2C%20bir%20model,verilerdeki%20trendleri%20ka%C3%A7%C4%B1rd%C4%B1%C4%9F%C4%B1%20anlam%C4%B1na%20gelir.&text=Underfitting%20sorunu%20olan%20modellerde%20hem,ve%20y%C3%BCksek%20bias'a%20sahiptir)**.**

**3.**[**https://www.hindawi.com/journals/complexity/2018/3813029/#copyright**](https://www.hindawi.com/journals/complexity/2018/3813029/#copyright)

**4. Nieto, PJ García, et al. "Hybrid PSO–SVM-based method for forecasting of the remaining useful life for aircraft engines and evaluation of its reliability." *Reliability Engineering & System Safety* 138 (2015): 219-231.**

**5**.Yoshua Bengio, et al., [Learning Long-Term Dependencies with Gradient Descent is Difficult](http://www-dsi.ing.unifi.it/~paolo/ps/tnn-94-gradient.pdf), 1994.

**6.**Alex Graves, et al., [Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures](http://ieeexplore.ieee.org/document/1556215/?reload=true&arnumber=1556215), 2005.

**7.**Felix A. Gers, et al., [Learning to Forget: Continual Prediction with LSTM](http://www.mitpressjournals.org/doi/abs/10.1162/089976600300015015), 2000