# Iris Data Analysis Using ML Concepts

# Submitted By :

# Abhishek Kumar

Project Guide : Sofikul Mallick

Start Date : AUGUST 18, 2020

End Date : S E P T E M B E R 19 , 2020



**CERTIFICATE**

****

This is to certify that **Abhishek Kumar** successfully completed the project titled "A study on Iris Data Analysis using ML Concepts" under my supervision during the period from 18 AUGUST 2020 to 19 SEPTEMBER 2020 which is in fulfillment of their training in Data Science and Machine learning.

Signature of the Supervisor

Date: SEPTEMBER 19 , 2020 Sofikul Mallick

**CONTENTS:-**

1. Acknowledgment
2. Abstract
3. Introduction
   1. 3.1 Problem statement
   2. 3.2 Goal of competition
4. Required statistical concepts For this project
   1. 4.1 Decision tree
   2. 4.2 Types of decision tree
   3. random forest
5. Approach for the model building
   1. 5.1 Develop the analysis plan
   2. 5.2 Evaluation of underlying techniques
   3. 5.3 Estimate the model and assess overall model fit
   4. 5.4 Interpret the variate
   5. 5.5 Validate the model
6. Methodology
7. Project objective
8. Project goal
9. Detail architecture of work Flow
10. Overview
11. How random forests work
12. The out-of-bag error Estimate
13. Algorithm
14. Project implementation
15. Step-by-step working
    1. 15.1 Loading the libraries
    2. 15.2 Getting data
    3. 15.3 Data Visualization
    4. 15.4 Model performance and evaluation
    5. 15.5 Code Implementation
16. Summary 17.Bibiliongraphy

**ACKNOWLEDGEMENT**

The achievement that is associated with the successful completion of any task would be incomplete without mentioning the names of those people whose endless cooperation made it possible. Their constant guidance and encouragement made all our efforts successful.

We take this opportunity to express our deep gratitude towards our project mentor ***Sofikul Mallick***, for giving such valuable suggestions, guidance and encouragement during the development of this project work.

# ABSTRACT

The report presents the three tasks completed during summer training Which are listed below:

1. Understand of the Problem objective & business implication
2. Understanding the data & build the model .
3. Evaluation of the model.

All these tasks have been completed successfully and results were according to Expectations. All the tasks were need very systematic approach, starting from the behavior of the data to the application of the algorithm and till evaluation of the model. The most challenging task was the domain knowledge, to understand the behavior of the data. Once the data has been prepared, we applied statistical algorithm for model building. It is one of the major area and really need very fundamental and conceptual knowledge of Advanced Statistics.

# Introduction

This project is taken from kaggle. It is a platform for predictive modeling and Analytics competitions.

Here organization and researchers post the data. Statisticians and data scientist from all over the world compete to produce the best models.

## Problem Statement :-

The model for Iris Flower Species System is described. Existing iris flower dataset is preloaded in MATLAB and is used for clustering into three different species. The dataset is clustered using the k-means algorithm and neural network clustering tool in MATLAB. Neural network clustering tool is mainly used for clustering large data set without any supervision. It is also used for pattern recognition, feature extraction, vector quantization, image segmentation, function approximation, and data mining. Results/Findings: The results include the clustered iris dataset into three species without any supervision.

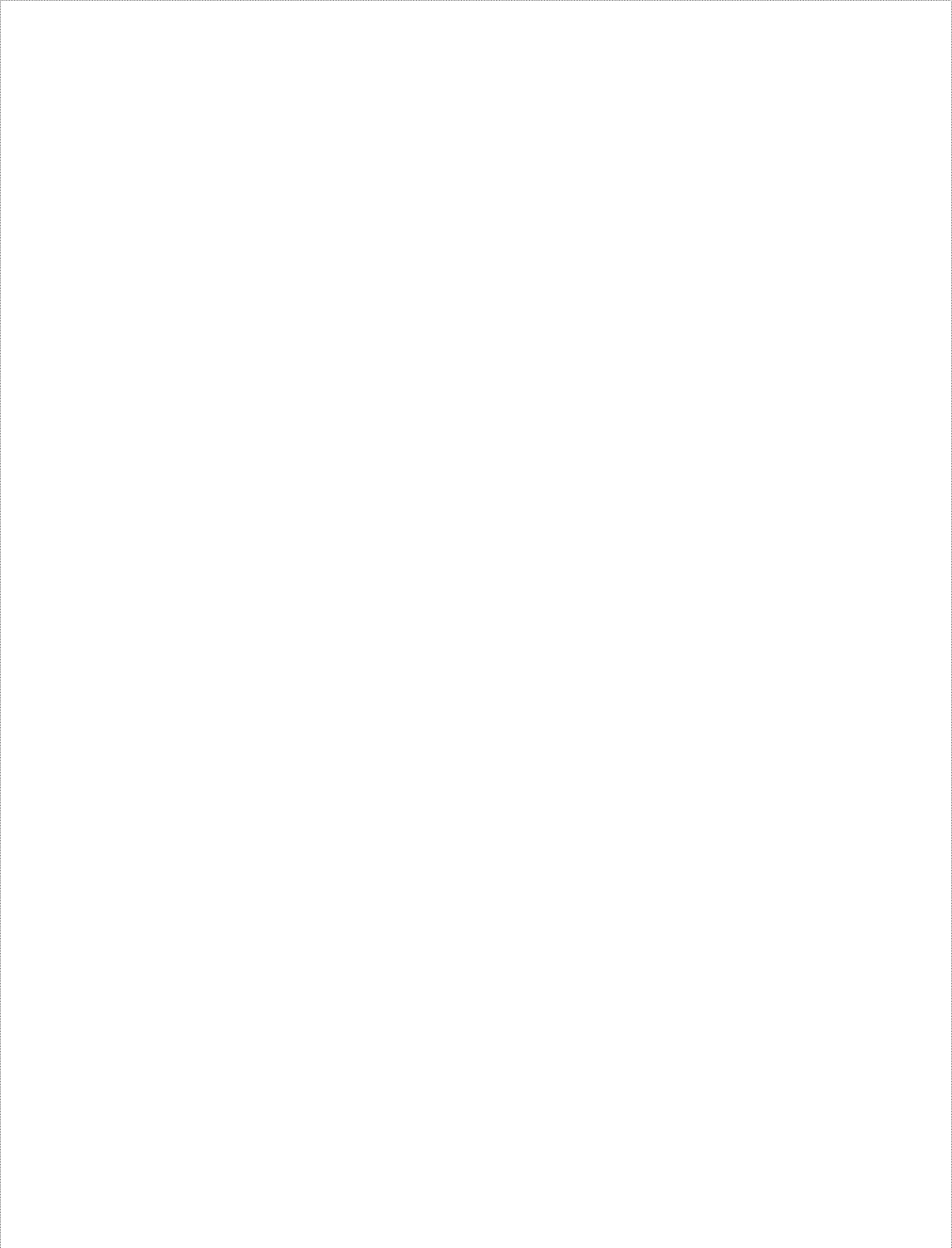
## Goal of the Competition :-

We need to build a machine learning model to accurately predict whether or not the patients in the dataset have diabetes or not?

## Required statistical concepts for this project

**Decision Tree :-**

**Decision tree** can be used to predict a pattern or to classify the class of a data. It is commonly used in data mining. The goal to use the decision tree algorithm is to create the model that predict the target variable based upon the several input variables. In decision tree each leaf represents a value of target variable given the value of input variables represented by the path from the root of the

leaf. A tree can be learned by splitting the source set into the subset based on an attribute value test. This process repeated in each derived subset called recursive. The general fashion for tree is top down induction.

**Decision tree used in data mining are of two main types:-**

#### Classification Tree:-

When the predicted outcome is the class to which the data belongs.

#### Regression Tree:-

When the predicted outcome can be consider a real number.

The term classification & regression tree(CART) analysis is an umbrella term used to refer to both of the above procedure.

# Approach for the model building

### Develop the Analysis Plan :-

For the conceptual model establishment, we need to understand the selected techniques and model implementation issue. Here we will establish predictive model, which will be in based on random forest algorithm. This will our model consideration based upon sample size and required type of variable(metric versus nonmetric).

**Evaluation of Underlying techniques :-**

Since all multivariate techniques rely on underlying assumption, both statistical & conceptual, that substantially affect their ability to represent multivariate relationships. For the techniques based on the statistical inference, the assumptions of the multivariate normality, linearity, independence of the error term, and equality of variance in a dependence relationship must all be met. Since our data is categorical so we do not need to identify the linearity or any independence relation.

## Estimate the Model and Assess Overall Model Fit :-

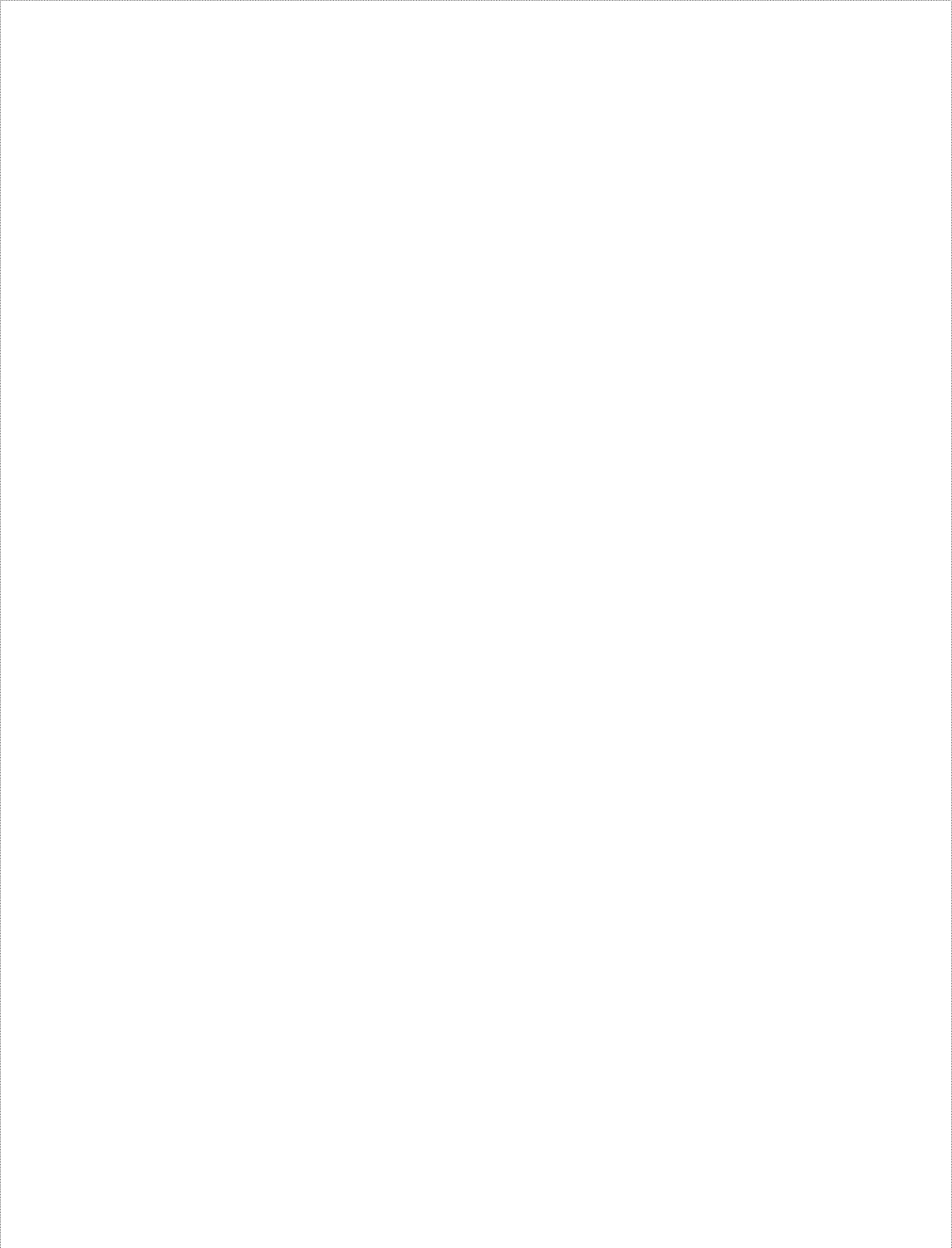
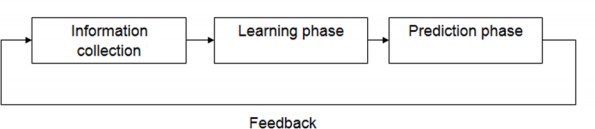
With the assumption satisfied, the analysis proceeds to the actual estimation of the model and assessment of overall model fit. In the estimation process we can choose option to meet specific characteristics of the data or to maximize the fit of data. After the model is estimated, the overall model fit is evaluated to ascertain whether it achieves acceptable levels on statistical criteria. Many times, the model will be specified in an attempt to better level of overall fit or explanation.

## Interpret the variate : -

With the acceptable level of model fit, interpreting the variate reveals the nature of relationship. The interpretation of effects for individual variables is made by examining the estimated weight for each variable in the variate.

## Validate the Model :-

Before accepting the results, we must subject them to one final set of diagnostic analyses that assess the degree of generalization of the results by the available validation methods. The attempt to validate the model is directed towards demonstrating the generalization of the results to the total population.



These diagnostic analyses add little to the

interpretation of the results but can be viewed as insurance that the results are the most descriptive of the data.

# Methodology

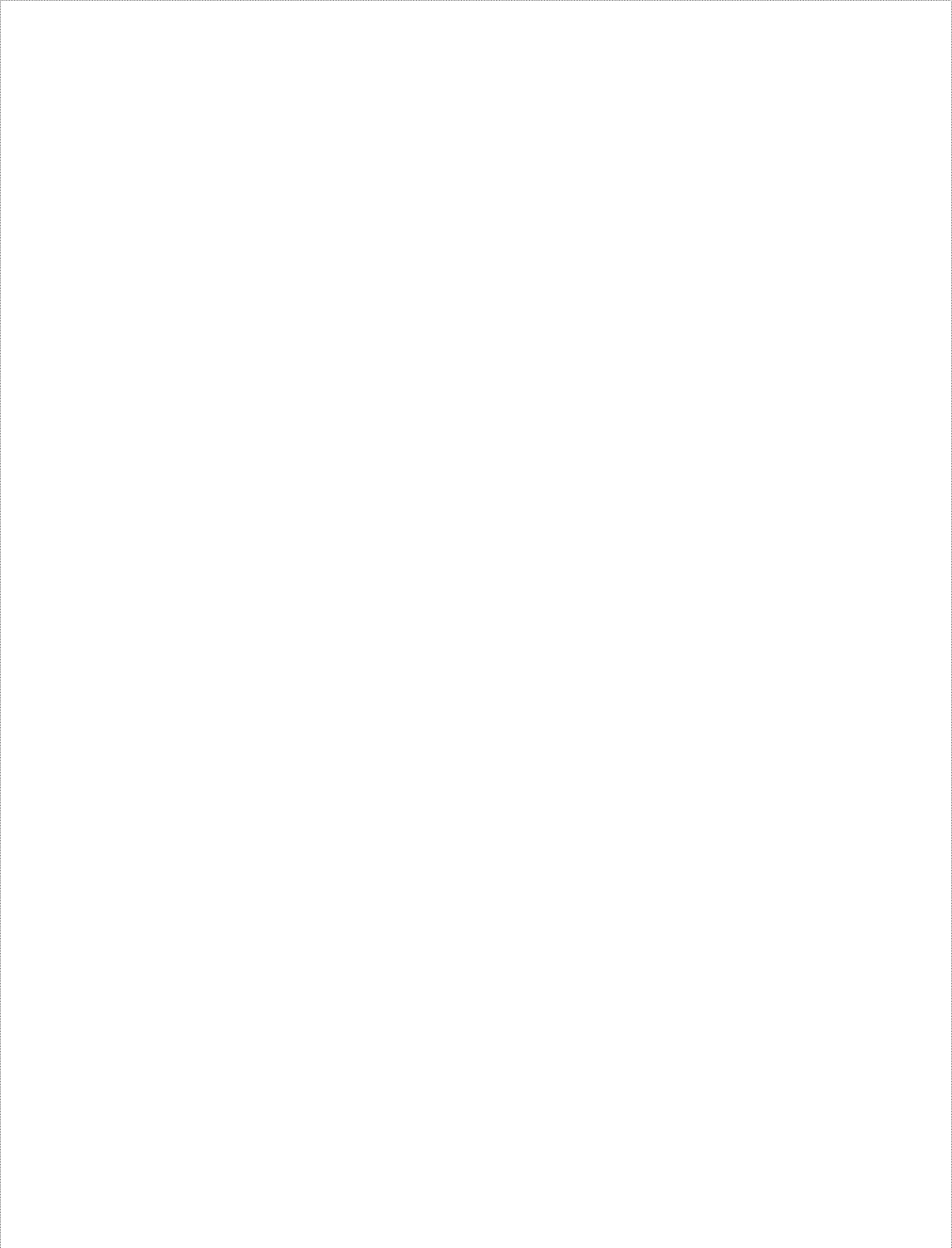
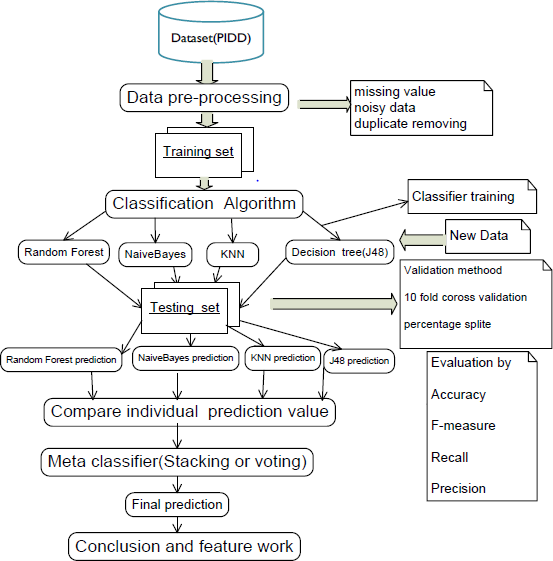
**Project Objective**

The objective of the project is :-

* To study about the iris data using machine learning
* Extraction of important variable from the dataset useful for species prediction.
* Data Visualization for the better prediction results

# Project Goal

Goal of this analysis study is to predict and research and measure various features of the iris flowers and recorded digitally using algorithm which provides high accuracy . Finally select the best algorithm to measure feature of different iris flowers.



###### Detail Architecture of work flow

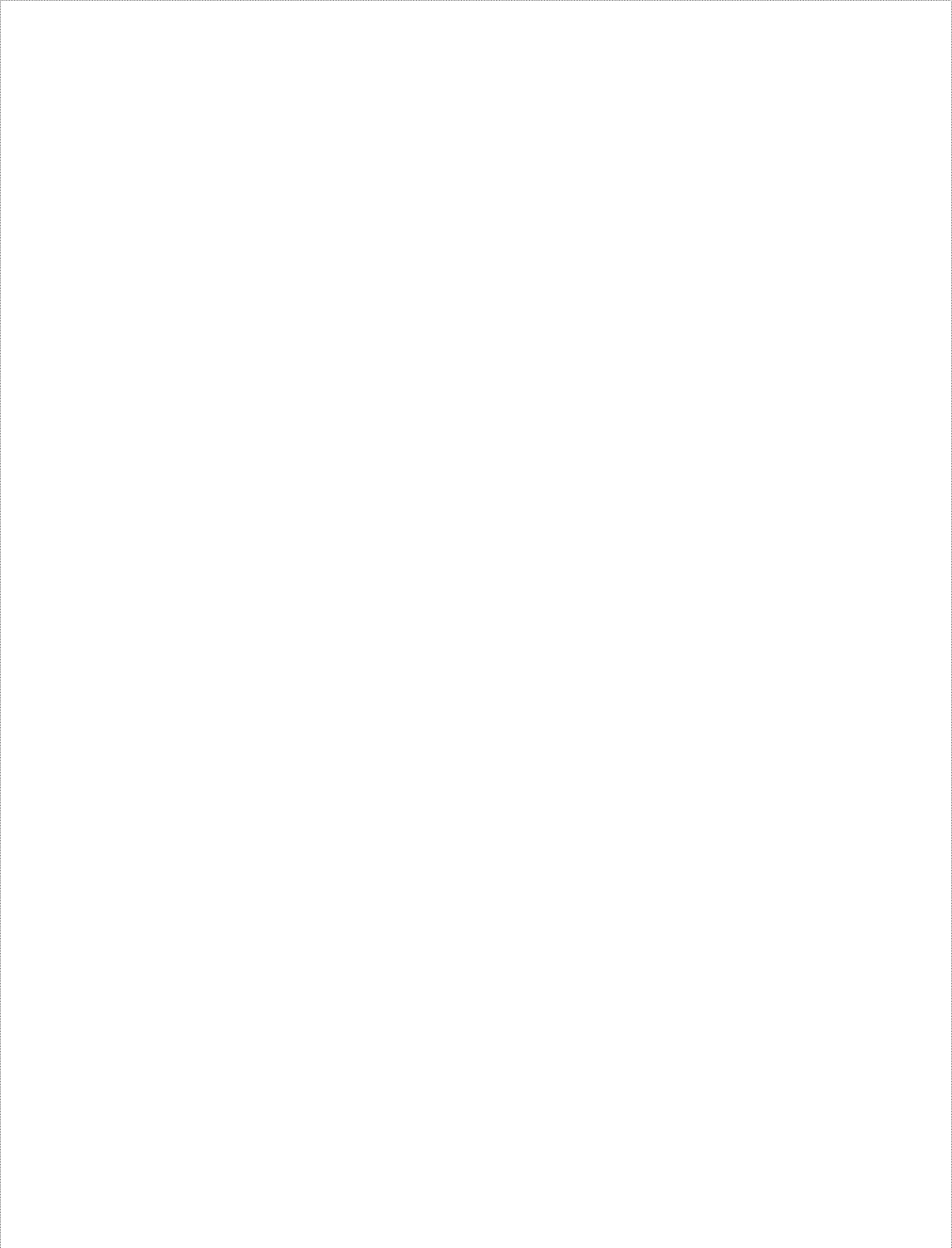
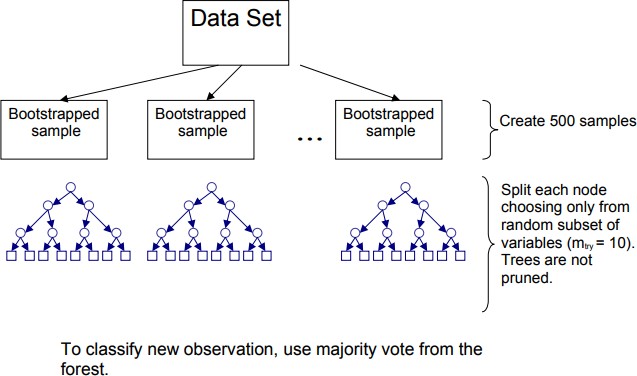
**Overview**

**The forest error rate depends on two things:**

* The **correlation** between any two trees in the forest. Increasing the correlation increases the forest error rate
* The **strength** of each individual tree in the forest. A tree with a low error rate is a strong classifier. Increasing the strength of the individual trees decreases the forest error rate

.

Reducing m reduces both the correlation and the strength. Increasing it increases both. Somewhere in between is an "optimal" range of m - usually quite wide. This is the only adjustable parameter to which random forests is somewhat sensitive.



**How Random Forests Work**

##### When the training set for the current tree is drawn by sampling with replacement, about one-third of the cases are left out of the sample. This oob (out of bag) data is used to get a running unbiased estimate of the classification error as trees are added to the forest. It is also used to get estimates of variable importance.

After each tree is built, all of the data are run down the tree, and proximities are computed for each pair of cases. If two cases occupy the same terminal node, their proximity is increased by one. At the end of the run, the proximities are normalized by dividing by the number of trees.

Proximities are used in replacing missing data, locating outliers, and producing illuminating low-dimensional views of the data.

**The out-of-bag (oob) error estimate**

In random forests, there is no need for cross- validation or a separate test set to get an unbiased estimate of the test set error.

###### It is estimated internally, during the run, as follows:

* Each tree is constructed using a different bootstrap sample from the original data. About one third of the cases are left out of the bootstrap sample and not used in the construction of the Kth tree.
  + Put each case left out in the construction of the kth tree down the kth tree to get a classification. In this way, a test set classification is obtained for each case in about one-third of the trees. At the end of the run, take j to be the class that got most of the votes every time case n was oob. The proportion of times that j is not equal to the true class of n averaged over all cases is the oob error estimate. This has proven to be unbiased in many tests.

The Algorithm:-

###### The random forests algorithm (for both classification and regression) is as follows:

1. Draw ntree bootstrap samples from the original data.
2. For each of the bootstrap samples, grow an unpruned classification or regression tree, with the following modification: at each node, rather than choosing the best split among all predictors, randomly sample mtry of the predictors and choose the best split from among those Variables. (Bagging can be thought of as the special case of random forests obtained when mtry = p, the number of predictors.)
3. Predict new data by aggregating the predictions of the ntree trees (i.e., majority votes for Classification, Average for regression).

###### An estimate of the error rate can be obtained, based on the training data, by the following:

1. At each bootstrap iteration, predict the data not in the bootstrap sample (what Breiman Calls “out-of- bag”, or OOB, data) using the tree grown with the bootstrap sample.
2. Aggregate the OOB predictions. (On the average, each data point would be out-of-bag around 36% of the times, so aggregate these predictions.) Calculate the error rate, and call it the OOB estimate of error rate.

# Project Implementation

Following Libraries are Used For the Project:-

### Library(numpy)

Description: NumPy stands for Numerical Python. NumPy is a python library used for working with arrays. It also has functions for working in the domain of linear algebra, fourier transform, and matrices.

### Libaray(pandas)

Description: pandas is a sohware library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.

### Library(matplotlib)

Description: Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy It provides an object-oriented API for embedding plots into applications using general-purpose GUI.

### Library(seaborn)

Description: Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics

Step by step working:-

1. **Loading the Libraries**

library(numpy) library(pandas) library(matplotlib) library(seaborn)

**2. Getting Data**

Downloading the data set into the hard drive and import it to the RStudio

library(read table)

iris\_data = pd.read\_csv("SVM\_IRIS.csv")

iris\_data.head()

View(iris data)

| **sepal\_length** | **sepal\_width** | **petal\_length** | **petal\_width** | **species** |
| --- | --- | --- | --- | --- |
| **0** | 5.1 | 3.5 | 1.4 | 0.2 | Iris-setosa |
| **1** | 4.9 | 3.0 | 1.4 | 0.2 | Iris-setosa |
| **2** | 4.7 | 3.2 | 1.3 | 0.2 | Iris-setosa |
| **3** | 4.6 | 3.1 | 1.5 | 0.2 | Iris-setosa |
| **4** | 5.0 | 3.6 | 1.4 | 0.2 | Iris-setosa |

The raw dataset contained 150 rows of data, with 6 variables.

# 3. Data Visualization

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | Id | SepalLengthCm | SepalWidthCm | PetalLengthCm | PetalWidthCm | Species |  | | 1 | 5.1 | 3.5 | 1.4 | 0.2 | Iris-setosa | | | 2 | 4.9 | 3 | 1.4 | 0.2 | Iris-setosa | | | 3 | 4.7 | 3.2 | 1.3 | 0.2 | Iris-setosa | | | 4 | 4.6 | 3.1 | 1.5 | 0.2 | Iris-setosa | | | 5 | 5 | 3.6 | 1.4 | 0.2 | Iris-setosa | | | 6 | 5.4 | 3.9 | 1.7 | 0.4 | Iris-setosa | | | 7 | 4.6 | 3.4 | 1.4 | 0.3 | Iris-setosa | | | 8 | 5 | 3.4 | 1.5 | 0.2 | Iris-setosa | | | 9 | 4.4 | 2.9 | 1.4 | 0.2 | Iris-setosa | | | 10 | 4.9 | 3.1 | 1.5 | 0.1 | Iris-setosa | | | 11 | 5.4 | 3.7 | 1.5 | 0.2 | Iris-setosa | | | 12 | 4.8 | 3.4 | 1.6 | 0.2 | Iris-setosa | | | 13 | 4.8 | 3 | 1.4 | 0.1 | Iris-setosa | | | 14 | 4.3 | 3 | 1.1 | 0.1 | Iris-setosa | | | 15 | 5.8 | 4 | 1.2 | 0.2 | Iris-setosa | | | 16 | 5.7 | 4.4 | 1.5 | 0.4 | Iris-setosa | | | 17 | 5.4 | 3.9 | 1.3 | 0.4 | Iris-setosa | | | 18 | 5.1 | 3.5 | 1.4 | 0.3 | Iris-setosa | | | 19 | 5.7 | 3.8 | 1.7 | 0.3 | Iris-setosa | | | 20 | 5.1 | 3.8 | 1.5 | 0.3 | Iris-setosa | | | 21 | 5.4 | 3.4 | 1.7 | 0.2 | Iris-setosa | | | 22 | 5.1 | 3.7 | 1.5 | 0.4 | Iris-setosa | | | 23 | 4.6 | 3.6 | 1 | 0.2 | Iris-setosa | | | 24 | 5.1 | 3.3 | 1.7 | 0.5 | Iris-setosa | | | 25 | 4.8 | 3.4 | 1.9 | 0.2 | Iris-setosa | | | 26 | 5 | 3 | 1.6 | 0.2 | Iris-setosa | | | 27 | 5 | 3.4 | 1.6 | 0.4 | Iris-setosa | | | 28 | 5.2 | 3.5 | 1.5 | 0.2 | Iris-setosa | | | 29 | 5.2 | 3.4 | 1.4 | 0.2 | Iris-setosa | | | 30 | 4.7 | 3.2 | 1.6 | 0.2 | Iris-setosa | | | 31 | 4.8 | 3.1 | 1.6 | 0.2 | Iris-setosa | | | 32 | 5.4 | 3.4 | 1.5 | 0.4 | Iris-setosa | | | 33 | 5.2 | 4.1 | 1.5 | 0.1 | Iris-setosa | | | 34 | 5.5 | 4.2 | 1.4 | 0.2 | Iris-setosa | | | 35 | 4.9 | 3.1 | 1.5 | 0.1 | Iris-setosa | | | 36 | 5 | 3.2 | 1.2 | 0.2 | Iris-setosa | | | 37 | 5.5 | 3.5 | 1.3 | 0.2 | Iris-setosa | | | 38 | 4.9 | 3.1 | 1.5 | 0.1 | Iris-setosa | | | 39 | 4.4 | 3 | 1.3 | 0.2 | Iris-setosa | | | 40 | 5.1 | 3.4 | 1.5 | 0.2 | Iris-setosa | | | 41 | 5 | 3.5 | 1.3 | 0.3 | Iris-setosa | | | 42 | 4.5 | 2.3 | 1.3 | 0.3 | Iris-setosa | | | 43 | 4.4 | 3.2 | 1.3 | 0.2 | Iris-setosa | | | 44 | 5 | 3.5 | 1.6 | 0.6 | Iris-setosa | | | 45 | 5.1 | 3.8 | 1.9 | 0.4 | Iris-setosa | | | 46 | 4.8 | 3 | 1.4 | 0.3 | Iris-setosa | | | 47 | 5.1 | 3.8 | 1.6 | 0.2 | Iris-setosa | | | 48 | 4.6 | 3.2 | 1.4 | 0.2 | Iris-setosa | | | 49 | 5.3 | 3.7 | 1.5 | 0.2 | Iris-setosa | | | 50 | 5 | 3.3 | 1.4 | 0.2 | Iris-setosa | | | 51 | 7 | 3.2 | 4.7 | 1.4 | Iris-versicolor | | | 52 | 6.4 | 3.2 | 4.5 | 1.5 | Iris-versicolor | | | 53 | 6.9 | 3.1 | 4.9 | 1.5 | Iris-versicolor | | | 54 | 5.5 | 2.3 | 4 | 1.3 | Iris-versicolor | | | 55 | 6.5 | 2.8 | 4.6 | 1.5 | Iris-versicolor | | | 56 | 5.7 | 2.8 | 4.5 | 1.3 | Iris-versicolor | | | 57 | 6.3 | 3.3 | 4.7 | 1.6 | Iris-versicolor | | | 58 | 4.9 | 2.4 | 3.3 | 1 | Iris-versicolor | | | 59 | 6.6 | 2.9 | 4.6 | 1.3 | Iris-versicolor | | | 60 | 5.2 | 2.7 | 3.9 | 1.4 | Iris-versicolor | | | 61 | 5 | 2 | 3.5 | 1 | Iris-versicolor | | | 62 | 5.9 | 3 | 4.2 | 1.5 | Iris-versicolor | | | 63 | 6 | 2.2 | 4 | 1 | Iris-versicolor | | | 64 | 6.1 | 2.9 | 4.7 | 1.4 | Iris-versicolor | | | 65 | 5.6 | 2.9 | 3.6 | 1.3 | Iris-versicolor | | | 66 | 6.7 | 3.1 | 4.4 | 1.4 | Iris-versicolor | | | 67 | 5.6 | 3 | 4.5 | 1.5 | Iris-versicolor | | | 68 | 5.8 | 2.7 | 4.1 | 1 | Iris-versicolor | | | 69 | 6.2 | 2.2 | 4.5 | 1.5 | Iris-versicolor | | | 70 | 5.6 | 2.5 | 3.9 | 1.1 | Iris-versicolor | | | 71 | 5.9 | 3.2 | 4.8 | 1.8 | Iris-versicolor | | | 72 | 6.1 | 2.8 | 4 | 1.3 | Iris-versicolor | | | 73 | 6.3 | 2.5 | 4.9 | 1.5 | Iris-versicolor | | | 74 | 6.1 | 2.8 | 4.7 | 1.2 | Iris-versicolor | | | 75 | 6.4 | 2.9 | 4.3 | 1.3 | Iris-versicolor | | | 76 | 6.6 | 3 | 4.4 | 1.4 | Iris-versicolor | | | 77 | 6.8 | 2.8 | 4.8 | 1.4 | Iris-versicolor | | | 78 | 6.7 | 3 | 5 | 1.7 | Iris-versicolor | | | 79 | 6 | 2.9 | 4.5 | 1.5 | Iris-versicolor | | | 80 | 5.7 | 2.6 | 3.5 | 1 | Iris-versicolor | | | 81 | 5.5 | 2.4 | 3.8 | 1.1 | Iris-versicolor | | | 82 | 5.5 | 2.4 | 3.7 | 1 | Iris-versicolor | | | 83 | 5.8 | 2.7 | 3.9 | 1.2 | Iris-versicolor | | | 84 | 6 | 2.7 | 5.1 | 1.6 | Iris-versicolor | | | 85 | 5.4 | 3 | 4.5 | 1.5 | Iris-versicolor | | | 86 | 6 | 3.4 | 4.5 | 1.6 | Iris-versicolor | | | 87 | 6.7 | 3.1 | 4.7 | 1.5 | Iris-versicolor | | | 88 | 6.3 | 2.3 | 4.4 | 1.3 | Iris-versicolor | | | 89 | 5.6 | 3 | 4.1 | 1.3 | Iris-versicolor | | | 90 | 5.5 | 2.5 | 4 | 1.3 | Iris-versicolor | | | 91 | 5.5 | 2.6 | 4.4 | 1.2 | Iris-versicolor | | | 92 | 6.1 | 3 | 4.6 | 1.4 | Iris-versicolor | | | 93 | 5.8 | 2.6 | 4 | 1.2 | Iris-versicolor | | | 94 | 5 | 2.3 | 3.3 | 1 | Iris-versicolor | | | 95 | 5.6 | 2.7 | 4.2 | 1.3 | Iris-versicolor | | | 96 | 5.7 | 3 | 4.2 | 1.2 | Iris-versicolor | | | 97 | 5.7 | 2.9 | 4.2 | 1.3 | Iris-versicolor | | | 98 | 6.2 | 2.9 | 4.3 | 1.3 | Iris-versicolor | | | 99 | 5.1 | 2.5 | 3 | 1.1 | Iris-versicolor | | | 100 | 5.7 | 2.8 | 4.1 | 1.3 | Iris-versicolor | | | 101 | 6.3 | 3.3 | 6 | 2.5 | Iris-virginica | | | 102 | 5.8 | 2.7 | 5.1 | 1.9 | Iris-virginica | | | 103 | 7.1 | 3 | 5.9 | 2.1 | Iris-virginica | | | 104 | 6.3 | 2.9 | 5.6 | 1.8 | Iris-virginica | | | 105 | 6.5 | 3 | 5.8 | 2.2 | Iris-virginica | | | 106 | 7.6 | 3 | 6.6 | 2.1 | Iris-virginica | | | 107 | 4.9 | 2.5 | 4.5 | 1.7 | Iris-virginica | | | 108 | 7.3 | 2.9 | 6.3 | 1.8 | Iris-virginica | | | 109 | 6.7 | 2.5 | 5.8 | 1.8 | Iris-virginica | | | 110 | 7.2 | 3.6 | 6.1 | 2.5 | Iris-virginica | | | 111 | 6.5 | 3.2 | 5.1 | 2 | Iris-virginica | | | 112 | 6.4 | 2.7 | 5.3 | 1.9 | Iris-virginica | | | 113 | 6.8 | 3 | 5.5 | 2.1 | Iris-virginica | | | 114 | 5.7 | 2.5 | 5 | 2 | Iris-virginica | | | 115 | 5.8 | 2.8 | 5.1 | 2.4 | Iris-virginica | | | 116 | 6.4 | 3.2 | 5.3 | 2.3 | Iris-virginica | | | 117 | 6.5 | 3 | 5.5 | 1.8 | Iris-virginica | | | 118 | 7.7 | 3.8 | 6.7 | 2.2 | Iris-virginica | | | 119 | 7.7 | 2.6 | 6.9 | 2.3 | Iris-virginica | | | 120 | 6 | 2.2 | 5 | 1.5 | Iris-virginica | | | 121 | 6.9 | 3.2 | 5.7 | 2.3 | Iris-virginica | | | 122 | 5.6 | 2.8 | 4.9 | 2 | Iris-virginica | | | 123 | 7.7 | 2.8 | 6.7 | 2 | Iris-virginica | | | 124 | 6.3 | 2.7 | 4.9 | 1.8 | Iris-virginica | | | 125 | 6.7 | 3.3 | 5.7 | 2.1 | Iris-virginica | | | 126 | 7.2 | 3.2 | 6 | 1.8 | Iris-virginica | | | 127 | 6.2 | 2.8 | 4.8 | 1.8 | Iris-virginica | | | 128 | 6.1 | 3 | 4.9 | 1.8 | Iris-virginica | | | 129 | 6.4 | 2.8 | 5.6 | 2.1 | Iris-virginica | | | 130 | 7.2 | 3 | 5.8 | 1.6 | Iris-virginica | | | 131 | 7.4 | 2.8 | 6.1 | 1.9 | Iris-virginica | | | 132 | 7.9 | 3.8 | 6.4 | 2 | Iris-virginica | | | 133 | 6.4 | 2.8 | 5.6 | 2.2 | Iris-virginica | | | 134 | 6.3 | 2.8 | 5.1 | 1.5 | Iris-virginica | | | 135 | 6.1 | 2.6 | 5.6 | 1.4 | Iris-virginica | | | 136 | 7.7 | 3 | 6.1 | 2.3 | Iris-virginica | | | 137 | 6.3 | 3.4 | 5.6 | 2.4 | Iris-virginica | | | 138 | 6.4 | 3.1 | 5.5 | 1.8 | Iris-virginica | | | 139 | 6 | 3 | 4.8 | 1.8 | Iris-virginica | | | 140 | 6.9 | 3.1 | 5.4 | 2.1 | Iris-virginica | | | 141 | 6.7 | 3.1 | 5.6 | 2.4 | Iris-virginica | | | 142 | 6.9 | 3.1 | 5.1 | 2.3 | Iris-virginica | | | 143 | 5.8 | 2.7 | 5.1 | 1.9 | Iris-virginica | | | 144 | 6.8 | 3.2 | 5.9 | 2.3 | Iris-virginica | | | 145 | 6.7 | 3.3 | 5.7 | 2.5 | Iris-virginica | | | 146 | 6.7 | 3 | 5.2 | 2.3 | Iris-virginica | | | 147 | 6.3 | 2.5 | 5 | 1.9 | Iris-virginica | | | 148 | 6.5 | 3 | 5.2 | 2 | Iris-virginica | | | 149 | 6.2 | 3.4 | 5.4 | 2.3 | Iris-virginica | | | 150 | 5.9 | 3 | 5.1 | 1.8 | Iris-virginica | | |  |  |  |  |  |
|  |  |  |  |  | |
|  |  |  |  |  | |

# 4.Model performance and evaluation

Evaluating the performance of a model is one of the core stages in the data science process. It indicates how successful the scoring (predictions) of a dataset has been by a trained model.

5.Code Implementaion

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

%matplotlib inline

iris\_data = pd.read\_csv("SVM\_IRIS.csv")

iris\_data.head()

X = iris\_data[["sepal\_length", "sepal\_width", "petal\_length", "petal\_width"]]

X.head()

y = iris\_data["species"]

y.head()

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.1)

from sklearn.svm import SVC

sm = SVC()

sm.fit(X\_train, y\_train)

y\_pred = sm.predict(X\_test)

y\_pred

sm.predict([[7, 7, 7, 6]])

from sklearn.metrics import classification\_report, confusion\_matrix, accuracy\_score

confusion\_matrix(y\_test, y\_pred)

print(classification\_report(y\_test, y\_pred))

accuracy\_score(y\_test, y\_pred)

# Summary

We explored the Iris dataset, and then built a few popular classifiers using different library functions. We saw that the petal measurements are more helpful at classifying instances than the sepal ones. Furthermore, most models achieved a test accuracy of over 95%.

# Bibliography

* + <http://www.blopig.com/blog/2017/04/a-very-> basic-introduction-to-random-forests-using-r/
  + https://[www.youtube.com/watch?v=nKW8Ndu7](http://www.youtube.com/watch?v=nKW8Ndu7) Mjw
  + https://martechtoday.com/how-machine-learning- works-150366
  + [www.blopig.com/blog/2017/04/a-very-basic-](http://www.blopig.com/blog/2017/04/a-very-basic-) introduction-to-random-forests-using-r/
  + https[://w](http://www.youtube.com/watch?v=4cH8PzA3ov)ww[.y](http://www.youtube.com/watch?v=4cH8PzA3ov)o[utube.com/watch?v=4cH8PzA3ov](http://www.youtube.com/watch?v=4cH8PzA3ov) 0&t=447s
  + https://[www.youtube.com/watch?v=Fn1t4rEIf\_8](http://www.youtube.com/watch?v=Fn1t4rEIf_8)