

Assessing Impact of Digital Lens Usage on Eye Dryness using Schirmer's Effect

Project report submitted for
Industrial Project Based Learning in Data Science and Machine
Learning

Under the Supervision of

P.Mohan

Director, Gyan Astra IT Solutions

&

Y.V.N Phani Kishore

Director, Gyan Astra IT Solutions

By

Team Members Name (Roll Number)

P Pranavi	21R11A6742
C Manognasri	21R11A6729
A. Sai Srujana	21R11A05A8
P Moulika	21R11A05E0

Department of Computer Science & Engineering

Nov 2023 - May 2024



Geethanjali College of Engineering &

Technology **Accredited by NBA (UGC**

Autonomous)

(Affiliated to J.N.T.U.H, Approved by AICTE, New Delhi, NAAC - A+)

Geethanjali College of Engineering & Technology

(Affiliated to J.N.T.U.H, Approved by AICTE, NEWDELHI.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that Mr./Ms. C.Manognasri, A.Sai Srujana, P Moulika , P Pranavi bearing roll no 21R11A6729,21R11A05A8,21R11A05E0, 21R11A6742 has successfully completed Industrial Project Based Learning in Data Science and Machine Learning held during November 2023 to May 2024 at Geethanjali college of Engineering and Technology.

Coordinator

Mr. E. Mahender

Assistant Professor

Dept of C.S.E

H.O.D - C.S.E

Dr. A. Sree Lakshmi

Professor & Head

Dept of C.S.E

ABSTRACT

By convention, Digital Eye Strain it is an effect that caused by the digital screens. These digital screens include mobiles, laptops, televisions. By using these gadgets over an extended period of time and prolonged screen usage can reduce the frequency of blinking the eyes and causes Eye strain, dry eyes, digital eye fatigue, redness, itchiness, Myopia as well.

By using the given data, we analyzed which factors are leading to dryness of our eyes. And also, we found that which factors are less affecting the eyes and which factors are more affecting the eyes. We tried different machine learning models by taking potential factors into consideration. We predicted the outputs by testing and validating the models. We finalized a model which gives maximum accuracy among all the models. These machine learning models helps us to know how much our eyes got affected by these digital screens.

TABLE OF CONTENTS

S.NO	TOPICS	PAGE NUMBERS
i	Abstract	i
1	Introduction	1
2	Literature survey	2
3	Problem statement	3
4	Objectives	3
5	Methodology	4
5.1	Data Collection	4
5.2	Importing dataset and required packages	6
5.3	Data Cleaning	7
5.4	Outlier detection	8
5.5	Outlier Handling	9
5.6	Handling Numeric Concatenation	11
5.7	Data Preprocessing	11
5.8	Exploratory Data Analysis	12
5.9	Feature Importance	16
6	Algorithms	17
7	Model Implementation	18
7.1	Logistic Regression	18
7.2	Decision tree	19
7.3	Random forest regressor	20
7.4	Decision Tree Regressor	21
7.5	Support Vector Regressor	21
7.6	Gradient Boosting Regressor	23
7.7	Decision tree Classifier	25
7.8	Building Web application	26
8	Result	30
9	Conclusion	33
10	Future Scope	33
11	References	34

LIST OF FIGURES

Fig No	Text	Page No
1.1	Schimers Tear Testing Procedure	1
5.2	Columns with Null values	6
5.3.1	Visualization of Null values	7
5.3.2	Visualization of Null values	8
5.4.1	Visualizing outliers using box plots for 1 st four columns	9
5.4.2	Visualizing outliers using box plots for last four columns	9
5.5.1	Visualization of outlier handling for 1 st four columns	10
5.5.2	Visualization of outlier handling for last four columns	10
5.6	Handling Numeric Concatenated values of columns	11
5.8.1	Distribution of Results	12
5.8.2	Frequency of values for target columns	13
5.8.3	Frequency of Categorical values of target variables	13
5.8.4	Relation between wearables	14
5.8.5	Representation of level of gadgets with respect to eyes	14
5.8.6	Correlation Heatmap	15
5.9	Feature Importance	16
7.1.1	Classification report of Logistic regression	18
7.1.2	Confusion Matrix of Logistic Regression	18
7.2.1	Decision tree	19
7.2.2	Classification report of Decision Tree	19
7.3.1	Random Forest Regressor	20
7.3.2	Metrics of Random Forest Regressor	20
7.3.3	R-squared error for of Random Forest Regressor	20
7.4	Decision Tree Regressor	21
7.5.1	Metrics of Support Vector Regressor	21
7.5.2	SVR Actual vs Predicted for Schimer's Test1	22
7.5.3	SVR Actual vs Predicted for Schimer's Test2	22
7.6.1	Metrics of Gradient Boosting Regressor	23
7.6.2	Actual vs Predicted values for Schimer's Test 1	23

7.6.3	Actual vs Predicted values for Schimer's Test 2	24
7.7.1	Decision Tree Classifier	25
7.7.2	Accuracy of Decision Tree Classifier	25
7.8.1	Web app Screen 1	27
7.8.2	Web app Screen 2	27
7.8.3	Web app Screen 3	28
7.8.4	Web app Screen 4	28
7.8.5	Web app Screen 5	28
7.8.6	Web app Screen 6	29
7.8.7	Running Flask application on localhost	29
8.1	Web address of Flask Application	30
8.2	Result 1	30
8.3	Result 2	31
8.4	Result 3	31
8.5	Result 4	32

1. INTRODUCTION

Digital screens effecting our eyes by continuously watching and over using them. The digital screens such as smartphones, tablets, laptops, and televisions, can have several effects on the eyes, and these effects are collectively known as ‘Digital Eye Strain’.

The common effects such as Eye strain the eyes starts strain when we use the screen for extending periods. And the eye strain can lead to dryness, redness, itchiness etc. The bright light and blue light from screens can make our eyes feel tired, dry, and strained. And some studies suggest that excessive screen time, especially during childhood may contribute to the development of worsening of myopia which is also called nearsightedness.

It is important to know how much our eyes got effected by these digital screens. So, to know that the doctors suggest a test called Schimers. A Schimer’s test determines whether a person’s eye produces enough tears to keep moist and healthy.

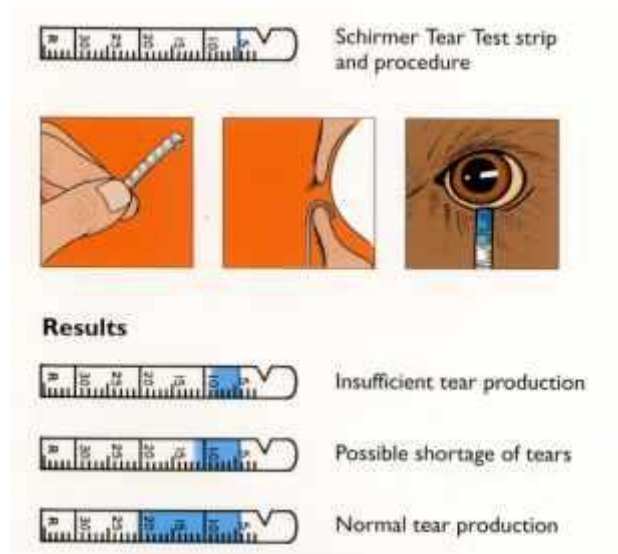


Fig 1.1 Schimers Tear Testing Procedure

To conduct a Schimer’s test a doctor places a piece of filter paper inside the lower eyes of both eyes and the person closes their eyes. After 5 minutes, the doctor removes the filter paper. The doctor then assesses how far the tears have traveled on the paper. The smaller the amount of moisture on the paper, the fewer tears that person has produced. The Schimer’s test confirms and determines the dry eye symptoms which include Severity of dryness in the eyes, Moderately dryness in the eyes, mildly dryness in the eyes and Normal. By Schimer’s test we get to know how much our eyes got effected and what are the habits that we need to adapt and to reduce the effect of digital screen usage on our eyes.

2. LITERATURE SURVEY

➤ Existing Research on Digital Eye Strain Prediction.

The research states that an individual who uses the digital screens for an extend period of time without frequently blinking the eyes are more likely affected with Eye Strain condition. The symptoms of eye strain include dry eyes, redness, itchiness, headache, near-sightedness etc. Here, we used machine learning models like Logistic regression. We transformed the target variables data into Normal or Abnormal to predict the condition of eye dryness. In this model we achieved an accuracy of 96.6%. And applied Label Encoding on target variables for better use of other models.

➤ Feature Selection and Model Building

To effectively predict the output the model should be trained with potential or key factors that affects the output. To select the potential features, we have done feature importance by using random forest regressor. This method gave us each column's variance percentage which contributes to target variable. The columns with more variance percentage are classified as potential factors and taken into consideration. By using these potential factors, we used decision tree model and we gained an accuracy of 86.6%. This model's accuracy is lower than previous one.

➤ Evaluation Metrics and Performance

Here, we tried Decision tree regressor and calculated the metrics like MSR and RMSR. And the RMSR values are negative indicates the model performance was poor. Next, we tried Support Vector Regressor and calculated the metrics, but the MSR values are in the range 40-60 which results the average performance of the model. The lower MSR value represents the highest performance of the model. Now, we used Gradient Boost Regressor model and calculated the metrics, these metrics gave us above average results. Then we transformed the target variables into categorical variables and applied Decision Tree Classifier model which gave us overall accuracy of 87%. We tested and validated all the above-mentioned models and the Decision Tree Classifier gave us more accurate results. Hence, we selected the Decision Tree Classifier Model.

3. PROBLEM STATEMENT

Understanding the impact of digital screen usage on eye health and associated symptoms involves analysing various factors like age, duration of screen time, online platforms, nature of activities, screen illumination, years of exposure, daily screen hours, types of devices used, distance from the screen, night-time usage, blinking frequency, difficulty in focusing, frequency and severity of complaints, observed ocular symptoms, and specific eye examination results.

By examining these variables, we aim to uncover patterns and correlations to develop strategies for maintain optimal eye health in the digital age.

4. OBJECTIVES

1. Data Collection and Description: Gathering a comprehensive dataset containing information on digital screen usage habits, Schirmer's test results, and demographic variables of individuals
2. Importing Dataset and Required Packages: Importing necessary Python libraries such as Pandas, NumPy, Matplotlib, Seaborn, and Scikit-learn for data manipulation, visualization, and machine learning model development.
3. Data Cleaning: Handling missing values, Detecting and handling outliers.
4. Data Preprocessing: Transform features and create new variables to facilitate analysis, including categorizing Schirmer's test results.
5. Exploratory Data Analysis (EDA): Analysing the distribution, relationships, and patterns in the dataset using visualizations such as histograms, pair plots, correlation matrices, and feature importance plots. Gaining insights into the impact of digital screen usage on tear production and identify significant features for predicting dry eye syndrome severity.
6. Model Implementation: Developing machine learning models.
7. Building Web Application: Developing a web application using HTML, CSS, and Flask framework for interaction with the trained models.
8. Handling result: Displaying prediction results to users through the web application, providing insights into their potential dry eye syndrome severity based on input parameters and Schirmer's test results. Evaluating the model's performance and usability in real-time predictions, ensuring its effectiveness in providing personalized assessments.

5. METHODOLOGY

5.1 Data Collection

- **Description of data**

The dataset has 28 features and 300 columns observation.

- **Brief description of the dataset**

S. No	Feature Name	Description
1	Name	The name of the individual.
2	Age	The age of the individual.
3	Sex	Specifies the gender of the individual.
4	wearables	Whether the individual uses specs.
5	Duration	duration of usage of specs.
6	onlineplatforms	Using online platform or not.
7	Nature	Nature of computer usage in a day.
8	screenillumination	Level of screen illumination.
9	workingyears	Number of years working with computer.
10	hoursspentdailycurricular	Total hours spent for online classes.
11	hoursspentdailynoncurricular	Total hours spent for other reasons.
12	Gadgetsused	Gadgets used most of time.
13	levelofgadgetwithrespecttoeyes	Use of gadgets with respect to eyes.
14	Distancekeptbetweeneyesandgadget	Distance kept between eyes and gadget.
15	Avgnighttimeusageperday	Usage of gadget at night time.
16	Blinkingduringscreenusage	Whether blink during digital screen usage or not.
17	Difficultyinfocusingafterusingscreens	Difficulty in focusing from one distance to another after using screen.
18	frequencyofcomplaints	Frequency of complaints related to eye health
19	Severityofcomplaints	Severity of complaints related to eye health.
20	RVIS	Repetitive Visual Stimulus for eyes.
21	Ocularsymptomsobservedlately	Ocular symptoms observed lately.
22	Symptomsobservingatleasthalfof-thetimes	Symptoms observing at least half of the times.
23	Complaintsfrequency	Complaints frequency.
24	frequencyofdryeyes	Frequency of dry eyes.
25	Schimers1Lefteye	Schirmer's test 1 result for the left eye
26	Schimers1righteye	Schirmer's test 1 result for the right eye
27	Schimers2Lefteye	Schirmer's test 2 result for the left eye
28	Schimers2righteye	Schirmer's test 2 result for the right eye

- **Datatype of each feature**

The features in the dataset have 2 types of datatypes which are numerical and categorical.

S. No	Feature Name	Data Type
1	Name	Categorical variable
2	Age	Numerical variable
3	Sex	Numerical variable
4	wearables	Numerical variable
5	Duration	Numerical variable
6	onlineplatforms	Numerical variable
7	Nature	Numerical variable
8	screenillumination	Numerical variable
9	workingyears	Numerical variable
10	hoursspentdailycurricular	Numerical variable
11	hoursspentdailynoncurricular	Numerical variable
12	Gadgetsused	Numerical variable
13	levelofgadgetwithrespecttoeyes	Numerical variable
14	Distancekeptbetweeneyesandgadget	Numerical variable
15	Avgnighttimeusageperday	Numerical variable
16	Blinkingduringscreenusage	Numerical variable
17	Difficultyinfocusingafterusingscreens	Numerical variable
18	frequencyofcomplaints	Numerical variable
19	Severityofcomplaints	Numerical variable
20	RVIS	Numerical variable
21	Ocularsymptomsobservedlately	Numerical variable
22	Symptomsobservingatleasthalfofthetimes	Numerical variable
23	Complaintsfrequency	Numerical variable
24	frequencyofdryeyes	Numerical variable
25	Schimers1Lefteye	Numerical variable
26	Schimers1righteye	Numerical variable
27	Schimers2Lefteye	Numerical variable
28	Schimers2righteye	Numerical variable

5.2 Importing dataset and required packages

Pandas, NumPy, Matplotlib, Seaborn, and Scikit-learn are the libraries which are used for this dataset. The dataset is loaded using Pandas library, NumPy library is used to handle numerical computing, Seaborn and Matplotlib is used to visualize the data distributions, trends, and relationships. Finally, Scikit-learn offers a rich collection of machine learning algorithms and utilities, enabling to build predictive models, perform model selection, and evaluate model performance.

More information about the features:

#	Column	Non-Null Count
0	Name	300 non-null
1	Age	300 non-null
2	Sex	300 non-null
3	wearables	300 non-null
4	Duration	190 non-null
5	onlineplatforms	295 non-null
6	Nature	281 non-null
7	screenillumination	298 non-null
8	workingyears	294 non-null
9	hoursspentdailycurricular	295 non-null
10	hoursspentdailynoncurricular	298 non-null
11	Gadgetsused	300 non-null
12	levelofgadgetwithrespecttoeyes	300 non-null
13	Distancekeptbetweeneyesandgadget	299 non-null
14	Avgnighttimeusageperday	300 non-null
15	Blinkingduringscreenusage	300 non-null
16	Difficultyinfocusingafterusingscreens	300 non-null
17	frequencyofcomplaints	300 non-null
18	Severityofcomplaints	300 non-null
19	RVIS	300 non-null
20	Ocularsymptomsobservedlately	298 non-null
21	Symptomsobservingatleasthalfofthetimes	293 non-null
22	Complaintsfrequency	300 non-null
23	frequencyofdryeyes	300 non-null
24	Schimers1Lefteye	300 non-null
25	Schimers1righteye	300 non-null
26	Schimers2Lefteye	300 non-null
27	Schimers2righteye	300 non-null

Fig 5.2 Columns with Null values

The above figure specifies that there are columns with null values.

Duration, Online platforms, Nature, Screen Illumination, working years, Hours spent daily curricular, Hours spent daily Non curricular, Distance kept between eyes and gadgets, Ocular Symptoms observed lately and Symptoms observing at least half of the times features have null values. So, from this we can specify that there are Null values to handle the data.

5.3 Data Cleaning

- Handling Missing Values

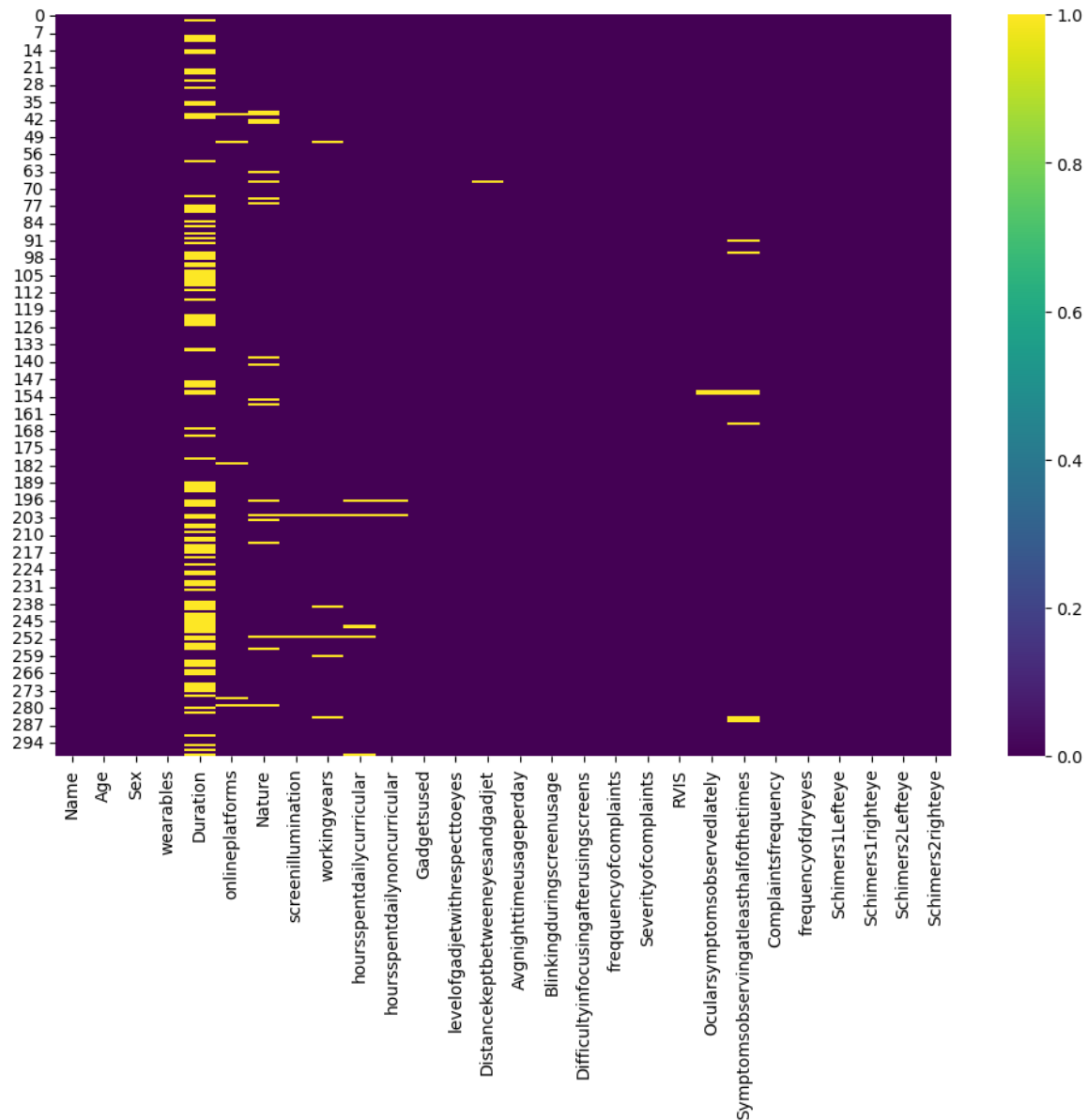


Fig 5.3.1 Visualization of Null values

The above figure shows that there are some missing values present in the dataset. The missing values are handled through technique called Mean Imputation. In this technique the missing value of any feature is replaced with the mean value of that feature. This approach is straightforward and easy to implement, making it especially easy when dealing with numerical features. As the most of the features which are having missing values are of numerical datatypes the mean imputation technique is efficient approach rather than dropping columns.

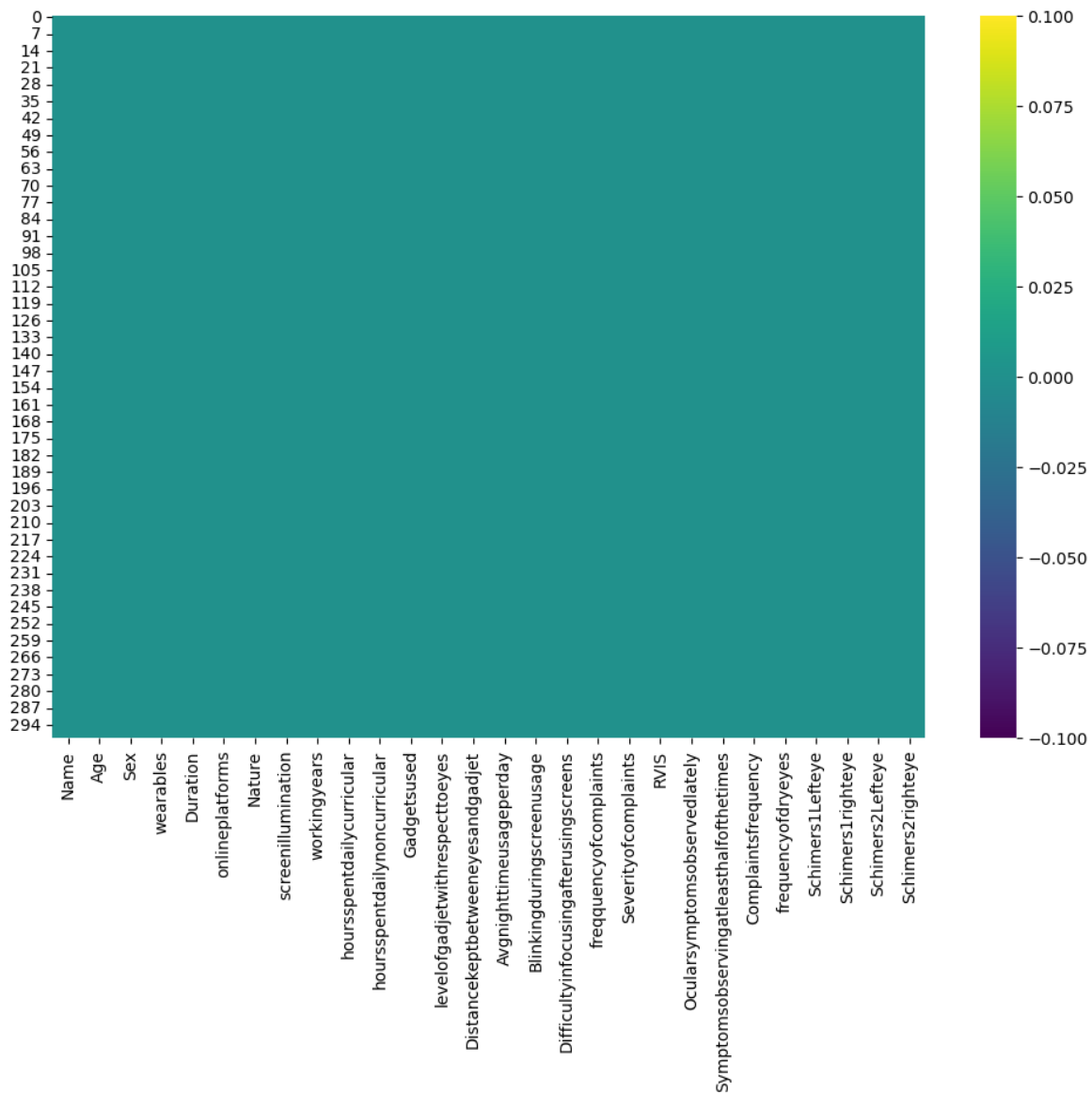


Fig 5.3.2 Visualization of Null values

The above figure shows that there are no null values or missing values in any of the feature.

5.4 Outlier detection

In the dataset there are some features which have outliers. These outliers are identified by a commonly used technique called Interquartile Range (IQR) Method. This IQR method identifies outliers based on the interquartile range, which is the difference between the third quartile (Q3) and the first quartile (Q1). Outliers are defined as data points that fall below

$Q1 - 1.5 * IQR$ or above $Q3 + 1.5 * IQR$.

The outliers are represented visually using box plots which are shown below:

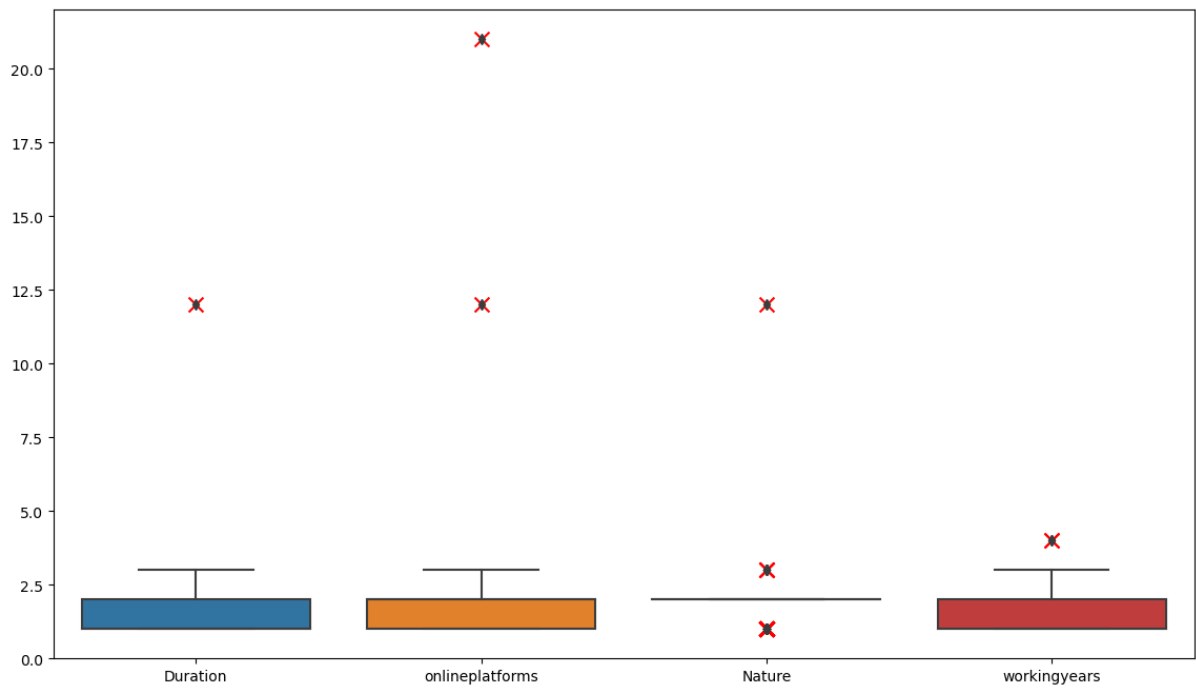


Fig 5.4.1 Visualizing outliers using box plots for 1st four columns

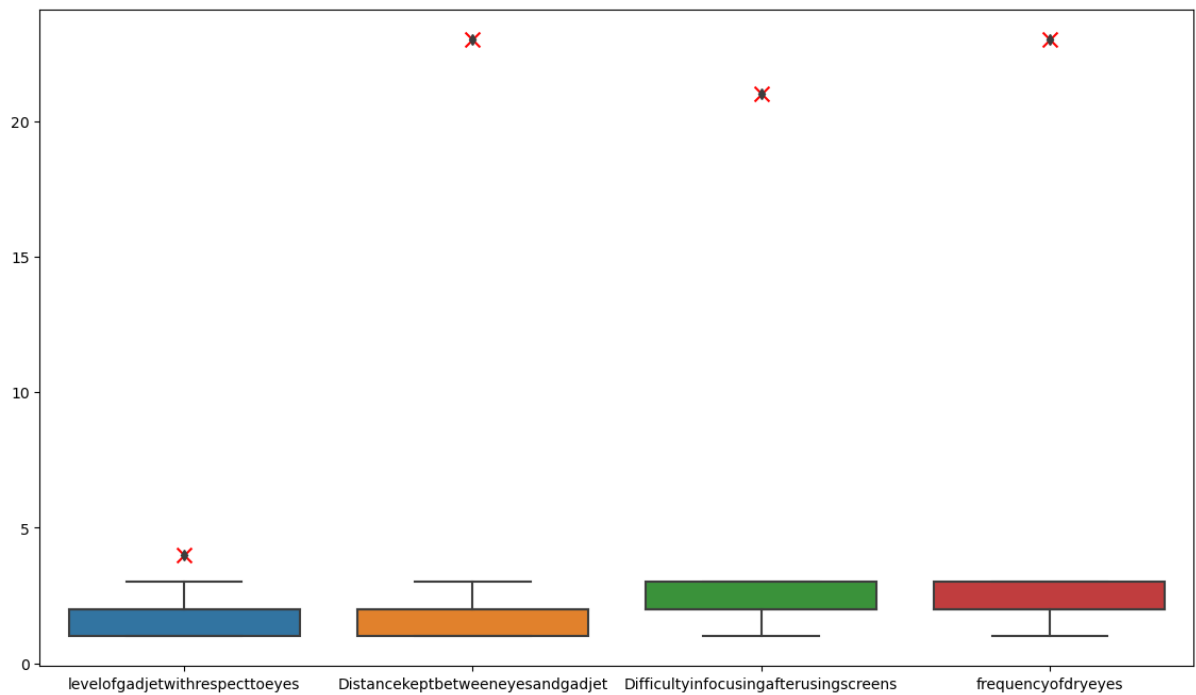


Fig 5.4.2 Visualizing outliers using box plots for last four columns

The cross marks represents that there is a value in that feature which is an outlier or anomalies due to various reasons such as measurement errors, data entry mistakes, extreme values or rare occurrences or others such that indicate data quality issues or inaccuracies that need to be addressed.

5.5 Outlier Handling

The outliers are handled using a technique called Winsorizing. In this method the anomalies or outliers are replaced with the mode (most frequently occurring value) of that feature. This process is commonly referred to as “capping with mode” or “Winsorizing with mode.”

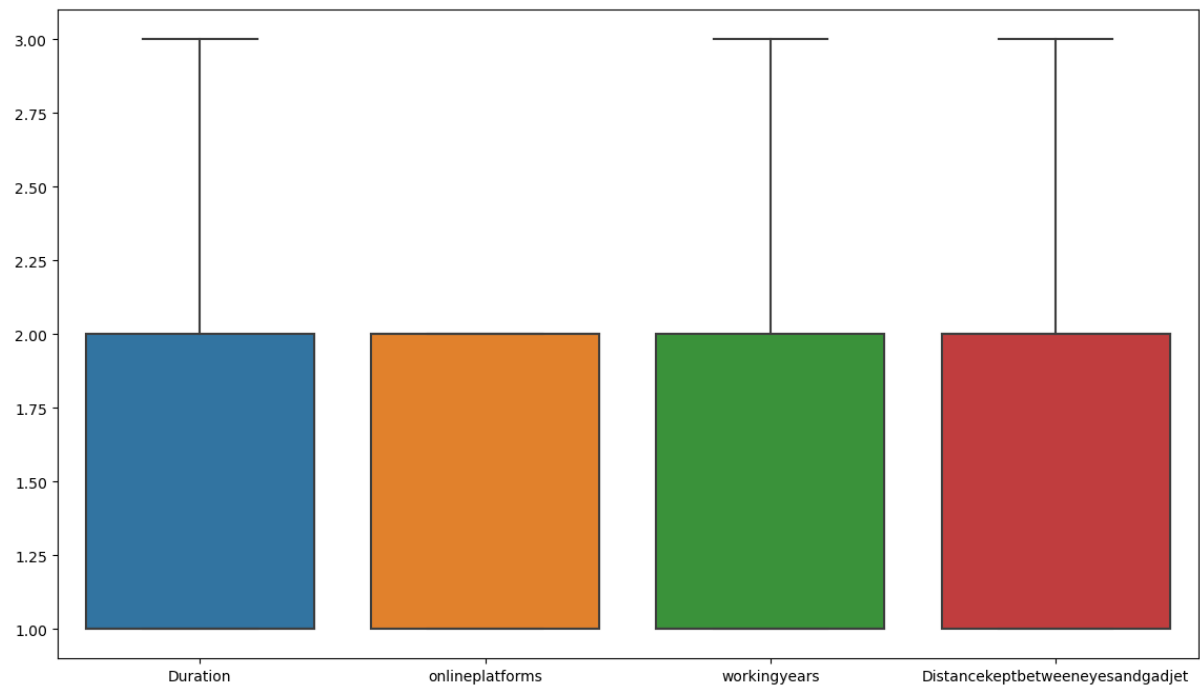


Fig 5.5.1 Visualization of outlier handling for 1st four columns

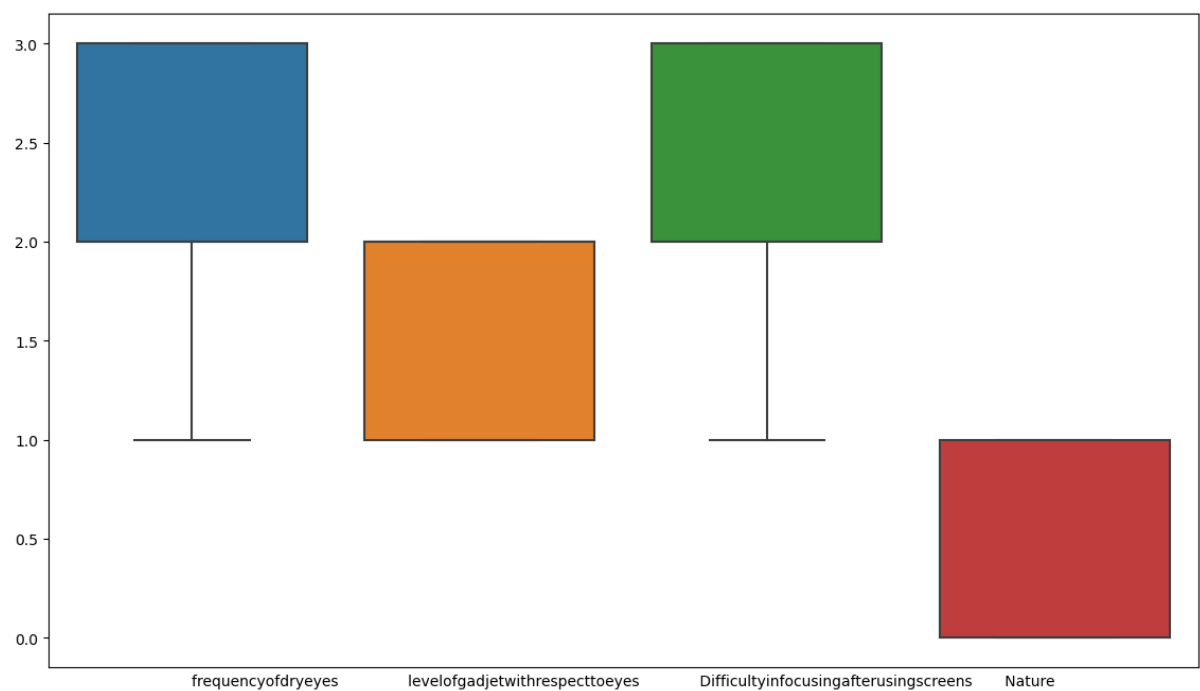


Fig 5.5.2 Visualization of outlier handling for last four columns

5.6 Handling Numeric Concatenation

Ocularsymptomsobservedlately	Symptomsobservingatleasthalfofthetimes
13569	1345
6	3
135679	12345
135	23
7	23
1357	1
12359	1235
13579	4
9	5

Fig 5.6 Handling Numeric Concatenated values of columns

The dataset contains the concatenated values for features Ocularsymptomsobservedlately and Symptomsobservingatleasthalfofthetimes. The concatenated value is composed of numerical digits, which may represent a combination of individual numeric values. So, for this type of features there is a technique to handle such values called as Aggregate analysis. This method works by taking the mean of concatenated values. This technique is useful for this situation because the concatenated values represent numerical.

5.7 DATA PREPROCESSING

- **Feature Transformation**

The dataset examines the impact of digital screens on tear production in the eyes, measured through the Schirmer's test. This test yields numerical values indicating the level of tear production. To facilitate analysis, we've created new features as follows:

1. **Result Feature:** This feature indicates whether a person has dry eye syndrome. It categorizes individuals into "Normal" or "Abnormal" groups. "Normal" signifies an average tear production greater than 10mm, while "Abnormal" indicates an average tear production of less than 10mm.

These transformations aim to prepare the data for analysis, particularly for logistic regression modelling.

2. **Schirmer's Test Categorization:** We've transformed the numerical values in columns such as Schirmer1lefteye, Schirmer1righteye, Schirmer2lefteye, and Schirmer2righteye into categorical values:

Schirmer1lefteye and Schirmer1righteye are categorized as:

Severe Dry Eyes: Below 5 mm

Moderately Dry Eyes: 5 to 10 mm

Mildly Dry Eyes: 10 to 15 mm

Normal: 15 mm and above

Schirmer2lefteye and Schirmer2righteye are categorized as:

Severe Dry Eyes: Below 3 mm

Moderately Dry Eyes: 3 to 6 mm

Mildly Dry Eyes: 6 to 10 mm

Normal: 10 mm and above

5.8 EXPLORATORY DATA ANALYSIS

Exploratory Data Analysis (EDA) is an essential step in the machine learning process, where the main goal is to analyse and understand the characteristics of the data before applying any modelling techniques. EDA is the process of summarizing the main characteristics of the data, such as the distribution, the relationship between variables, and identifying any patterns or anomalies that may exist.

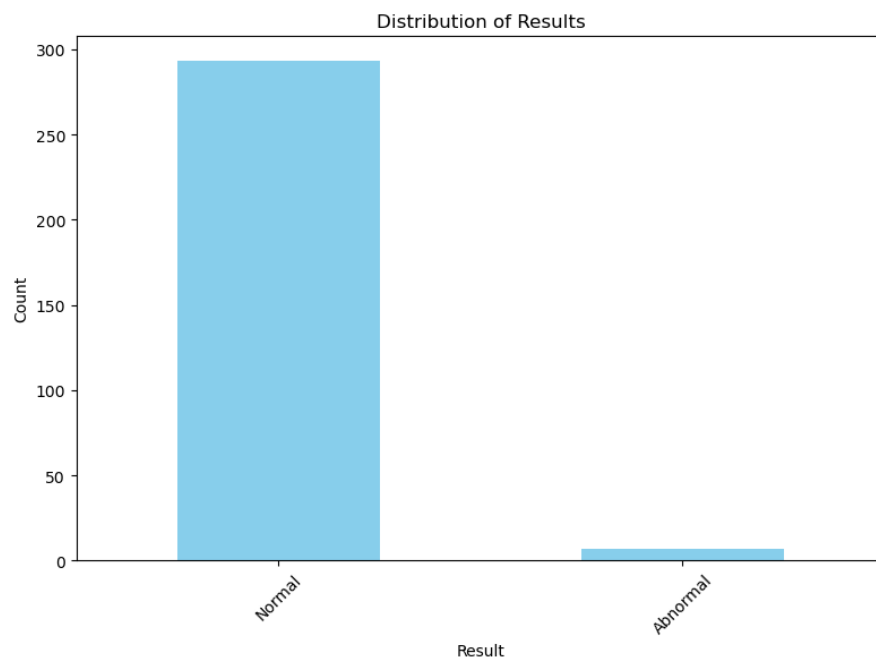


Fig 5.8.1 Distribution of Results

The above figure shows that there is maximum number of persons having normal as result and very few having abnormal.

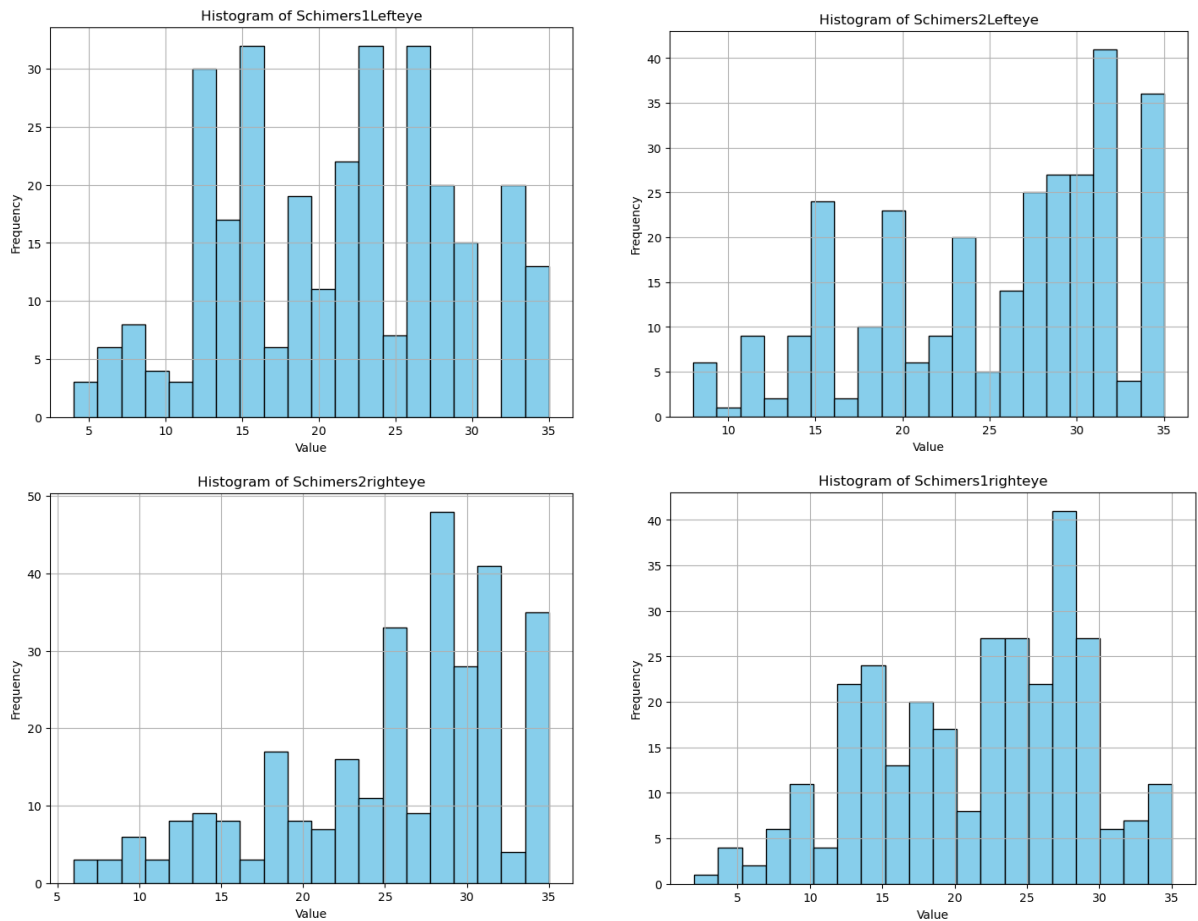


Fig 5.8.2 Frequency of values for target columns

The histogram shows the relation with each of frequency for Schirmer1lefteye, Schirmer1righteye, Schirmer2lefteye, and Schirmer2righteye.

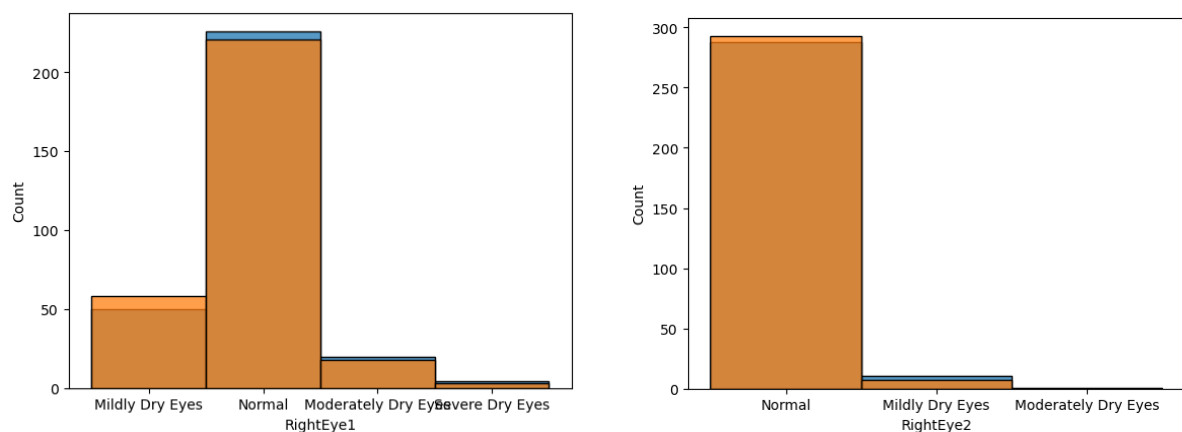


Fig 5.8.3 Frequency of Categorical values of target variables

The above figure shows how categorial value of Schimers test 1 of both left and right eyes correlate and Schimers test2 of both left and right eyes.

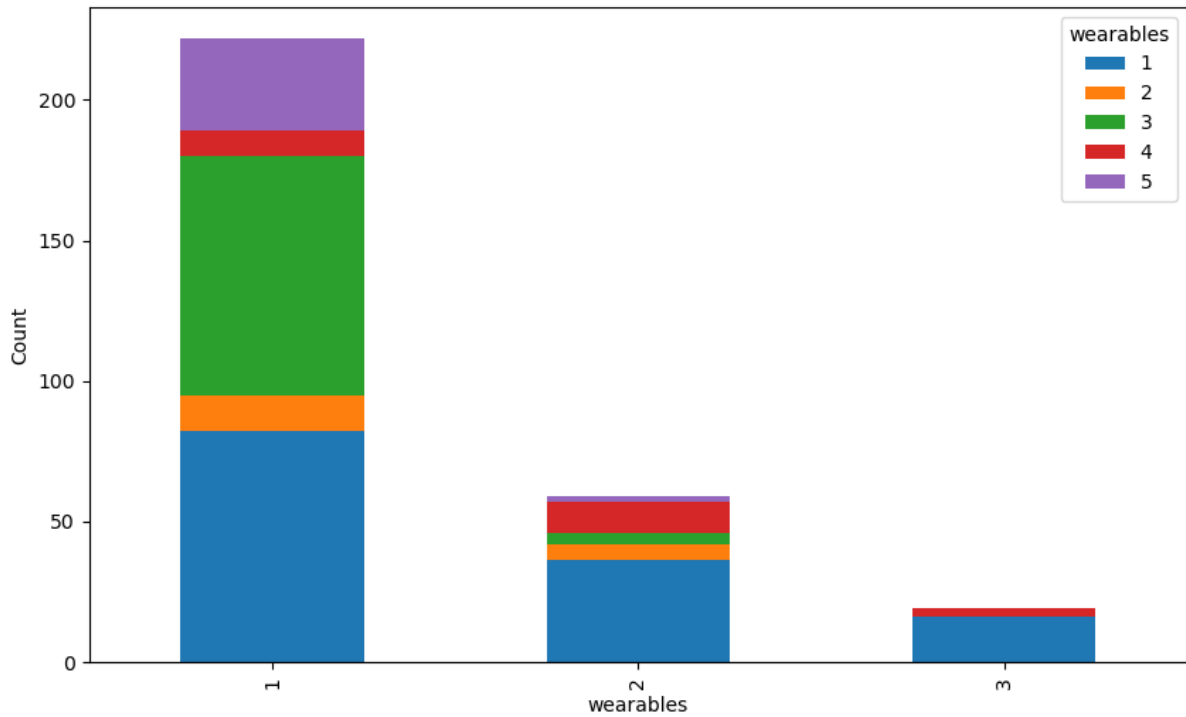


Fig 5.8.4 Relation between wearables

The above pair plot shows pair relation between wearables specifying does wear specs or contact lenses and duration of specs usage.

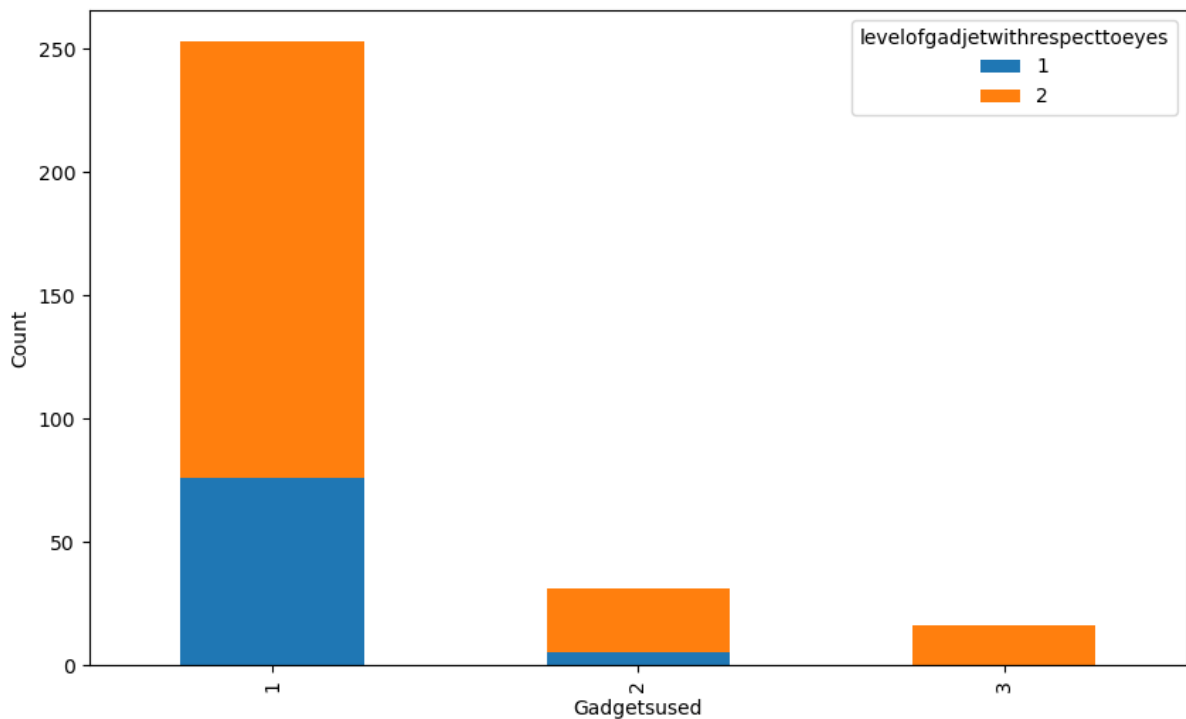


Fig 5.8.5 Representation of level of gadgets with respect to eyes

The above figure shows that the gadgets used at the level of eyes and at the above level of eyes. Mobile gadget is most used and the used at the level of eyes.

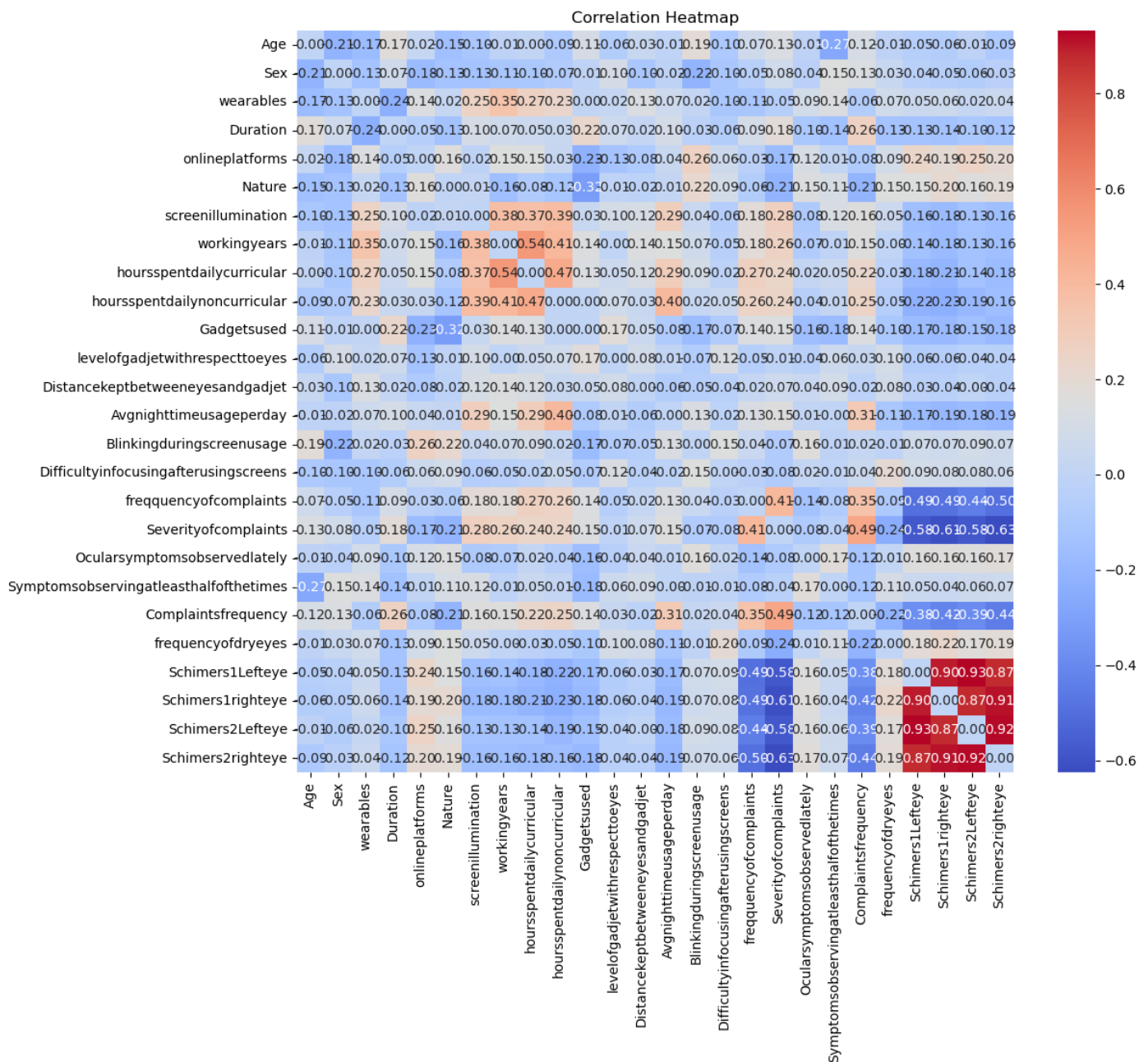


Fig 5.8.6 Correlation Heatmap

A correlation matrix or heatmap is a statistical table or picture that shows the relationship between the variables in a data set. Each cell in the matrix contains a correlation coefficient, which indicates the strength of the relationship between two variables. The values of the correlation coefficients range from 1, which indicates a strong relationship, to 0, which indicates a neutral relationship, and to -1, which indicates a not strong relationship.

So, from the figure the features with red in colour is strong in relationship we can say that it ranges between 0.6 to 0.8 and the features range between is in neutral relationship they may range between -0.2 to 0.4 is neutral relationship and is in light blue colour and less than are in dark blue.

5.9 FEATURE IMPORTANCE

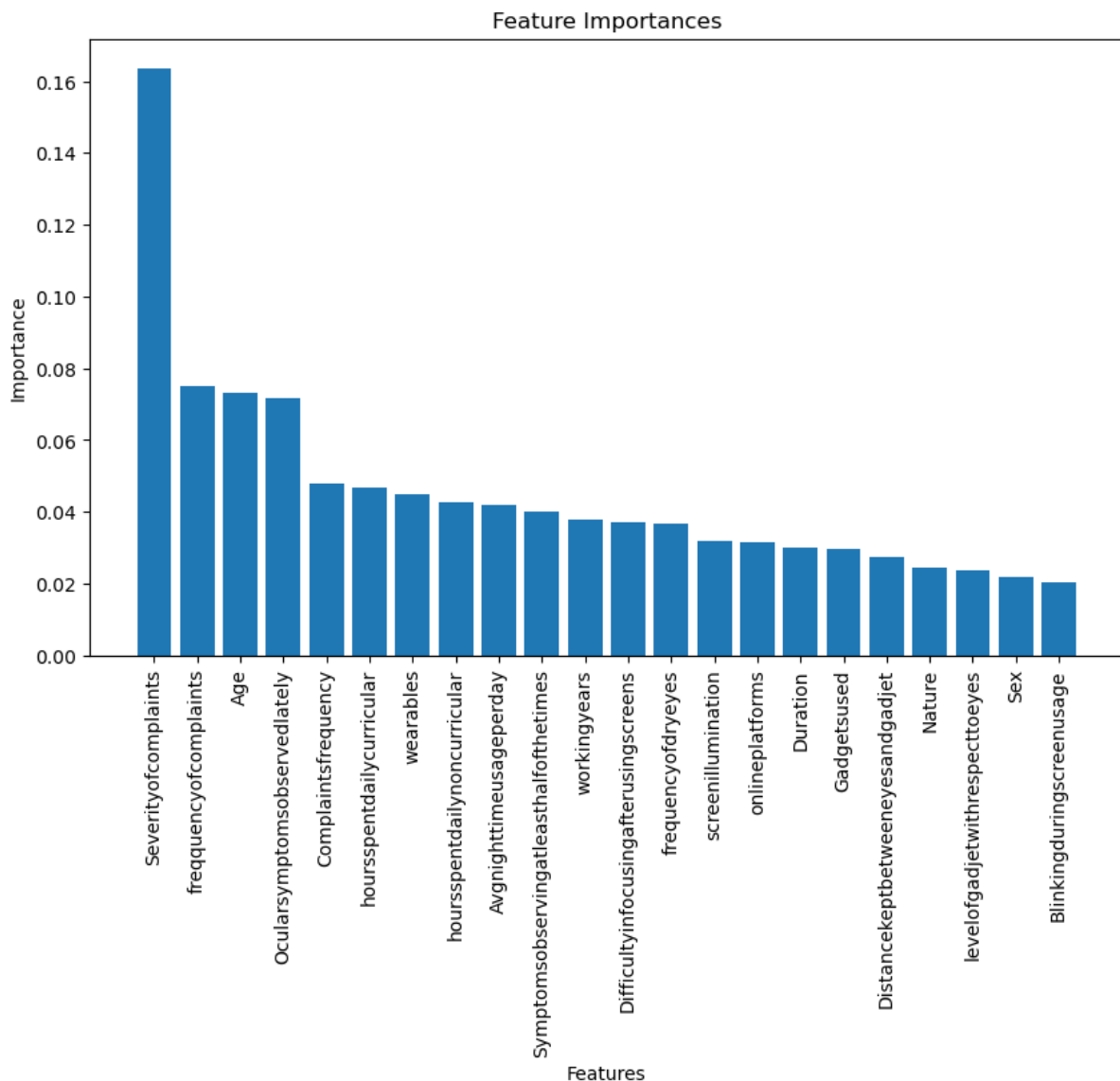


Fig 5.9 Feature Importance

There are 28 features in the dataset and for model training there can be some features dropped which are not related or not necessary. So, for knowing which feature is important we used random forest to find important features.

From the above figure we can say that the features named Gadgetsused, Duration, Nature, DistanceKeptbetweeneyesandgadget, sex, Blinkingduringscreenusage, onlineplatforms and levelofgadgetwithrespecttoeyes which are below the required threshold so dropping these features may give model develop with greater accuracy.

Rvis is a feature which is not related to the dataset and Name feature also not contributing much or may not give great importance. So, dropping them may helps to develop model with good accuracy.

6. ALGORITHMS

The algorithms used for building model are as follows:

1. **Logistic Regression:** Logistic Regression is used for binary classification problems. It models the probability that a given input belongs to a particular class. Despite its name, it's a linear model for classification rather than regression. It uses the logistic function to squash the output between 0 and 1.
2. **Decision Tree:** Decision Trees recursively split the dataset into subsets based on the most significant attribute at each step. The process continues until a stopping criterion is met. It's interpretable and can handle numerical and categorical data.
3. **Random Forest Regressor:** Random Forest is an ensemble technique that creates multiple decision trees and merges their predictions. Each tree is trained on a random subset of the training data and a random subset of features. This randomness helps to reduce overfitting and improves the model's generalization.
4. **Decision Tree Regressor:** Like Decision Trees for classification, it works for regression problems. Instead of predicting class labels, it predicts continuous values. It splits the dataset into subsets based on the value of features to minimize the variance within each subset.
5. **Support Vector Regressor (SVR):** SVR is a variation of Support Vector Machines (SVM) used for regression problems. It tries to fit as many instances as possible within a margin of tolerance while limiting margin violations. SVR uses a kernel function to transform the input features into higher-dimensional space, where a linear model is constructed.
6. **Gradient Boosting Regressor:** Gradient Boosting builds an ensemble of weak learners, typically decision trees, in a sequential manner. Each tree corrects the errors of the previous ones, focusing on the residuals. It minimizes a loss function, often the mean squared error, by adding trees that improve the model's predictions.
7. **Decision Tree Classifier:** Decision Trees for regression, works for classification problems. It recursively splits the dataset based on the feature that provides the most information gain or Gini impurity reduction. It creates a tree-like model where leaves represent class labels

7. MODEL IMPLEMENTATION

The implemented models are as follows:

7.1 Logistic Regression

Classification Report:				
	precision	recall	f1-score	support
Abnormal	0.00	0.00	0.00	1
Normal	0.98	0.98	0.98	59
accuracy			0.97	60
macro avg	0.49	0.49	0.49	60
weighted avg	0.97	0.97	0.97	60

Fig 7.1.1 Classification report of Logistic regression

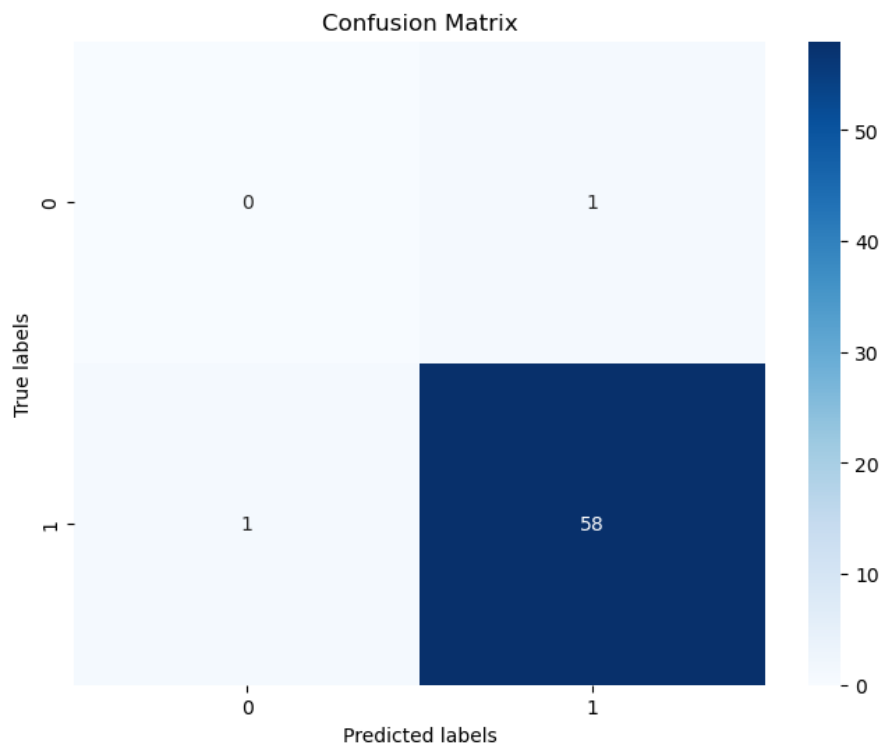


Fig 7.1.2 Confusion Matrix of Logistic Regression

Logistic regression accuracy: 0.966

The model is trained using result feature as target variable which specifies only whether eye dryness is normal or abnormal.

The confusion matrix and classification report indicate that the model's performance skewed towards the majority class ("Normal"), leading to imbalanced evaluation metrics.

7.2 Decision tree

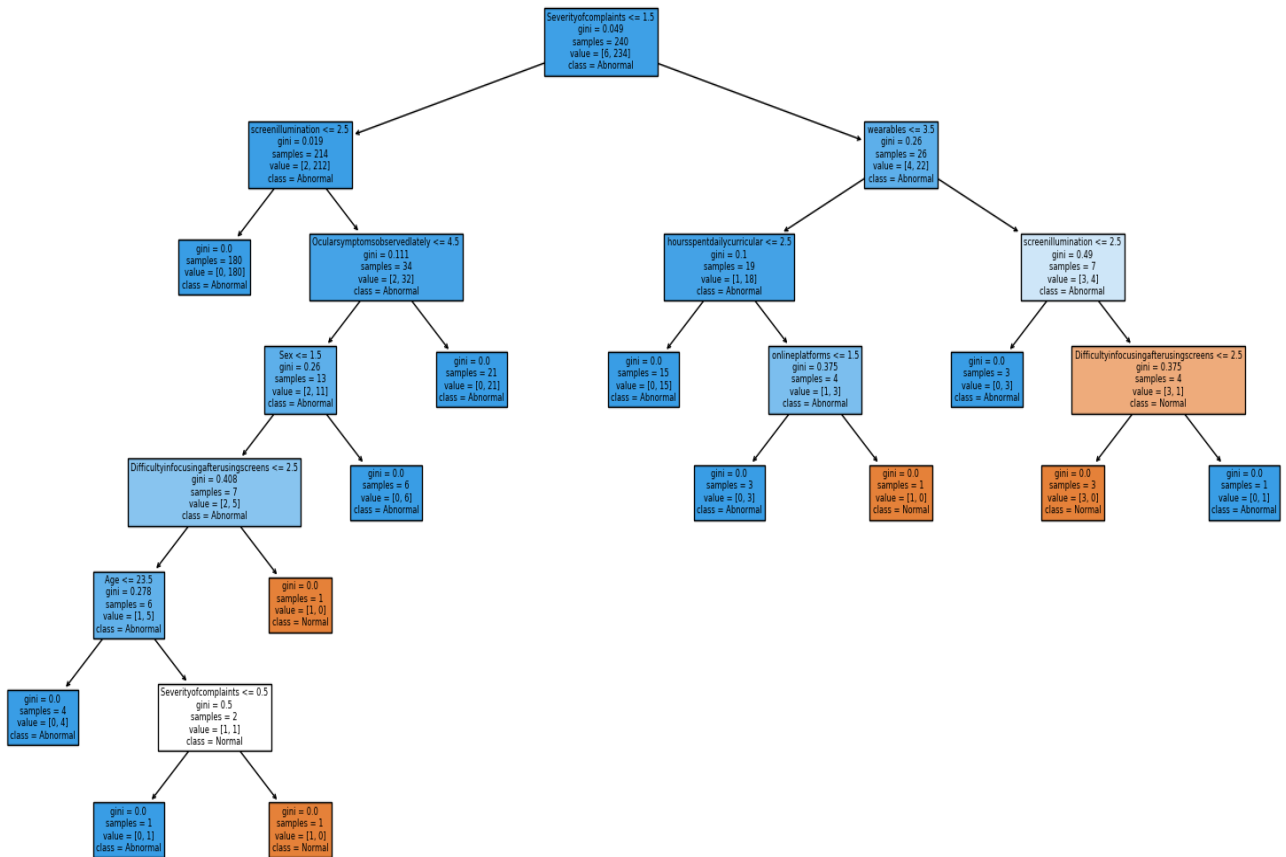


Fig 7.2.1 Decision tree

Classification Report:				
	precision	recall	f1-score	support
Abnormal	0.33	1.00	0.50	1
Normal	1.00	0.97	0.98	59
accuracy			0.97	60
macro avg	0.67	0.98	0.74	60
weighted avg	0.99	0.97	0.97	60

Fig 7.2.2 Classification report of Decision Tree

Decision tree accuracy: 0.866

The model is trained using result feature as target variable which specifies only whether eye dryness is normal or abnormal.

The model performs well, especially for the "Normal" class, but its performance on the "Abnormal" class is relatively weaker, likely due to the class imbalance with only one instance.

7.3 Random Forest regressor

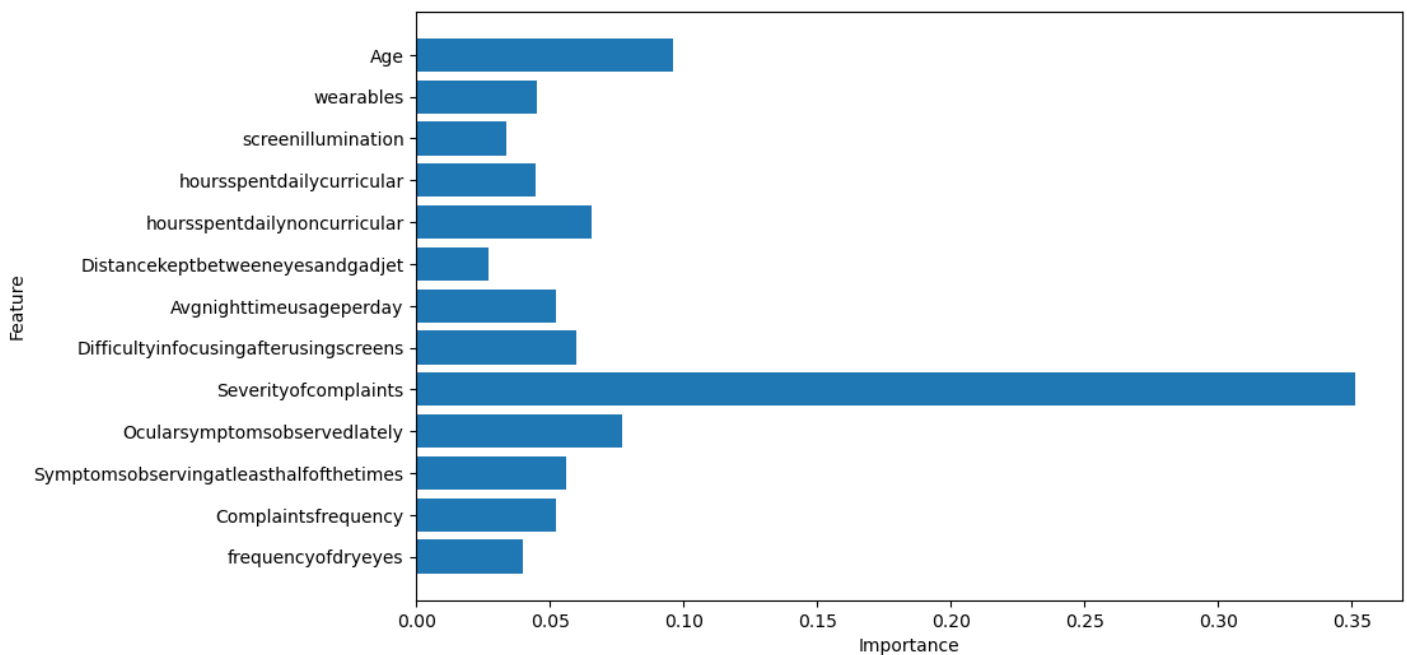


Fig 7.3.1 Random Forest Regressor

```

Mean Absolute Error for Schimers1Lefteye: 7.028000000000005
Mean Squared Error for Schimers1Lefteye: 74.94372
Mean Absolute Error for Schimers1righteye: 10.164000000000001
Mean Squared Error for Schimers1righteye: 126.81460000000001
Mean Absolute Error for Schimers2Lefteye: 10.01
Mean Squared Error for Schimers2Lefteye: 133.19366
Mean Absolute Error for Schimers2righteye: 10.86
Mean Squared Error for Schimers2righteye: 151.12215999999998

```

Fig 7.3.2 Metrics of Random Forest Regressor

The MAE Mean Absolute Error values provided range from approximately 7 to 10. Lower MAE values suggest better predictive accuracy.

```

R-squared for Schimers1Lefteye: -1.8222451456310673
R-squared for Schimers1righteye: -2.704556561085972
R-squared for Schimers2Lefteye: -2.81537020316027
R-squared for Schimers2righteye: -2.2565708401976936

```

Fig 7.3.3 R-squared error for of Random Forest Regressor

7.4 Decision Tree Regressor

```
R-squared for Schimers1Lefteye: -3.5206310679611637
R-squared for Schimers1righteye: -1.794117647058823
R-squared for Schimers2Lefteye: -4.699774266365688
R-squared for Schimers2righteye: -1.2199341021416807
```

Fig 7.4 Decision Tree Regressor

R-squared values represent the coefficient of determination, which is a measure of how well the independent features explain the variance in the dependent variable i.e. target. It ranges from 0 to 1. Here the value is in negative which specifies that model is performing purely.

7.5 Support Vector Regressor

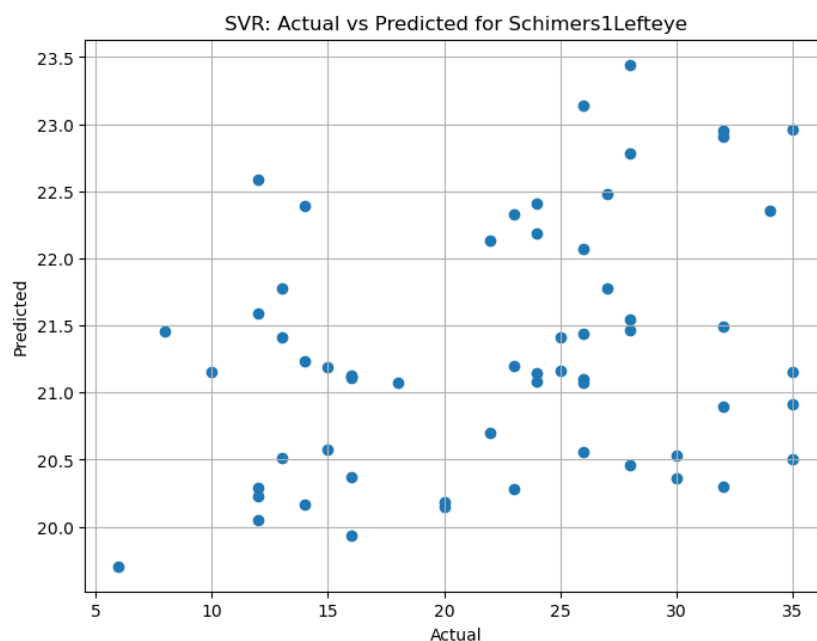
```
Evaluation Metrics for Schimers1Lefteye:
Mean Squared Error (MSE): 57.934750163754856
Mean Absolute Error (MAE): 6.582259398989942
R-squared (R2): 0.04496162452598418

Evaluation Metrics for Schimers1righteye:
Mean Squared Error (MSE): 47.19984708958982
Mean Absolute Error (MAE): 5.848843956497311
R-squared (R2): 0.07330143148711099

Evaluation Metrics for Schimers2Lefteye:
Mean Squared Error (MSE): 50.80892154614288
Mean Absolute Error (MAE): 5.99827193022794
R-squared (R2): 0.019952648116578042

Evaluation Metrics for Schimers2righteye:
Mean Squared Error (MSE): 49.11698375463722
Mean Absolute Error (MAE): 5.521293948640605
R-squared (R2): 0.04542261712584561
```

Fig 7.5.1 Metrics of Support Vector Regressor



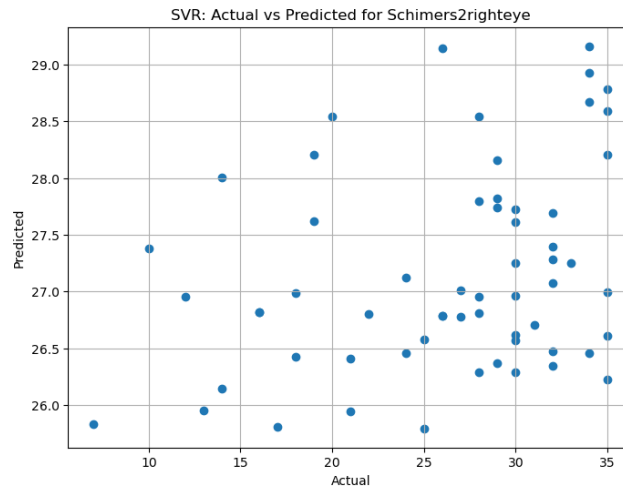


Fig 7.5.2 SVR Actual vs Predicted for Schimer's Test1

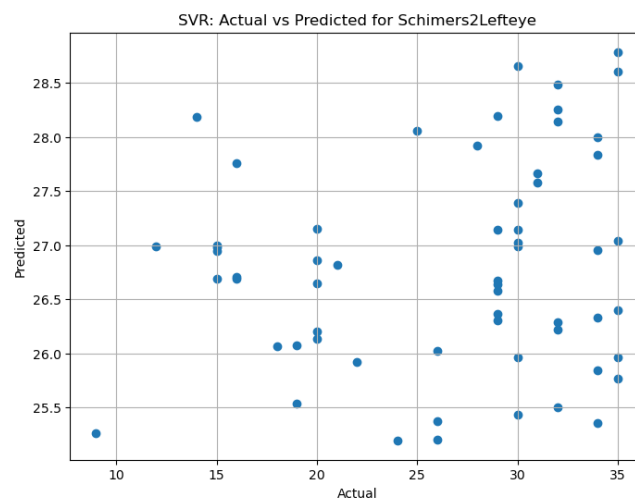
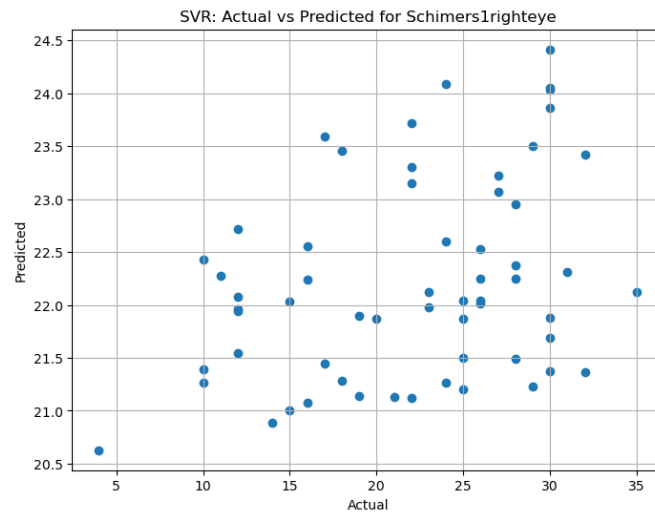


Fig 5.7.3 SVR Actual vs Predicted for Schimer's Test2

R squared values ranges in between 0.04 to 0.07. Here R squared indicates that the model does not explain any of the variability of the response data around its mean.

7.6 Gradient Boosting Regressor

Evaluation Metrics for Schimers1Lefteye:
Mean Squared Error (MSE): 51.65949054634538
Mean Absolute Error (MAE): 5.571058447897959
R-squared (R2): 0.1484075483238545

Evaluation Metrics for Schimers1righteye:
Mean Squared Error (MSE): 38.31810675130374
Mean Absolute Error (MAE): 4.914752171277754
R-squared (R2): 0.2476811501707381

Evaluation Metrics for Schimers2Lefteye:
Mean Squared Error (MSE): 42.51695749606904
Mean Absolute Error (MAE): 5.265846933783729
R-squared (R2): 0.17989537395867616

Evaluation Metrics for Schimers2righteye:
Mean Squared Error (MSE): 41.3289338244201
Mean Absolute Error (MAE): 5.243008438956157
R-squared (R2): 0.19678159220497016

Fig 7.6.1 Metrics of Gradient Boosting Regressor

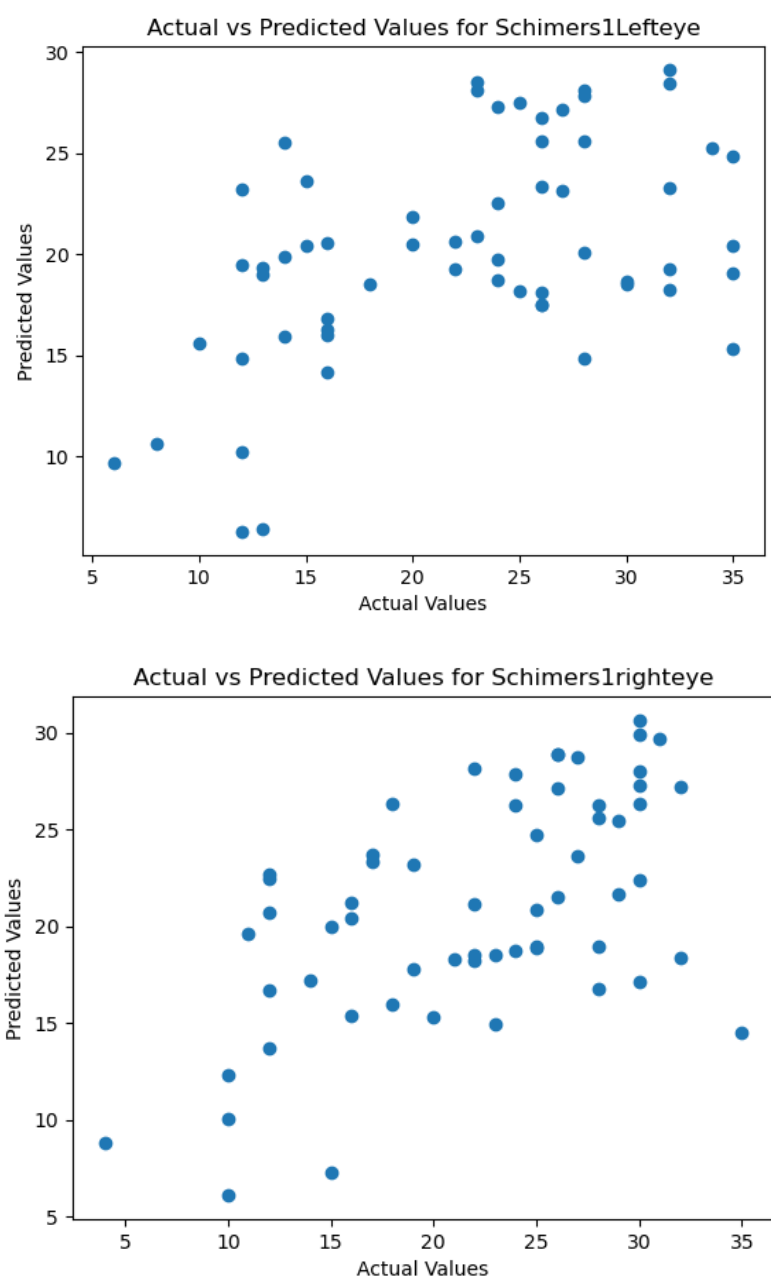


Fig 7.6.2 Actual vs Predicted values for Schimer's Test 1

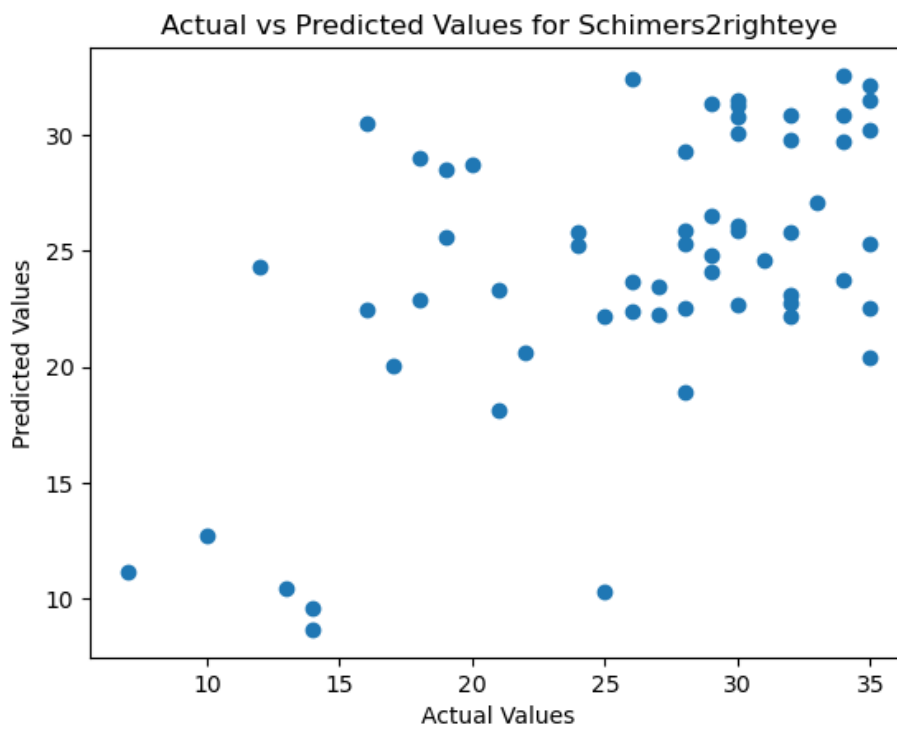
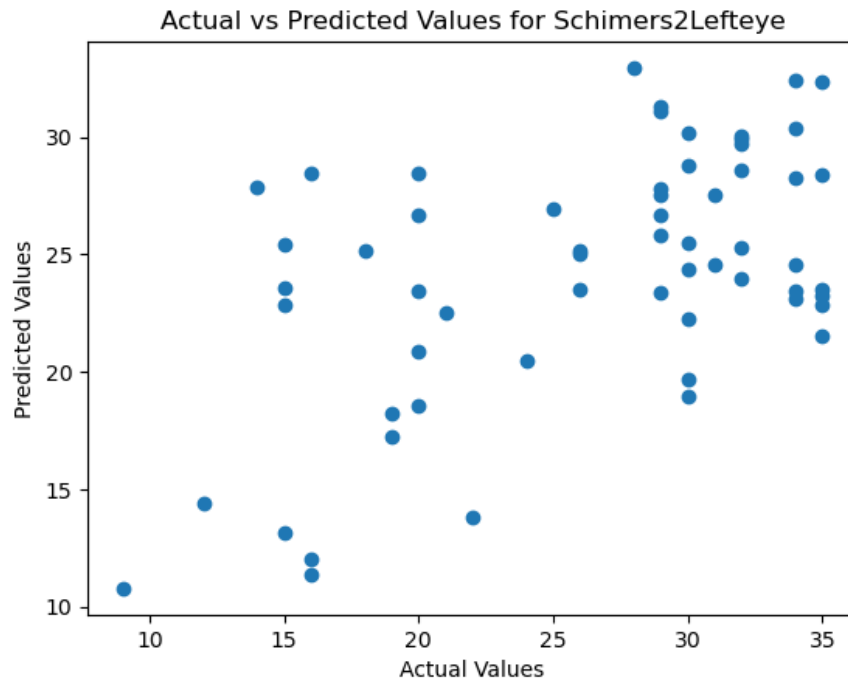


Fig 7.6.3 Actual vs Predicted values for Schimer's Test 2

These results suggest that the models are capturing some of the variance in the target variables, but there is still room for improvement. The R-squared values are relatively low, indicating that the models explain only a small portion of the variability in the target variables. Additionally, the MSE and MAE values provide insights into the magnitude of errors in predictions.

7.7 Decision tree Classifier

In this, the model is trained with categorical variable which is transformed at feature transformation phase.

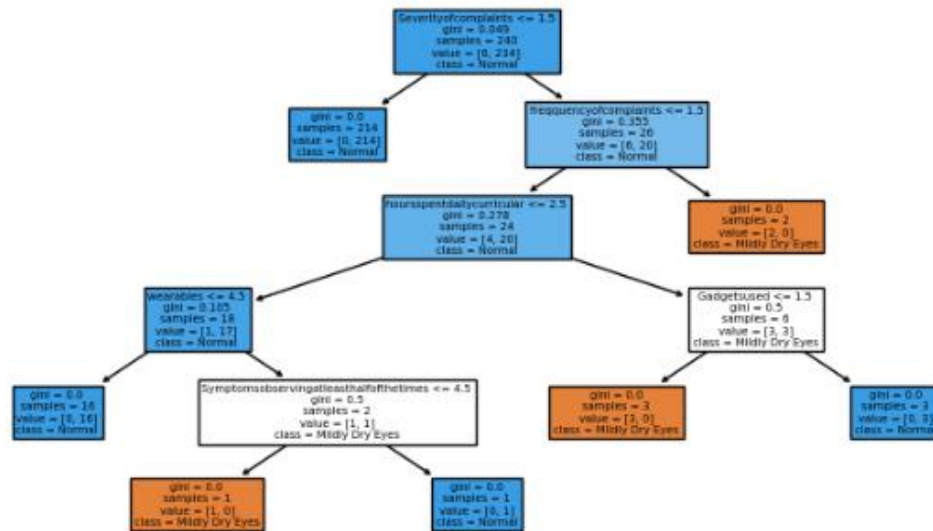


Fig 7.7.1 Decision Tree Classifier

```
Accuracy for Schimers1Lefteye_1: 0.7
Accuracy for Schimers1righteye_1: 0.7666666666666667
Accuracy for Schimers2Lefteye_1: 0.9666666666666667
Accuracy for Schimers2righteye_1: 0.9333333333333333
```

Fig 7.7.2 Accuracy of Decision Tree Classifier

The models, trained using Decision Tree classifiers, aim to predict the severity of eye-related symptoms in individuals. Each classifier was trained separately on its corresponding target variable, learning patterns in the features to make predictions. After training, the models were evaluated using accuracy metrics, revealing varying degrees of predictive performance across different symptom severities.

Overall Accuracy: 0.841

Final Observations:

We have developed 7 machine learning models which includes logistic regression, decision tree, random forest regressor, support vector regressor, Gradient Boosting Regressor and Decision tree Classifier. Based on models developed we have found that Decision tree Classifier have good accuracy and Gradient Boosting Regressor is next best where R-squared values are relatively low.

So, the model is trained using Decision tree Classifier and tested with the splitted data and also validated by passing random inputs without knowing values of target variables.

As the model is trained, the trained model is saved into a single pickle file, enabling their reuse for future predictions without the need for retraining.

The process of obtaining input from the user after the model has been developed is commonly referred to as inference or prediction. In this context, it involves taking input features from the user, passing them through the trained model, and obtaining the model's predictions as output. This step allows the model to make predictions on new, unseen data based on the learned patterns from the training data.

- **Building Web application**

The Web application is developed using HTML (Hyper Text Markup Language) and CSS (Cascading Style sheet). The web application was developed to predict Schirmer's based on digital screen usage. It allows users to input various parameters related to their digital screen usage habits, eyewear, computer usage, ocular symptoms and etc...

The web application includes taking inputs for prediction so it basically has inputs Parameters as:

The web application includes the following input parameters:

Name: User's name.

Age: User's age.

Sex: Gender selection (male/female).

Eyewear: Type of eye wear worn by the user (glasses, contact lenses, both, or none).

Duration of usage of spectacles: Duration of spectacle usage.

Online Platforms: Whether the user uses online platforms for webinars or classes.

Nature of Computer Usage: Frequency of computer usage (continuous or interrupted).

Normal Screen Illumination: Level of screen illumination.

Years Working with Computer: Number of years spent working with a computer daily.

Hours Spent for Online Classes: Duration of online classes for curricular purposes.

Hours Spent for Non-Curricular Purposes: Duration of non-curricular activities.

Most Used Gadget: Preferred gadget for digital screen usage.

Gadget Level with Respect to Eyes: Level of the gadget concerning the eyes.

Distance between Eyes & Gadget: Distance between eyes and gadget during usage.

Night Time Usage of Gadgets: Average nighttime usage of gadgets.

Remind to Blink: Whether the user reminds themselves to blink during screen usage.

Difficulty Focusing: Difficulty focusing after using the screen.

Complaints Frequency: Frequency of experiencing above complaints.

Ocular Severity: Severity of ocular complaints.

Ocular Symptoms: Ocular symptoms observed lately.

Symptoms Observation: Symptoms observed lately.

Complaint Frequency: Frequency of experiencing above complaints.

Dry Frequency: Frequency of dry eyes.

Predicting Schirmer's based on Digital Screen Usage

Name

Age

Sex

☐ Male ☐ Female

Do you wear Prescription glasses or contact lenses?

- ☐ Glasses only
☐ Contact lenses only
☐ None
☐ Both, at one or other time
☐ Not applicable

Fig 7.8.1 Web app Screen 1

Duration of usage of spectacles

☐ 0-5 years ☐ 5-10 years ☐ More than 10 years

Are you using any online platforms to attend any webinars or classes?

☐ Yes ☐ No

Nature of computer usage in a day

☐ Continuous ☐ Interrupted

What will be your Normal Screen Illumination while working in computer?

☐ 25% ☐ 25-30% ☐ 50%

How many Years you have been Working with Computer on daily basis?

☐ lesser than 5 years ☐ between 5-10 years ☐ greater than 10 years

How many hours Spent for Online Classes (Curricular purpose) in a day?

- ☐ 2-4 hours
☐ 4-6 hours
☐ 6-8 hours
☐ 8-10 hours

Fig 7.8.2 Web app Screen 2

How many hours Spent for reasons other than online classes (Non-Curricular purpose) in a day?

- ☐ Less than 2 hours
- ☐ 2-4 hours
- ☐ 4-6 hours
- ☐ 6-8 hours

What gadget do you use most of the times for above said purposes?

- ☐ Mobile ☐ Laptop ☐ Tablet

How do you use the gadget at least half of the times (level of gadget with respect to eyes)?

- ☐ Above the level of eyes (eyes have to look up)
- ☐ At the level of eyes

What distance do you keep between your eyes & gadget more than half of the times?

- ☐ Less than 25 cm
- ☐ 25-40 cm
- ☐ More than 40 cm

What is the average night time (after sunset) usage (in hours) of the gadgets per day?

- ☐ Less than 1 hour
- ☐ 1-2 hours
- ☐ 2-4 hours
- ☐ More than 4 hours

Fig 7.8.3 Web app Screen 3

Do you keep on reminding yourself to blink during digital screen usage?

- ☐ Yes ☐ No

After using screen, do you feel difficulty focusing from one distance to another more than half of the times?

- ☐ Yes ☐ No ☐ Maybe

How frequently do you experience the above complaints?

- ☐ Never
- ☐ less than 5 times a week
- ☐ more than 5 times a week

Severity of ocular complaints

- ☐ Never
- ☐ Disturbs work
- ☐ Severe enough to stop work and take rest

Fig 7.8.4 Web app Screen 4

Which out of the following ocular symptoms do you observe lately (after start of online classes) at least half of the times?

(MORE THAN ONE RESPONSES CAN BE MARKED)

- ☐ Heaviness/tired eyes
- ☐ Redness in eyes
- ☐ Watering from eyes
- ☐ Feeling of grittiness in eyes
- ☐ Itchy eyes (eye rubbing)
- ☐ Transient blurring of vision
- ☐ Sensitivity to bright light
- ☐ Contact lens discomfort
- ☐ Headache
- ☐ Other

Fig 7.8.5 Web app Screen 5

Are you observing any of these symptoms at least half of the times lately?

(MORE THAN ONE RESPONSES CAN BE MARKED)

- ☐ Difficulty concentrating
- ☐ Anxiety
- ☐ Sleep disturbances
- ☐ Dizziness
- ☐ Pain in neck

How frequently do you experience the above complaints?

- ☐ Never
- ☐ lesser than 5 times a week
- ☐ greater than 5 times a week

How often do your eyes feel dry?

- ☐ Often
- ☐ Sometimes
- ☐ Never

Fig 7.8.6 Web app Screen 6

After taking inputs through web application, we created a new Python file (app.py) to define Flask application and we defined routes for our application. Here one route is used to render the form for users to input their data and another route to handle form submission and generate the result.

Flask routes which are defined to handle user requests, with one route rendering a form for inputting data and another route processing form submission, generating predictions, and displaying the result. The Flask application integrates with the trained models through a function to make predictions based on user input. HTML templates are used to design the user interface for inputting data and displaying predictions. Overall, the implementation successfully combines machine learning models with a Flask web application to provide predictive functionality tailored to users' digital screen usage effects eyes.

```
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

Fig 7.8.7 Running Flask application on localhost

When running a Flask application, the framework automatically assigns a port number for the application to listen on. By default, Flask uses port 5000. This means that when the Flask application is started, it will be accessible through a web browser using the appropriate port number. Accessing the application in a web browser would then be done by navigating to “http://localhost:5000”. The port number is crucial as it enables communication between the Flask application and the web browser, allowing users to interact with the web application seamlessly.

8. RESULT



Fig 8.1 Web address of Flask Application

The URL "http://127.0.0.1:5000" you would be accessing a Flask web application running on your local machine through a web browser. Upon accessing this URL, web browser would display the home page in Flask application, allowing to interact with the web application's features and functionalities.

After passing input parameters by an user the model predicts using Decision tree classifier and gives result in Result.html page.

Result

Normal

Predictions:

Schimers Test 1

Schimers1Lefteye : Normal

Schimers1righteye : Normal

Schimers Value	Outcome
Greater than 15	Normal
10 - 15	Mildly Dry Eyes
5 - 10	Moderate Dry Eyes
Less than 5	Severe Dry Eyes

Schimers Test 2

Schimers2Lefteye : Normal

Schimers2righteye : Normal

Schimers Value	Outcome
Greater than 10	Normal
6 - 10	Mildly Dry Eyes
3 - 6	Moderate Dry Eyes
Less than 3	Severe Dry Eyes

Fig 8.2 Result 1

The prediction results indicate that the user is not affected by dry eyes, as both Test1 and Test2 evaluations for both left and right eyes suggest a normal condition. This implies that, based on the model's assessment, the user is unlikely to experience symptoms associated with dry eyes.

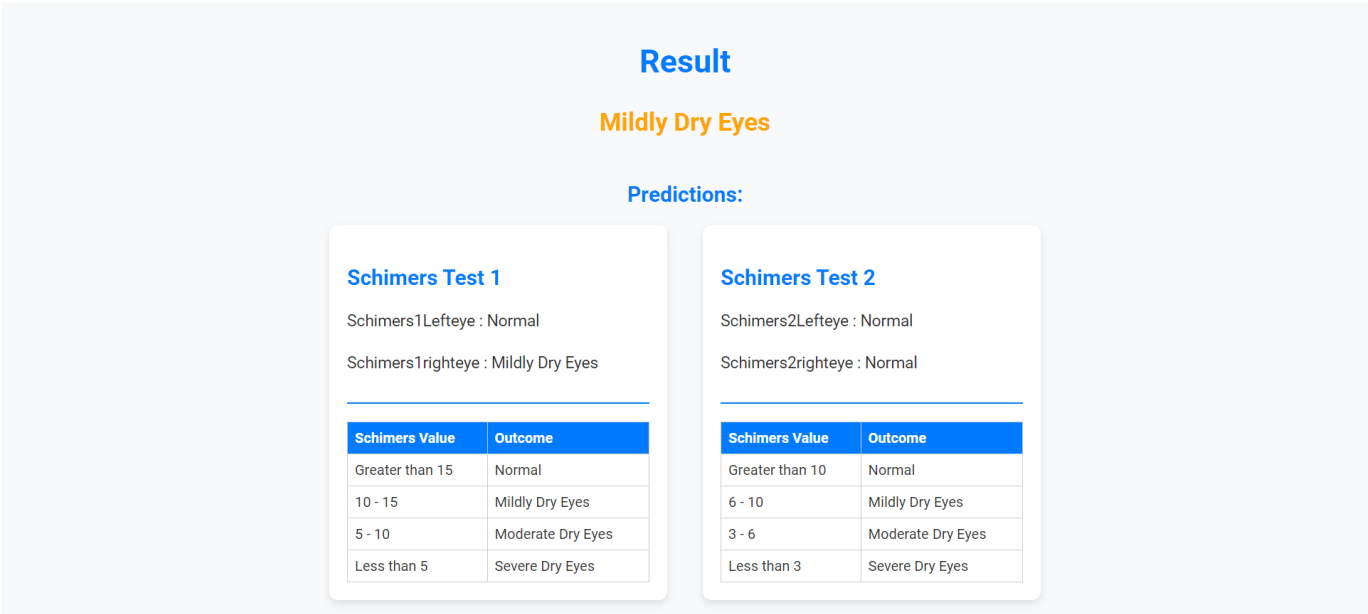


Fig 8.2 Result 2

The model predicts the user to have mildly dry eyes. Specifically, while Test1 indicates a mix of normal and mildly dry conditions for both left and right eyes, Test2 indicates normal conditions for both eyes. Despite the variation in predictions between the two tests, the overarching conclusion suggests a presence of mild dryness in the user's eyes.

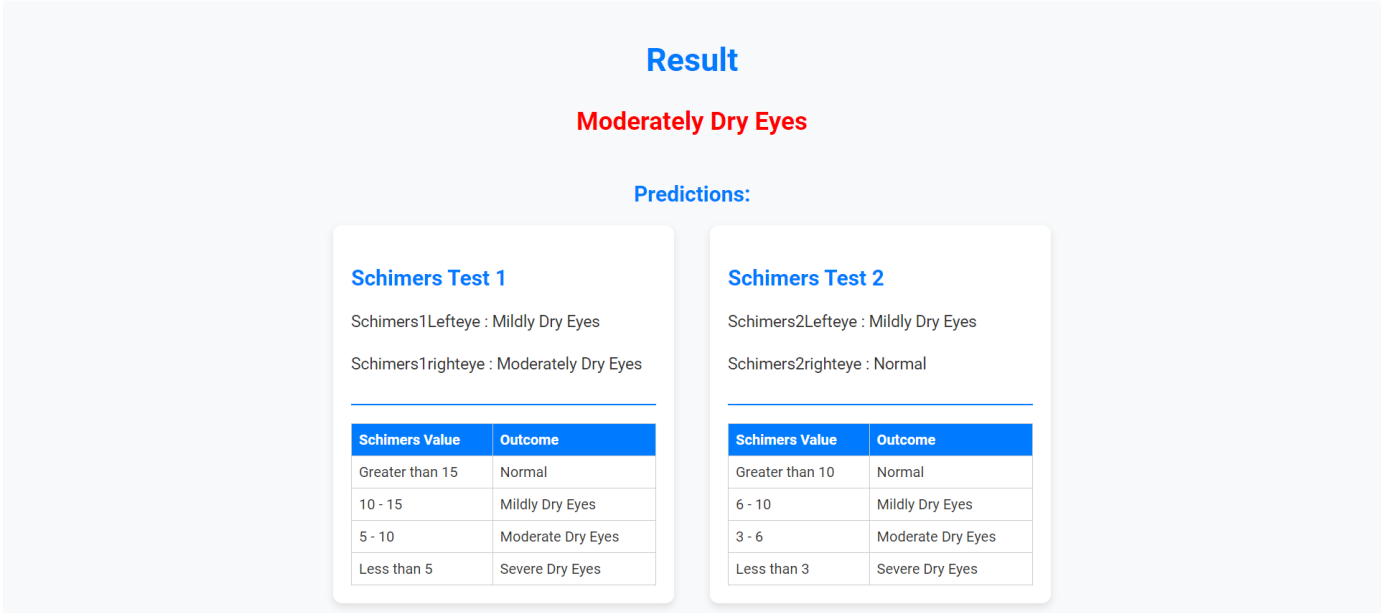


Fig 8.4 Result 3

According to the model's predictions, the user is likely affected by moderately dry eyes. Test1 indicates a range of conditions, with both left and right eyes showing predictions of mildly and moderately dry eyes. However, Test2 suggests a mix of mildly dry eyes and normal conditions for both eyes. Overall, while there is some variability between the tests, the model leans towards a diagnosis of moderately dry eyes for the user.

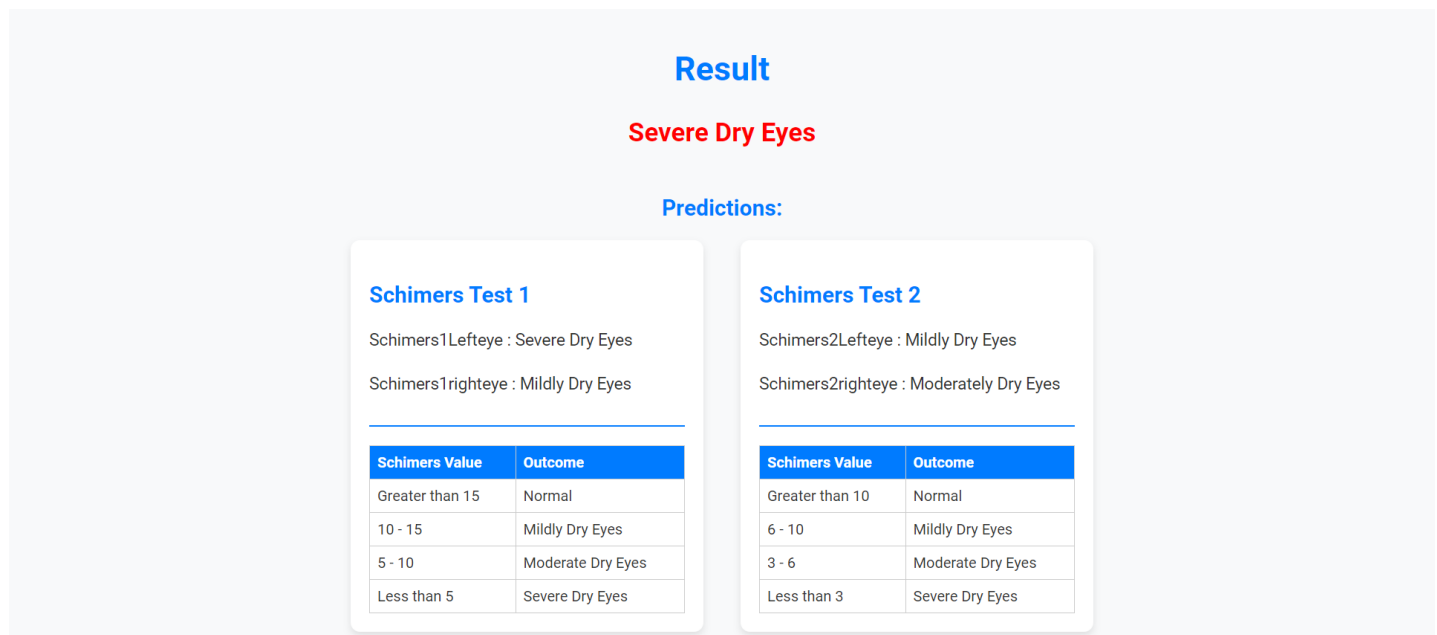


Fig 8.5 Result 4

The model predicts the user to have severe dry eyes. Specifically, while Test1 indicates a mix of severe dryness and mildly dry conditions for both left and right eyes, Test2 indicates mildly dryness, moderately dryness conditions for both eyes. Despite the variation in predictions between the two tests, the overarching conclusion suggests a presence of Severe dryness for the user's eyes.

The model's predictions suggest varying degrees of dry eye syndrome for the user. In the first scenario, both Test1 and Test2 indicate normal conditions, implying the absence of dry eyes. However, in the second case, while Test1 suggests mild dryness, Test2 shows a combination of normal and mildly dry conditions, indicating a mild presence of dry eyes. The third situation reveals predictions of moderate dryness, with Test1 indicating a mix of mild and moderate dryness, and Test2 suggesting a combination of mild and normal conditions. Finally, the model predicts severe dry eyes in the last scenario, with Test1 indicating severe and mild dryness, and Test2 showing a combination of mild and moderate dryness.

Overall, the model's assessments provide insights into the user's potential dry eye severity, considering variations between the tests and the overarching diagnosis.

These predictions collectively offer insights into the potential severity of dry eye symptoms for the user, considering the variations observed between the tests and the overall diagnostic trends.

Every time the user passes input parameter the model predicts based on those inputs and predicts the output.

9. CONCLUSION

In summary, Digital Eye Strain arises from prolonged exposure to digital screens, leading to symptoms such as redness, itchiness, blurred vision, headaches, backaches, and eye fatigue. This project endeavours to analyse various patterns and potential factors contributing to eye strain caused by digital screen usage.

Leveraging machine learning models, we aim to forecast the extent of eye dryness in individuals and identify habits conducive to mitigating the adverse effects of screen time on ocular health.

Throughout this project, we gained invaluable insights, spanning from data acquisition to model development. This encompassed managing missing data and outliers, conducting exploratory data analysis (EDA), deciphering diverse patterns, identifying potential influencers, and much more.

10.FUTURE SCOPE

1. Improving Features: We can look for more ways to describe how people use digital screens and how it affects their eyes.
2. Better Models: We can try using smarter computer programs to make more accurate predictions about eye health from the data.
3. Long-Term Studies: We can study people's screen habits and eye health over a long time to see how things change and learn more about how screens affect eyes.
4. Using Telemedicine: We can use technology to let doctors help people with their eye health from far away.

11. REFERENCES

1. <https://trysakai.longsight.com/portal/site/520b8591-6130-4776-bf04-504e18e885f1/tool/e59fe5bb-2464-4b5c-9679-4d9af5857ede?panel=Main>
2. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9434525/>
3. www.youtube.com