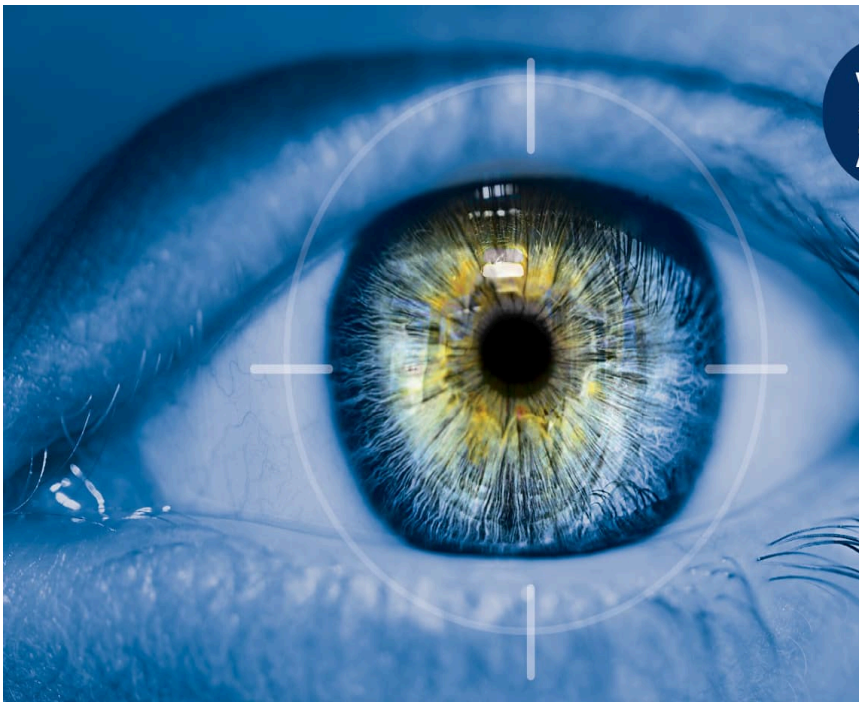
Focus on Vision:
A Statistical Study of Eye



A Project Report
Submitted by

Mr. Utkarsh Avinash Mishra

Sr No.	Topic	Page Number
1	Introduction	1
2	Objectives of the Study	2
3	Review of Literature	3
4	Procedure	4
5	Data Collection and Preprocessing	6
6	Statistical Testing	7
7	Data Exploration	11
8	Hypothesis Testing	18
9	Data Transformation	21
10	Synthetic Minority Oversampling Technique	24
11	Stepwise Logistic Regression	27
12	Logistic Regression	30
13	NeuralNet Binary Classifier	34
14	Streamlit Application	37
15	Conclusion	38
16	Future Work	39
17	References	40

1. Introduction

Today, we spend more time than ever in front of screens—phones, laptops, tablets—and it's starting to show in our eyes. Eye problems like strain, blurred vision, and even the need for glasses is becoming more common, especially among young people. According to the World Health Organization, over 2.2 billion people now suffer from some form of vision issue, and our digital habits are playing a big role.

But it’s not just screen time that affects our eyes. Things like poor lighting, lack of sleep, limited outdoor activity, and even family history can all contribute to vision problems. That’s why looking at just one factor isn’t enough—we need to look at the bigger picture.

This project, called **ClarityCare**, aims to do just that. We use data from many areas—like screen use, sleep, exercise, and family history—to better understand who might be at risk of needing vision correction. After cleaning the data and checking it for patterns, we use different types of models to predict the risk, including a simple statistical model (logistic regression) and a more advanced deep learning model.

In the end, we turn this work into an easy-to-use web app. People can answer a few questions about their lifestyle, and the app gives them a personalized risk score, along with tips to protect their vision. Our goal is to help people take better care of their eyes—before problems get worse.

2. Objective

A. Visualize and explore multiple data domains—demographics, screen-usage behaviors, environmental factors, and genetic history—through interactive charts and dashboards to uncover key patterns and insights.

B. Formally test hypotheses about relationships among variables (e.g., screen time vs. vision problems) using non-parametric correlation analyses and Chi-square tests of independence.

C. Preprocess the data for modelling by correcting errors, encoding categorical features, scaling numeric variables, and applying SMOTE to address class imbalance.

D. Select the most informative predictors via stepwise logistic regression combined with L1 regularization, thereby reducing dimensionality and ensuring interpretability.

E. Train and evaluate a logistic regression model to infer the probability of current or past vision correction needs, validating its fit, significance, and predictive contribution of each feature.

F. Develop a deep-learning classifier in TensorFlow to benchmark and enhance predictive performance, culminating in a deployable Streamlit application for real-time user risk assessment.

3. Literature Review

The increasing dependency on digital devices has led to a global surge in vision-related issues, especially among young adults and students. According to the World Health Organization, over 2.2 billion people suffer from vision impairment, with digital eye strain being a rapidly rising contributor due to excessive screen exposure. Studies such as Rosenfield (2016) and Sheppard & Wolffsohn (2018) have documented the adverse effects of prolonged digital device use, including myopia progression, eye fatigue, and blurred vision. These findings inspired our investigation into digital behavior and its correlation with vision problems.

Recent research has also emphasized the importance of incorporating environmental, behavioral, and genetic factors into health modeling. For instance, Huang et al. (2020) identified a strong association between poor lighting conditions and vision decline, while Saw et al. (2019) emphasized genetic predisposition as a critical risk factor for myopia. These multi-dimensional influences suggested that a purely clinical or demographic approach would be insufficient. Instead, an integrated, data-driven analysis—combining screen habits, sleep, exercise, and family history—was necessary to gain comprehensive insights.

Moreover, the rising use of AI in healthcare encouraged us to explore machine learning methods for predictive modeling. Logistic regression remains a cornerstone for interpretable medical models (Hosmer et al., 2013), while recent works by LeCun et al. (2015) and Esteva et al. (2017) highlight the power of deep learning in medical diagnostics. Inspired by this, we incorporated both statistical and neural models into our approach.

Thus, this project addresses a timely and meaningful intersection of public health, digital behavior, and machine learning. By leveraging statistical testing, data preprocessing, regression modeling, and neural networks, our study aims to create an accessible tool—ClarityCare—to help users assess their vision risk based on lifestyle patterns, encouraging proactive care and early intervention.

4. Procedure

The project begins with **data preprocessing**, where column names are renamed, and missing values are handled—numerical columns are imputed using the median, and categorical ones with the mode. Irrelevant features like air_quality are dropped and later reintroduced using a district-to-AQI mapping. A key feature, has_or_had_glasses, is derived based on vision history and eye power. Columns with all missing values are removed, and data types are standardized. Human-entry errors and outliers are corrected and filtered to ensure data integrity.

Next, **normality testing** is performed using Shapiro–Wilk, Kolmogorov–Smirnov, Anderson–Darling tests, and Q-Q plots. All variables significantly deviate from normality ($p < 0.05$), leading to the use of **medians for central tendency** and non-parametric tests for analysis.

Exploratory Data Analysis (EDA) is visualized across multiple dashboard pages, covering demographics (age, gender, occupation), digital screen usage (device preference, time by age/gender/occupation), symptoms, lighting habits, outdoor activity, sleep duration, family vision history, and more. Key insights include high screen time among youth, limited outdoor activity, and widespread vision symptoms linked to digital behaviour and genetics.

In **hypothesis testing**, Spearman's rank correlation reveals strong positive correlation between eye powers and moderate negative associations between eye power and digital habits. Chi-square tests confirm significant relationships between eye problems and variables like age, gender, and occupation.

For **data transformation**, categorical features are label encoded (avoiding one-hot due to high dimensionality), and numerical features are scaled using Min-Max normalization. A train-test split is applied **before SMOTE** to prevent data leakage and retain test integrity. **SMOTE** is then used to balance the has_or_had_glasses target, originally skewed (93.18% class 1), yielding a 50-50 split in the training set.

Feature selection is performed via **Stepwise Logistic Regression with L1 (Lasso)** regularization, identifying 11 key predictors such as age, screen hours, sleep, reading time, and environmental/lifestyle factors. VIF analysis confirms no multicollinearity.

A **logistic regression model** is then trained, achieving a **Pseudo R^2 of 0.7210** and a significant **Likelihood Ratio Test ($p < 0.001$)**. All predictors are statistically significant, with "reading_hours" contributing the most (70.1%) to prediction, followed by dark usage, age, and sleep duration.

A **neural network classifier** (128–64–32 architecture with ReLU, batch norm, dropout, and sigmoid output) is also developed. Trained with Adam optimizer and early stopping, it

achieves **86.93% accuracy** and **AUC 0.6335**, showing good performance despite class imbalance challenges.

Finally, the project is deployed as a **Streamlit application**—**ClarityCare**. It collects user inputs through a conversational interface, processes the inputs via encoding and scaling, and predicts vision risk using the logistic model. The output includes a confidence-based risk category with a disclaimer and an option to restart the session.

5. Data Preprocessing

The dataset underwent several preprocessing steps to ensure consistency, completeness, and readiness for analysis. Initially, column names were renamed for clarity using a defined mapping. Basic data checks were performed, including identifying missing values and data types. A custom feature has_or_had_glasses was derived based on vision correction history and eye power measurements.

Columns with all missing values were removed, and specific irrelevant features like air_quality were dropped temporarily. Missing values in numerical columns were imputed using the median, while categorical values were filled using the most frequent category.

After imputation, the data types were standardized: categorical columns were explicitly converted to category type, and numeric columns were coerced to numeric format. Finally, a new air_quality feature was reintroduced by mapping districts to their respective AQI values using a predefined dictionary.

Several records were misspelled due to human error. Those values were replaced with appropriate values ensuring data integrity. Also indentified outlier records through logical searching and filtering; removed those records.

6. Statistical Test

A. Normality

In statistics, normality refers to the condition where data follows a normal distribution (bell curve). Many statistical techniques and tests assume that the underlying data follows a normal distribution because many phenomena in nature are approximately normal. Assessing normality helps determine the most appropriate statistical methods for analysis.

- Why Assess Normality?

Assessing normality is crucial because many statistical methods assume that data follows a normal distribution. Understanding why this assumption matters is key to ensuring the accuracy and validity of your analysis.

a. Variable Characteristics

When data is not normally distributed, the mean may no longer be a reliable measure of central tendency, as it can be heavily influenced by skewness or outliers. In such cases, the median is a better representation of the data's centre. The median is the middle value in a dataset, unaffected by extreme values, making it a robust measure for non-normal distributions. Using the median ensures that the analysis remains representative of the central tendency even in the presence of skewed or non-normal data.

b. Parametric Test Assumptions

Many parametric tests (e.g., t-tests, ANOVA, regression) assume normality. These tests rely on properties of the normal distribution like symmetry and consistent variance. If data deviates from normality, these tests can give biased results, incorrect p-values, and unreliable confidence intervals.

c. Hypothesis Testing and p-values

Many hypothesis tests rely on normality to calculate the distribution of test statistics. Non-normal data can distort p-values and confidence intervals, leading to incorrect conclusions. For instance, a skewed distribution might lead to inflated type I error rates (false positives).

d. Outliers and Distribution Shape

Normality testing helps identify outliers, skewness, and heavy tails. Outliers can have a large effect on statistical measures like the mean and variance. Skewed or heavy-tailed distributions violate normality assumptions, affecting the robustness of tests.

- Tests for Normality
  a. Shapiro-Wilk Test
  - **Test Statistic**: The Shapiro-Wilk statistic W is calculated as:

$$W = \frac{(\sum a_i x_i)^2}{\sum (x_i - \bar{x})^2}$$

Where: x_i are the ordered sample values and a_i are constants based on the sample size.

- **Hypothesis:**

  Null hypothesis (H_0): Data follows a normal distribution.

  V/S

  Alternative hypothesis (H_1): Data does not follow a normal distribution.

- **Decision Criterion:**
  i. **If the p-value is less than or equal to α (p≤α)**: At **α level of significance, reject the null hypothesis**. This means there is statistically significant evidence to suggest that the data are **not** normally distributed.

ii. **If the p-value is greater than α (p>α)**: At **α level of significance, do not reject the null hypothesis**. This means there is statistically significant evidence to suggest that the data are normally distributed.

b. Kolmogorov-Smirnov (K-S) Test

- **Test Statistic**: The test statistic is defined as:

$$D = \max |F(x) - S_n(x)$$

Where: *F(x)* is the CDF of the normal distribution and $S_n(x)$ is the empirical CDF of the sample.

- **Hypothesis**:

Null hypothesis (H_0): Data follows a normal distribution.

V/S

Alternative hypothesis (H_1): Data does not follow a normal distribution.

- **Decision Criterion:**
  - i. **If the p-value is less than or equal to α (p≤α)**: At **α level of significance, reject the null hypothesis**. This means there is statistically significant evidence to suggest that the data are **not** normally distributed.
  - ii. **If the p-value is greater than α (p>α)**: At **α level of significance, do not reject the null hypothesis**. This means there is statistically significant evidence to suggest that the data are normally distributed.

c. Anderson-Darling Test

- **Test Statistic**: The Anderson-Darling statistic A^2 is calculated as:

$$A^2 = -n - \frac{1}{n}\sum_{i=1}^{n}(2i-1)\left[\ln(F(Y_i)) + \ln(1 - F(Y_{n+1-i}))\right]$$

Where: *F(x)* is the CDF of the normal distribution.

- **Hypothesis**:

Null hypothesis (H_0): Data follows a normal distribution.

V/S

Alternative hypothesis (H_1): Data does not follow a normal distribution.

- **Decision Criterion:**
  - i. **If the p-value is less than or equal to α (p≤α)**: At **α level of significance, reject the null hypothesis**. This means there is statistically significant evidence to suggest that the data are **not** normally distributed.
  - ii. **If the p-value is greater than α (p>α)**: At **α level of significance, do not reject the null hypothesis**. This means there is statistically significant evidence to suggest that the data are normally distributed.

d. Q-Q Plot

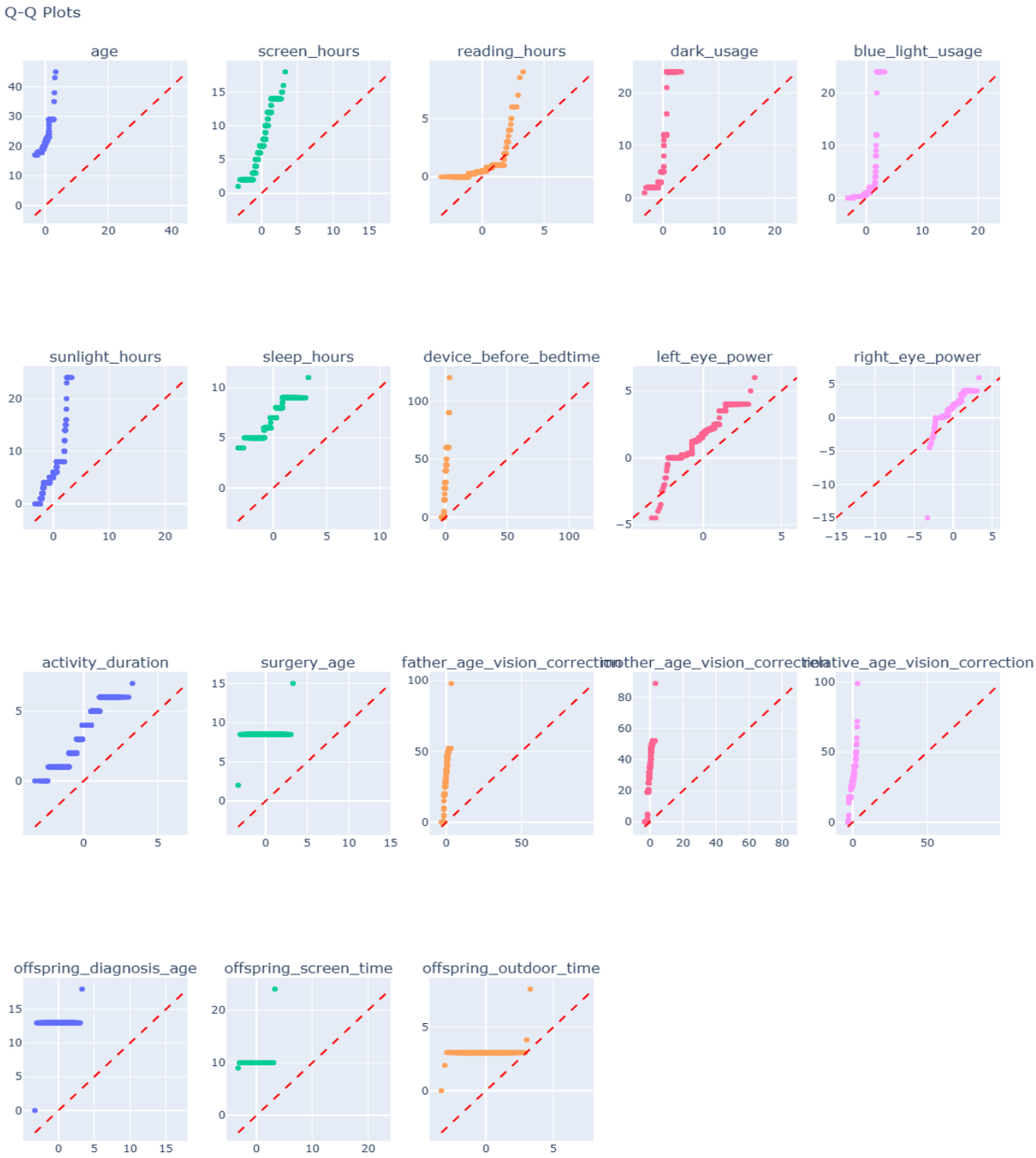
- **How It Works:**
  If the data is normally distributed, the points on the QQ plot will lie approximately on a straight line,but when points deviate from the line, it indicates that the data is not normally distributed.
- **Decision Criterion**:
  If the points lie close to the straight line, the data is normally distributed. If the points significantly
  deviate from the line (especially at the tails), the data does not follow a normal distribution.

Output:

No.	Variable	Shapiro-Wilk p-value	KS p-value	AD Statistic	AD Critical Value (1%)	Normality Conclusion
1	age	2.37×10^{-37}	3.80×10^{-54}	80.45	1.089	Not normal
2	screen_hours	4.37×10^{-23}	6.79×10^{-24}	24.36	1.089	Not normal
3	reading_hours	9.39×10^{-52}	1.10×10^{-98}	147.29	1.089	Not normal
4	dark_usage	8.77×10^{-41}	9.68×10^{-102}	134.37	1.089	Not normal
5	blue_light_usage	9.29×10^{-58}	1.01×10^{-228}	336.96	1.089	Not normal
6	sunlight_hours	8.11×10^{-44}	6.88×10^{-55}	77.14	1.089	Not normal
7	sleep_hours	1.46×10^{-29}	7.78×10^{-33}	48.17	1.089	Not normal
8	device_before_bedtime	6.39×10^{-25}	1.32×10^{-15}	27.07	1.089	Not normal
9	left_eye_power	6.25×10^{-23}	7.80×10^{-15}	22.29	1.089	Not normal
10	right_eye_power	3.46×10^{-31}	1.35×10^{-16}	23.83	1.089	Not normal
11	activity_duration	5.24×10^{-26}	1.08×10^{-29}	35.14	1.089	Not normal
12	surgery_age	1.56×10^{-64}	0	541.64	1.089	Not normal
13	father_age_vision_correction	1.18×10^{-25}	3.55×10^{-14}	22.08	1.089	Not normal
14	mother_age_vision_correction	1.58×10^{-26}	7.92×10^{-18}	26.02	1.089	Not normal
15	relative_age_vision_correction	1.48×10^{-29}	2.18×10^{-34}	32.7	1.089	Not normal
16	offspring_diagnosis_age	1.38×10^{-64}	0	541.84	1.089	Not normal
17	offspring_screen_time	1.12×10^{-64}	0	542.19	1.089	Not normal
18	offspring_outdoor_time	2.00×10^{-64}	0	539	1.089	Not normal

Note: Level of Significance(α) is set as 0.05

Q-Q Plot Output



Insights

a. Variable Characteristics

All variables show significant deviations from normality, making the mean an unreliable measure of central tendency due to its sensitivity to skewness and outliers. In contrast, the median offers a more robust and accurate reflection of the data's center, as it is unaffected by extreme values and better suited for non-normal distributions.

b. Highly Skewed or Heavy-Tailed Distributions

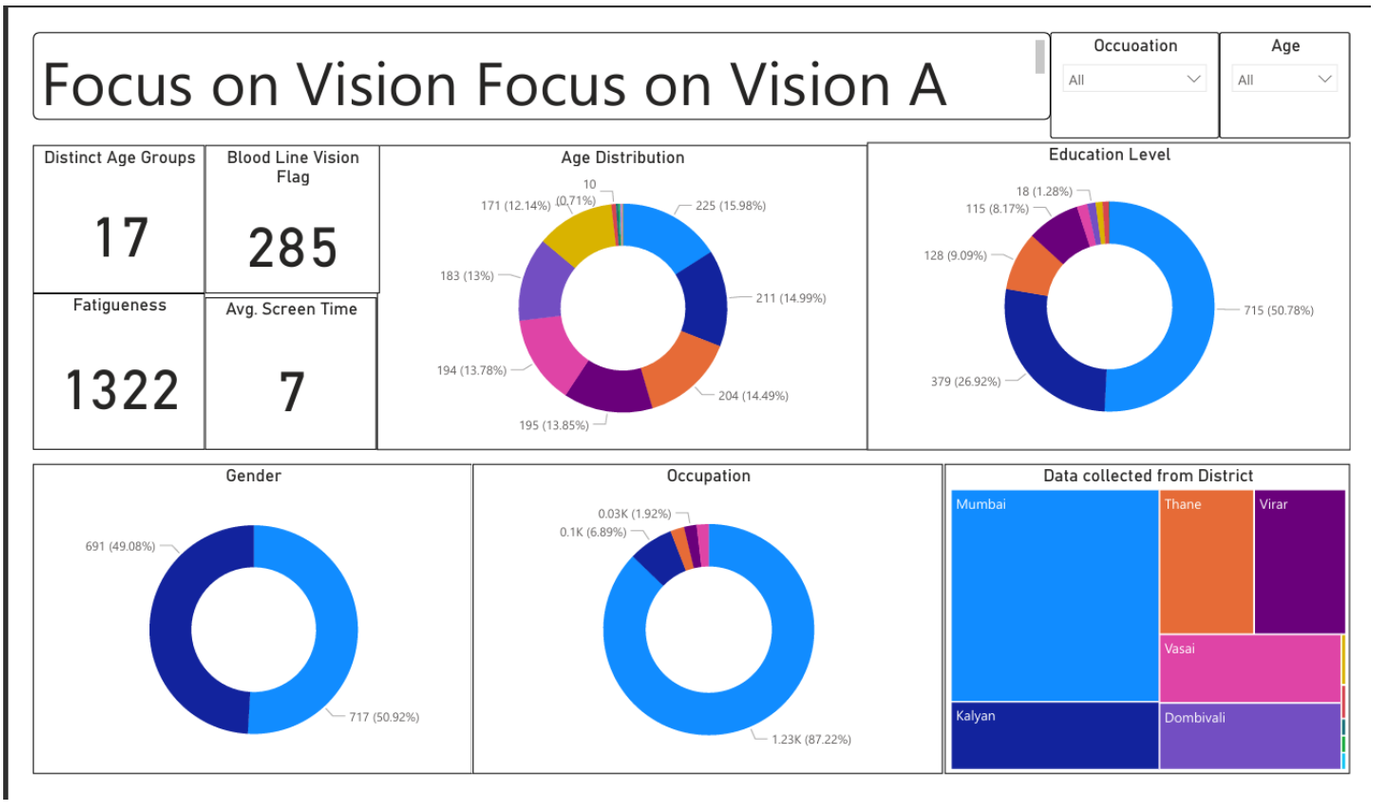
The extreme values of test statistics suggest that many variables may exhibit heavy tails, skewness, or both.

c. Implications for Statistical Analysis

Since the assumption of normality is violated: Parametric tests (e.g., t-tests, ANOVA, Pearson correlation) may not be appropriate. On-parametric alternatives (e.g., Mann-Whitney U test, Kruskal-Wallis test, Spearman correlation) should be considered.

7. Data Exploration

As our data has 61 variables and covers a variety of categories, the dashboard has been made into multiple pages. Below is the first page which focuses on demographics of the data to give a clear picture about the quality of the data.



Age Distribution: Spread across 17 distinct age groups. Most populated age group: 22 years (15.98%). Age groups 21, 20, 18, and 19 also show high representation.

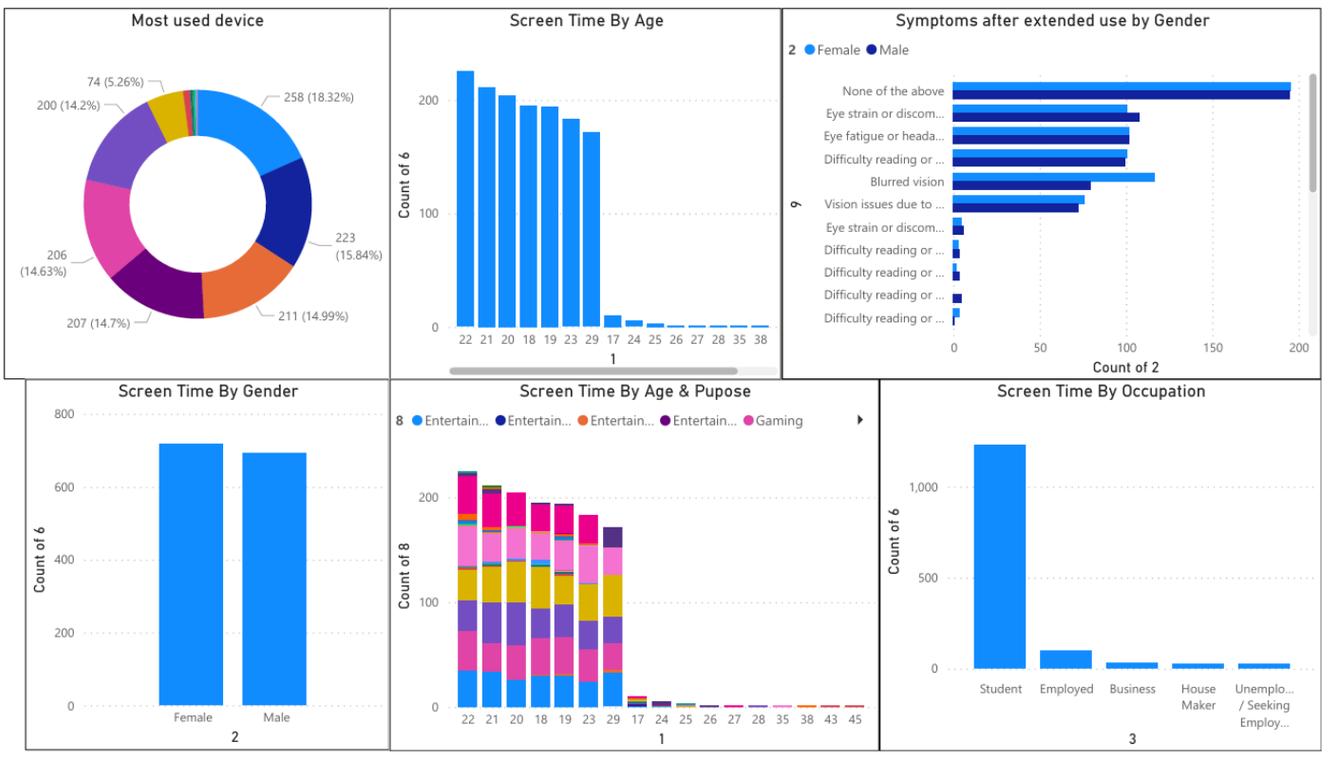
Gender: Near-equal split: Female: 50.92%. Male: 49.08%

Occupation: Majority are **students (87.22%)**. Small percentages are employed, business professionals, homemakers, or seeking employment.

Education Level: Most have **secondary (50.78%)** or **higher secondary (26.92%)** education. Graduates and postgraduates make up smaller portions.

Also, four KPI's are included in this page which includes:

- **Distinct Age Groups**: Had 17 different age groups in the dataset.
- **Blood Line Vision Flag**: Count of individual with their relative, parents and themselves having vision problems.
- **Fatigueness**: Individual feeling fatigued while reading or screen use
- **Avg. Screen Time**: Average screen time usage of every individual in the data.



This page contains the visualization regarding Digital Screen Usage & Behavior.

Most Used Device: Highest usage noted for **smartphones** (258 users or 18.32%). Followed by laptops and desktops.

Screen Time by Age: Peak usage for ages: **22, 21, 20, 18, and 19.** Steady decline as age increases beyond 29.

Screen Time by Occupation: Students show significantly higher screen usage compared to employed or homemakers.

Screen Time by Gender: Both genders exhibit comparable screen usage, with females slightly higher.

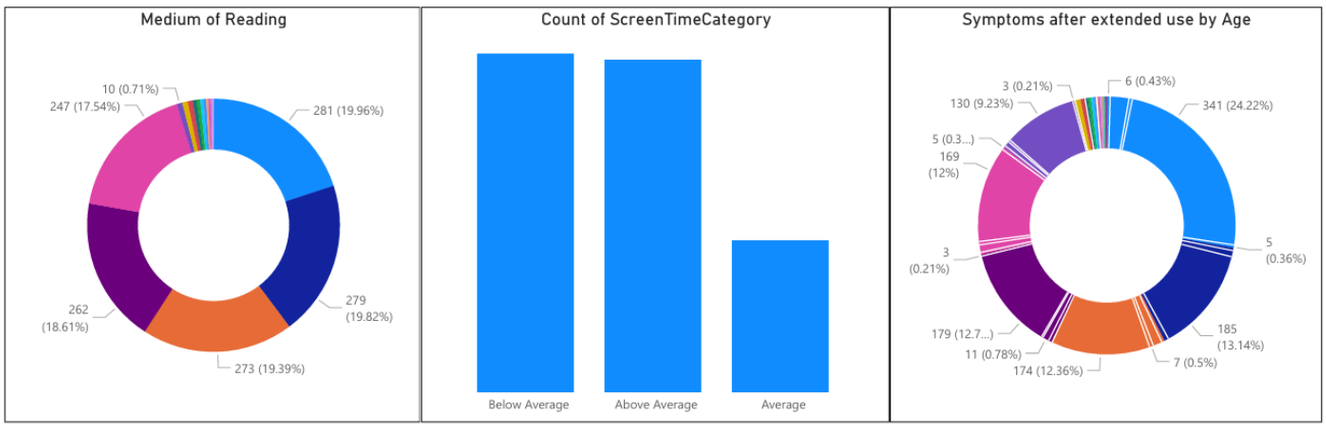
Symptoms After Extended Use by Gender: The symptoms experienced after extended screen use are compared between females and males, including issues like eye strain, eye fatigue, difficulty reading, blurred vision, vision issues and none. Majority experience none of the issues.

Screen Time by Age & Purpose: Screen time is broken down by age and purpose (e.g., Entertainment, Gaming). Majority are of 22 age group following 21,20 then 18. Majority of the purpose is either gaming, entertainment or social media.

Digital Screen Usage and Behavior

13	Never	Occasionally	Yes	**Total**
Dark Mode	11.25%	11.68%	13.84%	**36.77%**
Both equally	10.67%	9.88%	11.25%	**31.80%**
Light Mode	10.09%	10.45%	10.89%	**31.43%**
Total	**32.01%**	**32.01%**	**35.98%**	**100.00%**

13	Never	Occasionally	Yes	**Total**
Both equally	10.67%	9.88%	11.25%	**31.80%**
Dark Mode	11.25%	11.68%	13.84%	**36.77%**
Light Mode	10.09%	10.45%	10.89%	**31.43%**
Total	**32.01%**	**32.01%**	**35.98%**	**100.00%**



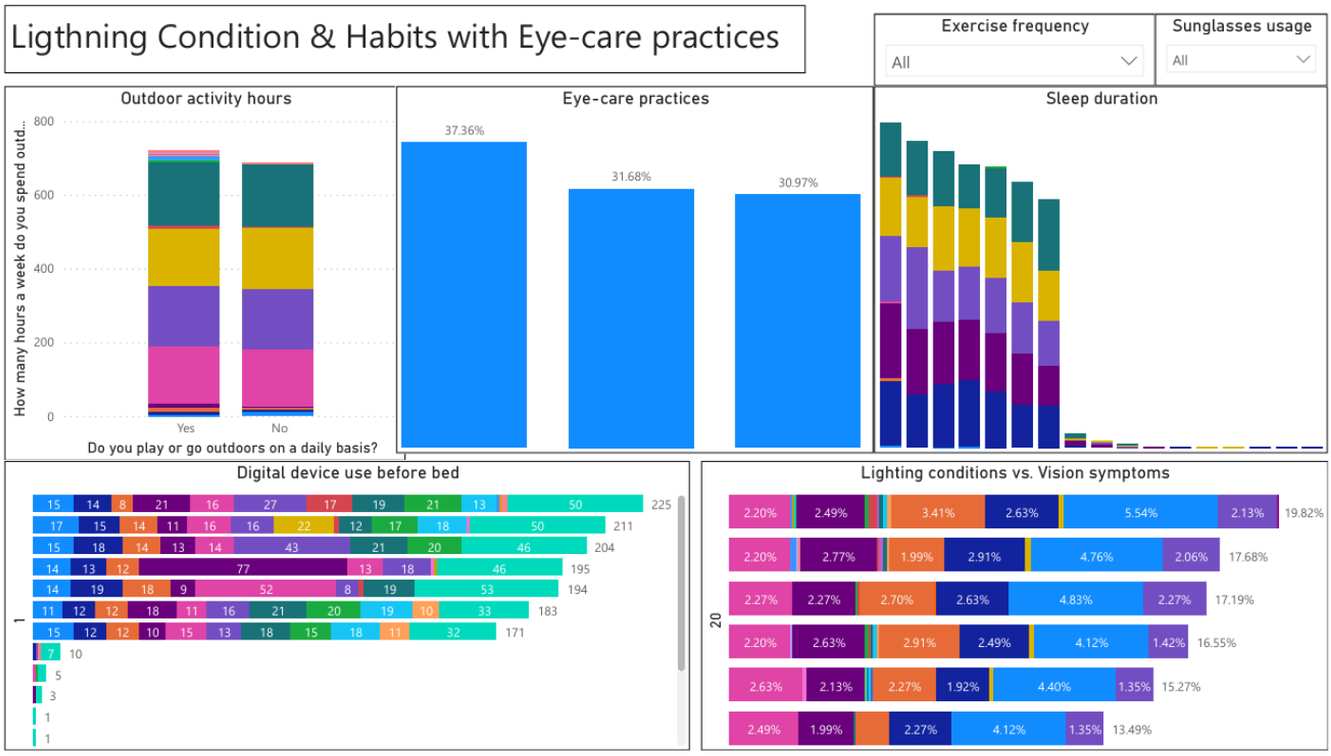
Above is the third page again focusing on the Digital Screen Usage & Behavior.

Medium of Reading: The medium of reading is categorized into categories; majorly in Smartphone/Tablet followed by Smartphone/Tablet/Laptop/PC, etc.

Count of Screen Time Category: Screen time is divided into categories: Below Average, Above Average, and Average, with corresponding counts. Majority of individual's screen time is below 7 hrs i.e. avg.screentimie.

Symptoms After Extended Use by Age: Symptoms after extended screen use are broken down by age, with counts for symptoms like eye strain, eye fatigue, difficulty reading, and blurred vision across age groups. Majority is of "None of the above".

Above tables show the Percentage of individual which uses night light or blue light filter in columns and usage of dark/light mode in rows. The difference between the two tables is that the first one shows the hours of usage of dark/light theme and for the second table is hours of usage of night/blue light filters.



Above table visualizes the “Lightning coonditions & habits with Eye-care practices”.

Outdoor Activity Hours: The number of hours spent on outdoor activity per week is presented categorized by hours; majority of individual spent around 8 hours followed by 4, 6 then 5 hours.

Eye-care Practices: Data on eye-care practices is included. Majority of individuals don’t take any precautions.

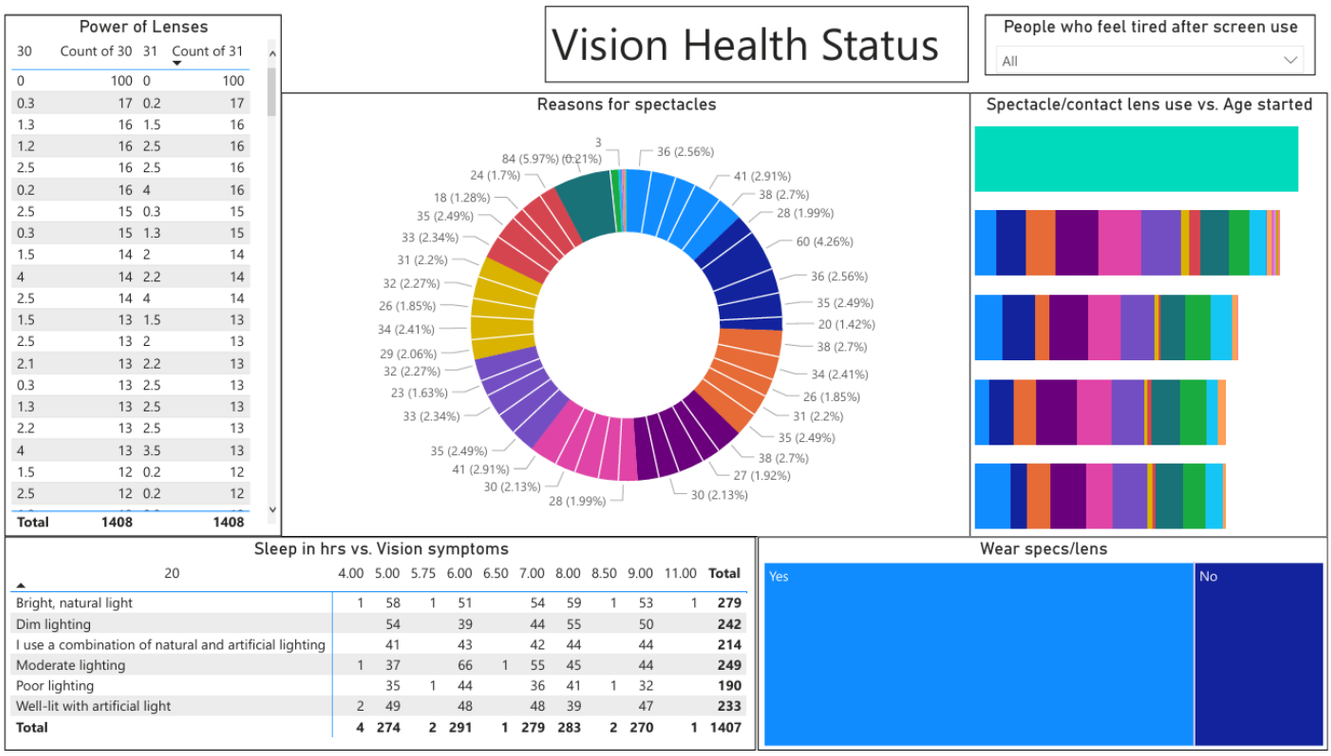
Exercise Frequency: To filter out individuals based on the exercise frequency as it can affect the health.

Sunglasses Usage: To filter out individuals based on the based on their sunglasses usage.

Sleep Duration: The sleep duration of the participants is analyzed into age groups. For example, in 22 years age group majority of individuals sleep for on an average 6 hours followed by 7 and 5.

Digital Device Use Before Bed: This visual shows the distribution of vision corrective tools in all age groups. Majority of individuals do not use any corrective tools.

Lighting Conditions vs. Vision Symptoms: This visual shows the distribution of vision symptoms categorized by lightning conditions.



Above page visualizes the Vision Health Status of individuals.

Power of Lenses: A table details the power of lenses and their counts (e.g., 0, 0.3, 1.3, 1.2, 2.5, 0.2, 4, etc.). The total count is 1408. First column shows the left eye power and the third column shows the right eye power along with their counts resp.

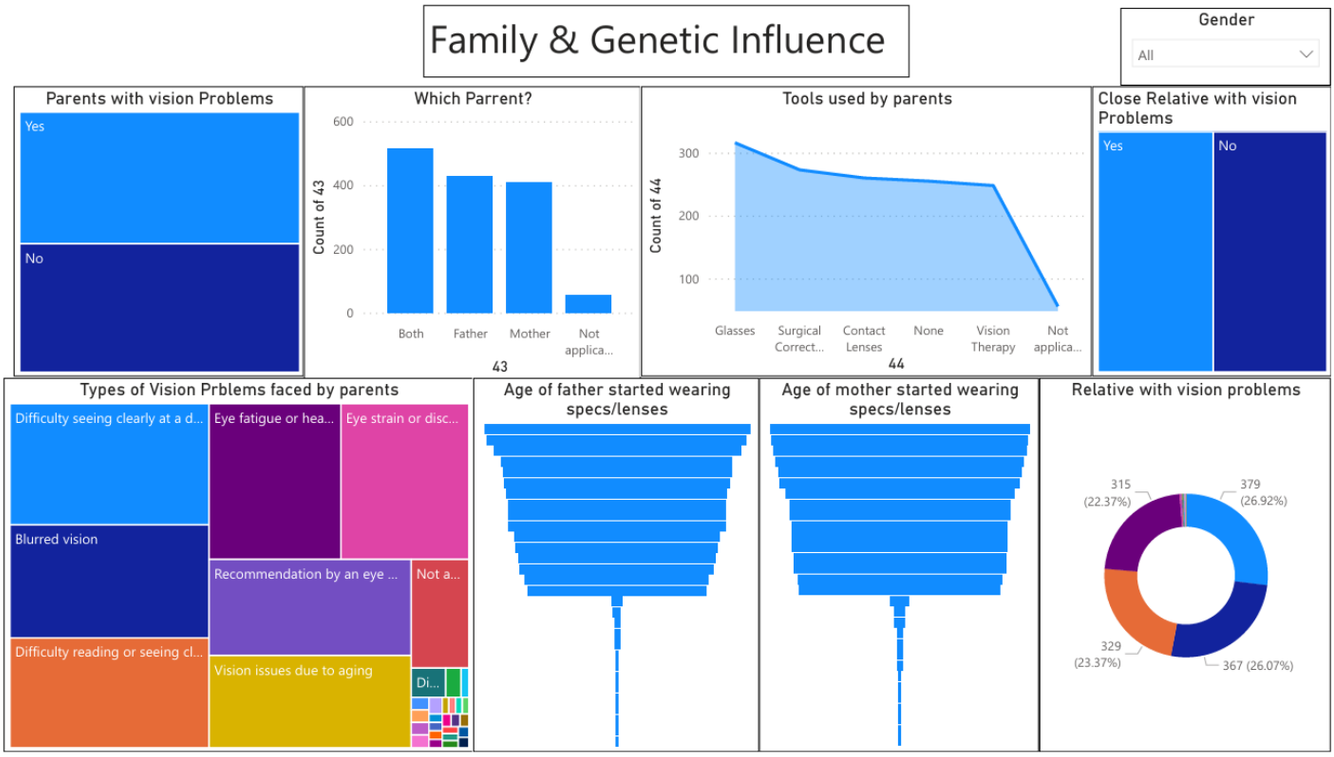
Reasons for Spectacles: Pie chart visualizes the reason for using any tools i.e spec or lenses.

People Who Feel Tired After Screen Use: Filters the data based on their tiredness.

Spectacle/Contact Lens Use vs. Age Started: The age at which individuals started using spectacles or contact lenses is analyzed.

Wear specs/lens: Shows the count of individuals having specs or not. Majority has specs.

Sleep in Hours vs. Vision Symptoms: A table shows the relationship between lighting conditions (Bright natural light, Dim lighting, Combination of natural and artificial lighting, Moderate lighting, Poor lighting, Well-lit with artificial light) and Hours of sleep.



Above page of the dashboard visualized the "**Family/Genetic Vision Health Status"**

Family & Genetic Influence: The dashboard explores the influence of family and genetics on vision problems.

Parents with Vision Problems: Data on whether parents have vision problems is presented, with counts shown(713 have vision problems).

Which Parent? : The specific parent (father or mother or both) with vision problems is identified.

Age of Father Started Wearing Specs/Lenses: The age at which fathers started wearing spectacles or lenses is analyzed.

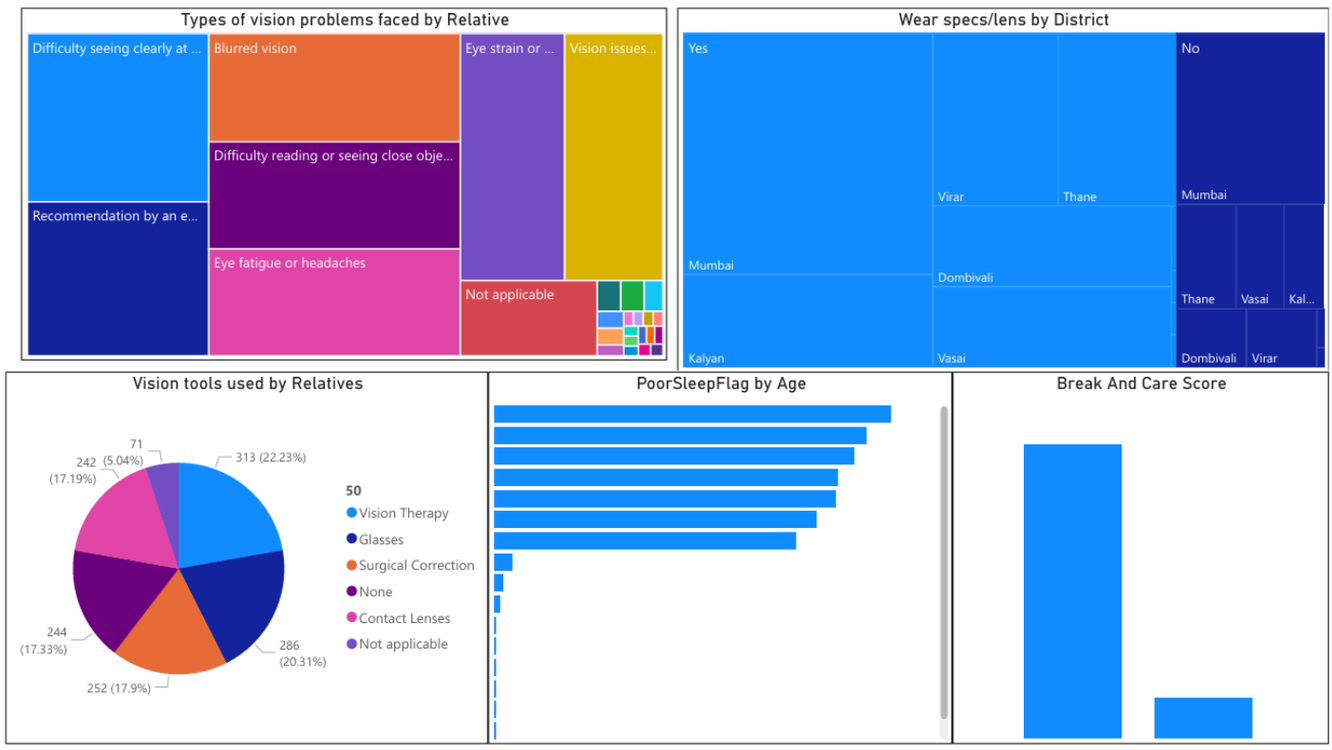
Age of Mother Started Wearing Specs/Lenses: The age at which mothers started wearing spectacles or lenses is analyzed.

Types of Vision Problems Faced by Parents: The types of vision problems experienced by parents are listed (e.g., Difficulty seeing clearly at a distance, Eye fatigue or headaches**)**

Tools used by parents: Majority of parents uses glasses followed by surgical corrective procedure and so on.

Close Relative with vision Problems: Data on whether close rerlative have vision problems is presented, with counts shown (708 have vision problems).

Relative with vision problems: Mostly relatives don't have any vision problems but not for the majority of the cases, as we can clearly see that majority of them suffer from anyone of the problems mentioned.



Above page is visualizes the **“Relative Vision Health”**

Types of Vision Problems Faced by Relative: The types of vision problems experienced by relative are listed (e.g., Difficulty seeing clearly at a distance, Eye fatigue or headaches**) .**

Wear specs/lens by District: Usage of specs shown in category of districts.

Vision tools used by Relatives: Majority of relatives uses vision therapy followed by glasses and so on.

PoorSleepFlag by Age: It’s an indicator showing the age groups with the worst sleep habits as they use digital device prior to their sleep.

Break And Care Score: Count of individuals who don’t take any breaks from screen use and do not take any precautions to reduce digital eye strain**.**

Insights

High screen time**, especially among students and youth (18–24), is the leading cause of symptoms like** eye strain, blurred vision, and fatigue**.**

Poor lighting**,** lack of outdoor activity**,** short sleep**, and** genetic predisposition **amplify the risk.**

Regular eye-care**, improved** environmental conditions**, and** reduced screen time **may significantly reduce symptoms.**

8. Hypothesis Testing

A. Correlation

Spearman's rank correlation coefficient (denoted as ρ or r_s) is a non-parametric statistical measure used to evaluate the strength and direction of a monotonic relationship between two variables. Unlike Pearson's correlation, Spearman's does not assume linearity or normality of the data.
It works by:

- Converting data values into **ranks**.
- Measuring how well the ranks of one variable are associated with the ranks of another.

It is particularly useful when:

- Data is **ordinal** or **non-normally distributed**.
- The relationship is **monotonic** but not necessarily linear.
- **Outliers** may distort parametric correlation results

- Why Assess Correlation?

Spearman's rank correlation coefficient, **ρ** (or $\mathbf{r_s}$), is a powerful statistical tool, particularly useful in various scenarios where traditional parametric methods, such as Pearson's correlation, might not be ideal. Here's why it's an important tool to assess:

a. **No Assumption of Normality**

Unlike Pearson, Spearman is non-parametric—it doesn't assume data is normally distributed. It can handle skewed, non-Gaussian distributions without requiring transformations.

b. **Captures Monotonic Relationships**

Spearman's correlation assesses monotonic relationships—where one variable consistently increases or decreases with another, even if the relationship isn't linear. This makes it ideal when data follows curves, plateaus, or thresholds that Pearson correlation might miss.

c. **Resistant to Outliers**

By converting values to ranks, Spearman minimizes the influence of extreme values that would otherwise distort Pearson correlation. This makes it more robust when dealing with anomalies or noisy data.

d. **Simple Rank-Based Calculation**

Because it uses ranks, Spearman's correlation is computationally straightforward—ideal for ordinal or irregularly spaced data where traditional linear measures fall short.

- **Test Statistic:** The Spearman correlation coefficient is calculated as

$$r_s = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

Where: d_i = difference between the ranks of the i^{th} pair
n = number of observations

Once you compute rsr_srs, if the sample is small (n ≤ 30), use exact critical values from a Spearman correlation table.
For large samples, you can approximate using a t-distribution:

$$t = \frac{r_s \sqrt{n-2}}{\sqrt{1 - r_s^2}}$$

- **Hypothesis**:

Null hypothesis (H_0): There is no monotonic association between the two variables. i.e. ρ = 0

V/S

Alternative hypothesis (H_1): There is a monotonic association between the two variables. i.e. $\rho \neq 0$ or $\rho > 0$ / $\rho < 0$

- **Interpretation:**

i. $\rho \approx +1$: Perfect positive monotonic relationship (as one variable increases, the other always increases).

ii. $\rho \approx -1$: Perfect negative monotonic relationship (as one variable increases, the other always decreases).

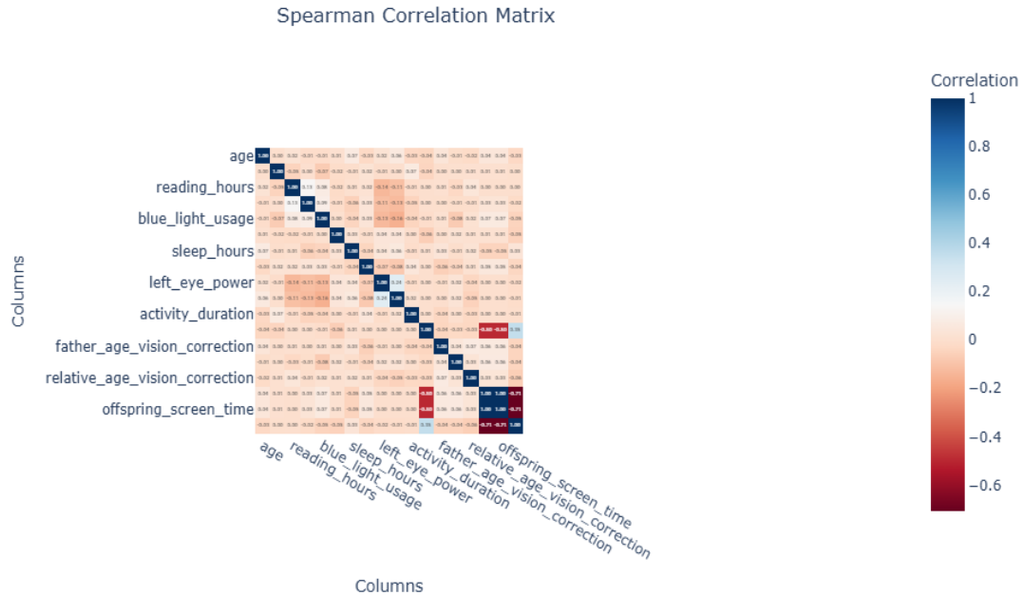
iii. $\rho \approx 0$: No monotonic relationship (no consistent pattern between the variables).

iv. $\rho > 0$: Positive monotonic relationship (as one variable increases, the other tends to increase).

v. $\rho < 0$: Negative monotonic relationship (as one variable increases, the other tends to decrease).

Output:

No.	Variable 1	Variable 2	Correlation	P Value	Statistical Strength
1	Right eye power	Left eye power	0.848	1.60E-20	Strong Positive
2	Reading_hours	Left eye power	-0.135	3.57E-07	Moderate Negative
3	Reading_hours	Right eye power	-0.112	2.74E-05	Moderate Negative
4	Dark_usage	Left eye power	0.114	1.76E-05	Moderate Negative
5	Dark_usage	Right eye power	-0.126	2.14E-06	Moderate Negative
6	Blue light usage	Left eye power	-0.131	8.03E-07	Moderate Negative
7	Blue light usage	Right eye power	-0.162	1.06E-09	Moderate Negative
8	device_before_bedtime	Left eye power	-0.065	0.015	Weak Negative
9	device_before_bedtime	Right eye power	-0.081	0.002	Weak Negative
10	Sleep hours	Eye power	- 0.058	0.030	Weak Negative
11	Age	Eye power	0.062	0.020	Weak Positive



B. Chi-Square Test of Independence

The Chi-Square Test of Independence (χ^2 test) is a statistical method used to determine whether there is a significant association between two categorical variables. It helps assess whether the distribution of one variable is independent of the distribution of another.A crosstab (contingency table) displays the frequency distribution of variables.

- Why Assess Chi-Square Test of Independence?
  Chi-Square Test of Independence, short for cross-tabulation, is a statistical tool used to analyze the relationship between two or more categorical variables by organizing data into a table format.

a. Relationships Between Categorical Variables

At its core, crosstab analysis helps us explore whether there is a relationship or association between two categorical variables.

b. Data Reduction and Simplification

AtCrosstabs condense complex datasets into a simplified matrix. This is powerful because:

- You can visualize a relationship at a glance.
- It transforms raw data into a structured summary.
- It's easy to interpret even without advanced statistical training.
- Especially useful when you're working with large datasets or survey results.

- **Test Statistic:** The test compares the observed frequencies in a contingency table to the expected frequencies, which are calculated under the assumption that the variables are independent. The formula for the Chi-Square statistic is:

$$\chi^2 = \sum \frac{(O - E)^2}{E}$$

Where: O = Observed frequency
E = Expected frequency, calculated as:

$$E = \frac{\text{Row Total} \times \text{Column Total}}{\text{Grand Total}}$$

Determine the degrees of freedom (df):

$$df=(r-1)\times(c-1)$$

Where: r = number of rows in the crosstab (excluding totals)
c = number of columns (excluding totals)

- **Hypothesis**:
  Null hypothesis (H_0): The two categorical variables are independent (no association).
  V/S
  Alternative hypothesis (H_1): The two categorical variables are associated (dependent).
- **Interpretation:**
  i. **If the p-value is less than or equal to α ($p \leq \alpha$) or $\chi^2_{calculated} > \chi^2_{critical}$**: At **α level of significance, reject the null hypothesis**. This means there is **no association** between the two categorical variables.
  ii. **If the p-value is greater than α ($p > \alpha$) or $\chi^2_{calculated} \leq \chi^2_{critical}$**: At **α level of significance, do not reject the null hypothesis**. This means there is **association** between the two categorical variables.

Output:

No.	Variable 1	Variable 2	χ^2	P Value	Degree of Freedom	Statistical Association
1	Gender	Eye Problem	7.761	0.0053	1	Significant
2	Relative Vision Problems	Eye Problem	855.760	2.04E-164	25	Highly Significant
3	Occupation	Eye Problem	485.570	1.77E-82	32	Significant
4	Age	Eye Problem	604.494	1.37E-254	100	Highly Significant

9. Data Transformation

A. Label Encoding

In many real-world datasets, features may include categorical data, such as gender, education level, occupation, or product category. Since most machine learning algorithms are designed to operate on numerical input, categorical variables must be transformed into a numeric format—a process known as encoding.Encoding is an essential step in data preprocessing that ensures categorical features are machine-readable, without altering the underlying information they represent.

- Why Assess Label Encoding?
    a. Numerical Compatibility: Most ML models (e.g., regression, SVM, KNN, neural networks) operate on mathematical operations that require numeric inputs.
    b. Model Performance: Proper encoding preserves the semantic meaning of categories and enhances model accuracy.
    c. Consistency: Encoding ensures a standardized data format, enabling seamless training, validation, and deployment of models.

Rationale for Using Label Encoding:

a. Dimensionality and Sparsity

One-Hot Encoding increases the feature space significantly, especially with 39 categorical variables. It produces sparse binary matrices that consume more memory and computational resources. From a statistical standpoint, this leads to the curse of dimensionality, which increases the risk of overfitting and slows down convergence. Label Encoding keeps the dataset compact with minimal sparsity, improving generalization and efficiency.

b. Manual Encoding Flexibility

Manual Label Encoding gave us full control over value mapping, ensuring consistency across training and deployment. It also improved interpretability and removed the need for external encoders or saved mappings, simplifying the preprocessing pipeline.

c. Manual Encoding Flexibility

Let:

- C be the number of categorical features
- n_i be the number of unique categories in feature iii
- $N=\sum_{i=1}^{C} n_i$

Then:

- Label Encoding transforms C columns into C numeric columns (space complexity: O(C))
- One-Hot Encoding transforms C columns into N binary columns (space complexity: O(N))

Given the dataset includes over 39 categorical features, One-Hot Encoding would result in a significant blow-up in dimensionality, degrading computational performance and increasing RAM consumption without statistical benefit for non-linear models.

B. Feature Scaling

The Min-Max Scaler transforms each feature in a dataset to a specified range, typically [0,1] or [−1,1], by linearly scaling the data based on its minimum and maximum values. It preserves the relative relationships (intervals) between data points while ensuring all features lie within the same range.

Key Objectives of Normalization:

- Scale Invariance: Ensures equal contribution from all features.
- Improved Convergence: Speeds up and stabilizes optimization processes like gradient descent.
- Distance-Based Fairness: Prevents features with larger ranges from affecting algorithms like KNN or clustering.
- Numerical Stability: Mitigates issues from large differences in feature values.

The Min-Max Scaler is a popular normalization technique, known for its simplicity and effectiveness in mapping data to a fixed range.

- Min-Max Scaler Formula

The Min-Max Scaler applies a linear transformation to each feature x in a dataset to produce a scaled feature x'. The formula for scaling a feature to the range [a,b] (commonly [0,1]) is:

$$x' = a + \frac{(x - x_{min})(b - a)}{x_{max} - x_{min}}$$

Where: x: Original feature value.

x_{min}: Minimum value of the feature in the dataset.

x_{max}: Maximum value of the feature in the dataset.

a,b: Desired output range (e.g., a=0,b=1 for [0, 1]).

- Assumptions
  a. Finite Range: The feature has a well-defined minimum and maximum in the training data. Outliers or extreme values can significantly affect the scaling.
  b. Linear Relationship Preservation: The linear transformation is appropriate for the problem, and no nonlinear scaling (e.g., logarithmic) is required.
  c. No Distributional Assumption: Unlike standardization, Min-Max Scaler does not assume the data follows a Gaussian or any specific distribution, making it distribution-agnostic.

Rationale for Using Min-Max Scaler:

a. Non-Normal Distribution of Data

The dataset contains both numerical (e.g., age, screen_hours) and categorical/ordinal features (e.g., gender, vision_correction). Tests and visualizations show that many numerical features are not normally distributed, often skewed or bounded within specific ranges.

The Standard Scaler assumes normality, which can distort data relationships when this assumption is violated. The Min-Max Scaler, in contrast, does not rely on normality and scales features to a fixed range, preserving the relative relationships regardless of distribution.

b. Preserving Bounded and Interpretable Ranges

Many features like age or screen_hours have natural bounds (e.g., age between 0-100). The Min-Max Scaler preserves these bounds by scaling to a [0, 1] range, maintaining interpretability.The Standard Scaler, however, can produce values outside of meaningful ranges, making it less interpretable.

c. Handling Outliers and Skewness

The Min-Max Scaler is sensitive to outliers but fits my project since I've preprocessed outliers appropriately. The Standard Scaler can also struggle with skewed data as it's influenced by the mean and standard deviation, which is less optimal for non-normal data.

C. Train-Test Split

The train-test split is a fundamental technique in machine learning and statistical modeling used to evaluate the performance of a model by dividing a dataset into two subsets: a training set used to fit the model and a **test set** used to assess its generalization performance on unseen data. This approach simulates how a model will perform in the real world, where it encounters new, previously unseen data.

- Why Assess Label Encoding?
  a. Generalization Estimation

Separating training and testing data simulates real-world deployment, providing a realistic measure of how well the model generalizes. Training on the same data it's evaluated on leads to overly optimistic results due to overfitting.

  b. Managing the Bias-Variance Tradeoff

The train-test split helps evaluate the balance between **bias** (underfitting, where the model is too simple) and **variance** (overfitting, where the model is too complex).

A model that performs well on the training set but poorly on the test set indicates high variance (overfitting). Conversely, poor performance on both sets suggests high bias (underfitting).

c. Model Selection and Optimization

The test set provides a consistent benchmark for comparing different models or hyperparameter configurations, enabling data-driven decisions about which model to deploy.

Rationale for Using Train-Test Split:

In this project, I split the data into training and test sets before applying SMOTE (Synthetic Minority Oversampling Technique) to the training set only. This deliberate choice avoids common pitfalls in model evaluation and improves generalizability, especially when dealing with imbalanced targets like screen fatigue or vision issue

a. Non-Normal Distribution of Data

Applying SMOTE before the split risks generating synthetic samples based on instances that later appear in the test set. This leads to data leakage, where the model indirectly learns from test data during training—artificially boosting performance metrics.

Splitting first ensures the test set remains untouched and independent, allowing for valid, leakage-free evaluation.

b. Ensuring Realistic Evaluation

If the test set includes SMOTE-generated samples, it no longer reflects the real-world imbalance. This can inflate scores like recall or F1 for the minority class, but doesn't represent how the model performs on genuine data.

Only the training set is balanced with SMOTE, so the test set retains the original imbalance—yielding performance metrics that better represent deployment conditions.

c. Improving Generalization

Training and testing on synthetic data can cause overfitting to SMOTE-specific patterns rather than true minority class behavior. This weakens the model's ability to handle real, unseen data.

By isolating SMOTE to the training set, the model learns generalizable patterns while being tested against real-world distributions—enhancing reliability in practical use.

10. Synthetic Minority Oversampling Technique

SMOTE is a data augmentation technique designed to address class imbalance in classification problems by generating synthetic samples for the minority class. It is widely used in applications like fraud detection, medical diagnosis, and anomaly detection, where the minority class is critical but underrepresented.

Key Objectives of SMOTE

- Balance Class Distribution: Reduce classifier bias toward the majority class.
- Improve Minority Class Performance: Enhance recall, precision, and F1-score for the minority class.
- Preserve Data Structure: Generate synthetic samples reflecting the minority class distribution.
- Enable Robust Learning: Ensure classifiers learn meaningful minority class patterns.

- Why Assess Normality?
  a. Mitigate Bias: Prevents classifiers from favoring the majority class in imbalanced datasets.
  b. Improve Metrics: Enhances minority class performance, critical in high-stakes applications.
  c. Avoid Overfitting: Generates synthetic samples to reduce memorization compared to naive oversampling.
  d. Enhance Decision Boundaries: Helps classifiers define better boundaries around minority samples.
  e. Statistical Robustness: Improves reliability of performance estimates.

- SMOTE Mathematical intuition
  SMOTE generates synthetic samples for the minority class by interpolating between existing minority class samples using their nearest neighbors. It operates in the feature space and leverages the k-nearest neighbors (k-NN) algorithm to create new, plausible data points.

  SMOTE Algorithm Steps:

  a. Identify Minority Class Samples:
  Let $X=\{x_1,x_2,\ldots,x_n\}$ be the set of minority class samples, where each $x_i \in R^d$ is a feature vector in d-dimensional space.

  b. Select a Minority Sample:
  For each minority sample x_i, compute its k-nearest neighbors within the minority class using a distance metric (typically Euclidean distance).
  c. Generate Synthetic Samples:
  Randomly select one of the k-nearest neighbors, say x_{nn}.
  Create a synthetic sample x_{new} by interpolating between x_i and x_{nn}:

  $$x_{new} = x_i + \lambda^* (x_{nn} - x_i)$$

  Where: $\lambda \sim$ Uniform(0,1) is a random scalar controlling the interpolation.
  d. Repeat:
  Repeat the process until the desired number of synthetic samples is generated, typically until the minority class is balanced with the majority class or a specified ratio is achieved.
  e. Combine Data:
  Append the synthetic samples to the original dataset, creating a balanced training set.

Rationale for Using SMOTE:

In this project, I used SMOTE (Synthetic Minority Oversampling Technique) to balance the severely imbalanced training data for the target variable 'has_or_had_glasses' (93.18% class 1 vs. 6.82% class 0). This decision was essential for building a fair, generalizable, and practically useful model.

a. SMOTE Need

Class Imbalance: The dataset is severely imbalanced, with 93.18% of instances in the majority class ('has_or_had_glasses' = 1) and only 6.82% in the minority class ('has_or_had_glasses' = 0). This skew can bias the model, as it tends to optimize for accuracy, dominated by the majority class.

Statistical Implications: In imbalanced datasets, the model may achieve high accuracy (e.g., 93% by always predicting the majority class) but fail to predict the minority class accurately. This results in poor recall, precision, and F1-score for the minority class. In practical settings (e.g., vision health), failing to identify the minority class can have significant consequences.

SMOTE's Role: SMOTE combats this by generating synthetic samples for the minority class, balancing the class distribution in the training set. This enables the model to learn from both classes effectively. For my project, SMOTE was applied only to the training set after the train-test split to prevent data leakage and ensure a realistic evaluation.

b. Why Not Build the Model with the Original Imbalanced Data

Impact of Imbalanced Data: Using the original imbalanced data leads to biased learning, where the model focuses on the majority class ('has_or_had_glasses' = 1) due to its overwhelming representation.

Low Sensitivity to the Minority Class: The model may rarely predict the minority class ('has_or_had_glasses' = 0), resulting in near-zero recall. With only 6.82% of the data in the minority class, the model might ignore these instances and still achieve high accuracy.

Skewed Decision Boundaries: In imbalanced datasets, algorithms like logistic regression or neural networks create decision boundaries that favor the majority class, reducing the model's ability to detect minority class instances.

Poor Generalization: While the model may perform well on the majority class in training, it will struggle in real-world scenarios, where identifying the minority class is crucial (e.g., vision correction for targeted interventions).

Statistical Implication: The loss function (e.g., cross-entropy) is dominated by the majority class, causing model parameters to be optimized for majority class predictions. This leads to underfitting of the minority class, limiting the model's practical utility.

c. SMOTE Effects Model Parameters

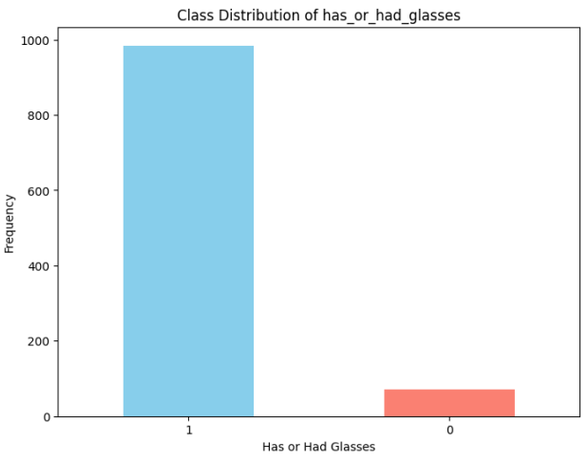
Balancing the Loss Function: SMOTE generates synthetic minority class samples, balancing the loss function across classes. This leads to more equitable coefficient adjustments in models like logistic regression, improving the decision boundary between classes.

Improved Gradient Updates: In gradient-based models, SMOTE ensures that gradients from the minority class are weighted appropriately, preventing dominance by the majority class and improving parameter optimization.

Feature Space Representation: By interpolating features like age, screen hours, and vision correction, SMOTE enriches the minority class representation, helping models like decision trees and KNN capture important patterns.

Statistical Implication: SMOTE optimizes model parameters to minimize errors for both classes, achieving a better balance between sensitivity and specificity, and reducing bias toward the majority class.

Input:

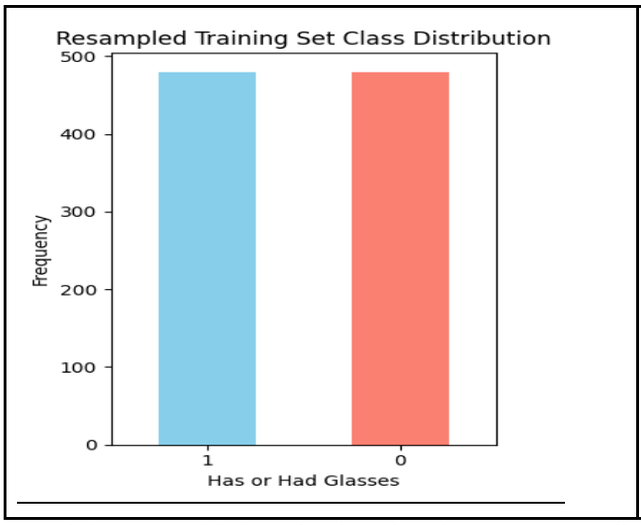



```
Class Distribution:
has_or_had_glasses
1    984
0     72
Name: count, dtype: int64

Class Percentages:
has_or_had_glasses
1    93.181818
0     6.818182
Name: count, dtype: float64

Conclusion: The dataset is imbalanced.
```

The dataset shows a pronounced class imbalance in the has_or_had_glasses feature. The majority class (1 - individuals with current or past glasses use) accounts for approximately 93.18% (984 instances), while the minority class (0 - individuals without glasses) represents only 6.82% (72 instances).

This imbalance poses a risk of model bias, where predictive performance skews toward the majority class. As a result, the model may fail to accurately identify individuals from the minority group. To ensure fairness and robustness, it is crucial to address this imbalance before training the model.

Output:




```
Class Distribution:
has_or_had_glasses
1    786
0    786
Name: count, dtype: int64

Class Percentages:
has_or_had_glasses
1    50.0
0    50.0
Name: count, dtype: float64

Conclusion: The dataset is balanced or nearly balanced.
```

After applying SMOTE (Synthetic Minority Over-sampling Technique), the class distribution of has_or_had_glasses has been balanced, with 786 instances per class, resulting in 50% representation for both. This transformation addresses the original imbalance, enhancing model fairness and improving the ability to accurately predict both classes—especially the previously underrepresented minority class.

11. Stepwise Logistic Regression

Logistic Regression is a statistical model for binary classification, predicting the probability of a binary outcome based on one or more predictor variables. Stepwise Logistic Regression is a feature selection method that iteratively adds or removes predictors based on a criterion like AIC, BIC, or p-value. L1 Regularization (Lasso) adds a penalty to the logistic regression objective, shrinking coefficients and driving some to zero for implicit feature selection.

Combining Stepwise Logistic Regression with L1 Regularization utilizes the strengths of both methods: iterative model building through stepwise selection and sparsity through L1 regularization, helping with overfitting and feature selection. This hybrid approach is ideal for high-dimensional datasets or when model interpretability is key.

Key Objectives:

- Feature Selection: Identify relevant predictors to improve interpretability and reduce overfitting.
- Regularization: Penalize large coefficients to control model complexity and enhance generalization.
- Predictive Performance: Balance bias and variance for robust binary outcome predictions.
- Statistical Inference: Provide interpretable models with statistically significant predictors.

- Stepwise Logistic Regression with L1 Regularization Mathematical intuition

Logistic regression models the probability of a binary outcome $y \in \{0,1\}$ given predictors $x=(x_1,x_2,\ldots,x_p)$. The probability is modeled using the logistic function:

$$P(y=1|\mathbf{x}) = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)}}$$

The model is typically fit by maximizing the log-likelihood function:

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} [y_i \log(P(y_i=1|\mathbf{x}_i)) + (1-y_i)\log(1-P(y_i=1|\mathbf{x}_i))]$$

L1 regularization adds a penalty term to the log-likelihood, resulting in the following objective function to minimize:

$$-\ell(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j|$$

Where:

- $\ell(\beta)$: Log-likelihood of the logistic regression model.
- λ: Regularization parameter controlling the strength of the penalty ($\lambda>0$).
- $\sum_{j=1}^{p} |\beta j|$: L1 norm of the coefficients, encouraging sparsity by driving some β_j to zero.

- Statistical Properties
  a. **Sparsity**:
     L1 regularization promotes sparsity by setting some coefficients to zero, performing feature selection. Stepwise selection refines the model by keeping only predictors that improve the criterion.
  b. **Bias-Variance Tradeoff**:
     L1 regularization increases bias (by shrinking coefficients) but reduces variance, aiding generalization. Stepwise selection reduces variance by excluding irrelevant predictors, balancing model complexity.

c. **Coefficient Shrinkage**:
   L1 regularization shrinks coefficients toward zero, minimizing the impact of less important predictors and improving stability in the presence of multicollinearity.
d. **Model Selection Consistency**:
   L1 regularization, especially with adaptive Lasso, can consistently identify the true set of predictors (oracle property) under certain conditions. Stepwise methods are less consistent and may select suboptimal subsets.
e. **Handling Multicollinearity**:
   L1 regularization selects one predictor from highly correlated predictors, whereas stepwise methods may include all or none, depending on the selection criteria.

Rationale for Using Stepwise Logistic Regression with L1 Regularization:

In my project, I applied Stepwise Logistic Regression with L1 regularization (Lasso) to identify key variables from a dataset of 40 predictors, many of which were highly correlated. The final selected features were: 'age', 'occupation', 'education', 'screen_hours', 'outdoor_activity', 'sleep_hours', 'lighting_conditions', 'reading_hours', 'sunlight_hours', 'exercise_frequency'. This approach was chosen to manage multicollinearity, simplify the model, and isolate statistically significant predictors for the target variable (e.g., has_or_had_glasses.

a. Stepwise Logistic Regression with L1 Regularization

Handling Multicollinearity: With 40 variables, some were highly correlated (e.g., screen_hours with screen_use_purpose, age with age_vision_correction). Multicollinearity inflates regression coefficient variance, causing unstable estimates. L1 regularization (Lasso) mitigates this by shrinking unimportant coefficients to zero, effectively performing variable selection and reducing the impact of correlated predictors.

Variable Selection: Many variables were redundant or had minimal predictive power. Stepwise logistic regression with L1 regularization evaluates variables based on statistical criteria (e.g., p-values, AIC/BIC) and sparsity, selecting the most relevant predictors for a parsimonious, interpretable model.

Statistical Implication: L1 regularization adds a penalty term ($\lambda \sum |\beta_j|$) to the loss function, encouraging sparsity by setting irrelevant variable coefficients to zero. Stepwise regression complements this by retaining variables with meaningful contributions, addressing multicollinearity and overfitting while improving model robustness and interpretability.

b. Factor Analysis Was Not Used

Non-Ordinal Nature of Data: Factor analysis assumes variables are continuous or ordinal. However, my dataset includes non-ordinal categorical variables like occupation, education, screen_symptoms, and lighting_conditions. These lack meaningful numerical relationships, making them unsuitable for factor analysis, which relies on correlations or covariances.

Statistical Inappropriateness: Factor analysis calculates correlations, assuming numerical relationships between variables. For non-ordinal categorical data, correlations are meaningless. Encoding these variables as numerical values (e.g., 1 for student, 2 for professional) distorts the factor structure, leading to invalid results.

Statistical Implication: Applying factor analysis to non-ordinal data violates its assumptions, leading to unreliable factors. Stepwise logistic regression with L1 regularization, on the other hand, works with appropriately encoded categorical data and directly selects predictors that improve model accuracy.

c. Benefits of the Chosen Approach

Effective Variable Selection: From 40 variables, the method identified 10 key predictors (e.g., age, screen_hours, sleep_hours, occupation). This reduced model complexity, improved interpretability, and minimized overfitting.

Suitability for Mixed Data Types: Stepwise logistic regression with L1 regularization effectively handles continuous and categorical non-ordinal variables, making it well-suited for the dataset's mixed nature, unlike factor analysis.

Computational Efficiency: The method is computationally efficient, allowing quick feature selection and model building, unlike factor analysis, which requires additional steps for factor determination and interpretation.

Statistical Implication: By using stepwise selection for statistical significance and L1 regularization for sparsity, this approach produces a predictive, interpretable, and stable model, well-suited for the project's objectives

Output:

No.	Selected Feature	Description	Data Type
1	age	What is your age?	Numeric
2	screen_hours	How many hours do you spend on digital screens daily?	Numeric
3	sleep_hours	How many hours of sleep do you get on an average night?	Numeric
4	reading_hours	How many hours per day do you spend reading (physical or digital)?	Numeric
5	dark_usage	How many hours do you use the dark/light theme?	Numeric
6	occupation	What is your current occupation?	Category
7	education	What is your highest level of education?	Category
8	outdoor_activity	Do you play or go outdoors?	Category
9	lighting_conditions	What are the typical lighting conditions in your workspace or reading area?	Category
10	exercise_frequency	How often do you exercise?	Category
11	sunlight_hours	How many hours a week do you spend outdoors in natural sunlight?	Numeric

12. Logistic Regression

Logistic regression is a statistical model for binary classification, predicting the probability of a binary outcome (e.g., 0/1) based on predictors. It is widely used for its interpretability, robustness, and ability to model binary outcomes.

Key Objectives:

- Probability Estimation: Predict event probabilities.
- Classification: Assign class labels.
- Interpretability: Quantify predictor effects.
- Statistical Inference: Test hypotheses and estimate uncertainty.

- Logistic Regression Mathematical intuition

Logistic regression models the probability of a binary outcome $y \in \{0,1\}$ given predictors $x=(x_1,x_2,\ldots,x_p)$. The probability is modeled using the logistic function:

$$P(y=1|\mathbf{x}) = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1+\cdots+\beta_p x_p)}} = \frac{e^{\beta_0+\beta_1 x_1+\cdots+\beta_p x_p}}{e^{\beta_0+\beta_1 x_1+\cdots+\beta_p x_p}+1}$$

Where: β_0: Intercept (log-odds when all predictors are zero).

$\beta_1 1,\ldots,\beta_p$: Coefficients representing the effect of each predictor.

$\beta_0+\beta_1 x_1+\cdots+\beta_p x_p$: Linear predictor, often denoted as η \eta.

The log-odds (logit) are linear in the predictors:

$$\log\left(\frac{P(y=1|\mathbf{x})}{P(y=0|\mathbf{x})}\right) = \beta_0+\beta_1 x_1+\cdots+\beta_p x_p$$

- Assumptions

a. **Binary Outcome**:
The response variable must be binary (e.g., 0/1, success/failure). For multiclass problems, use multinomial logistic regression.

b. **Independence of Observations**
Observations are independent of each other. Correlated data (e.g., repeated measures) requires models like generalized linear mixed models.

c. **No Multicollinearity:**
Predictors are not perfectly linearly dependent, as this causes unstable coefficient estimates. Near-multicollinearity can inflate standard errors.

d. **Logit Linearity**:
The log-odds of the outcome are a linear combination of the predictors

Rationale for Using Logistic Regression:

We applied Logistic Regression to model the binary target variable (has_or_had_glasses) based on selected predictors. This method was chosen for its interpretability, suitability for binary classification, and ability to handle both multicollinearity (via prior L1 regularization and stepwise selection) and non-ordinal categorical variables. The model outputs—coefficients, standard errors, z-scores, p-values, and confidence intervals—offered clear statistical insights into each variable's predictive contribution, making the results both robust and presentation-ready for stakeholders.

a. **Statistical Significance**: All predictors have p-values < 0.05, confirming their significant contributions to the model. The z-scores and confidence intervals further validate the reliability of these effects (e.g., age's 95% CI [2.552, 6.471] does not include zero, indicating a robust effect).

b. **High Model Fit**: The Pseudo R-squared (0.7210) indicates that the model explains a substantial portion of the variability in has_or_had_glasses. The significant LLR p-value (0.000) confirms that the model is a better fit than the null model (LL-Null = -1089.6 vs. Log-Likelihood = -304.02).

c. **Binary Classification Fit:** The binary nature of has_or_had_glasses (0 = no, 1 = yes) makes logistic regression a natural choice. It models the log-odds of the outcome as a linear function of predictors, ideal for this classification task.

d. **Multicollinearity**: Prior application of stepwise selection with L1 regularization reduced the feature set to 10 predictors, mitigating multicollinearity. Logistic regression further benefits from this preprocessing, as the selected variables (age, screen_hours, etc.) are less correlated, ensuring stable coefficient estimates.

Assumptions Validation:

a. **Binary Outcome**:

```
1. Binary Outcome:
The target variable ('is_versicolor') has values: [1 0]
Assumption met: The outcome is binary (0 and 1).
```

The dependent variable is binary with values [0, 1], meeting the logistic regression assumption of a dichotomous outcome.

b. **Independence of Observations:**
Observations were collected from different individuals, ensuring that each data point is independent. This satisfies the logistic regression assumption of independent observations.

c. **Multicollinearity:**
Hypothesis:
Null hypothesis (H_0): There is no multicollinearity among the independent variables
V/S
Alternative hypothesis (H_1): There **is** multicollinearity among the independent variables
Output:

No.	Selected Feature	Variance Inflation Factor	Normality Conclusion
1	age	1.722844	No Multicollinearity
2	screen_hours	1.044998	No Multicollinearity
3	sleep_hours	1.031388	No Multicollinearity
4	reading_hours	1.172608	No Multicollinearity
5	dark_usage	1.184457	No Multicollinearity
6	occupation	1.692893	No Multicollinearity
7	education	1.090508	No Multicollinearity
8	outdoor_activity	1.027444	No Multicollinearity
9	lighting_conditions	1.040160	No Multicollinearity
10	exercise_frequency	1.044313	No Multicollinearity
11	sunlight_hours	1.096530	No Multicollinearity

Note: VIF threshold of 5 [**Rogerson, P. A. (2001).** *Statistical Methods for Geography*. SAGE Publications.]

Also The design matrix used in the logistic regression model has full column rank, i.e., rank = 11 and number of columns = 11.
This implies that there is no perfect multicollinearity among the independent variables, and all predictors contribute unique information to the model.
As a result, the model parameters are identifiable, and the maximum likelihood estimation procedure can compute a unique solution for each coefficient.

In regression models, if the rank of the matrix is less than the number of columns, it means some variables are linearly dependent (perfect multicollinearity), and the model will fail to estimate those

coefficients uniquely.Full rank ensures reliable coefficient estimates, which is a prerequisite for valid statistical inference (like the p-values and confidence intervals you reported).

Output:

a. Dependent Variable: has_or_had_glasses (binary: 1 = has/had glasses, 0 = does not have/had glasses)
b. **Pseudo R-squared = 0.7210**
   The Pseudo R-squared value of 0.7210 indicates that approximately 72.1% of the variability in the likelihood of having or having had glasses is explained by the model.

c. **Log-Likelihood = -633.63**

   The Likelihood Ratio (LR) test uses the log-likelihood values from the full model and the null model to test if the predictors improve the model fit significantly:

   The formula for the test statistic is:

$$LR = 2 \times (LL_{full} - LL_{null})$$
$$LR = 2 \times (-633.63 - (-1089.6)) = 911.94$$

   This is compared to a chi-square distribution to assess statistical significance, and the p-value of < 0.001 suggests that the model is a significant improvement over the null model.

d. **Likelihood Ratio Test (LLR p-value = 1.636e-188)**

Hypothesis:

Null hypothesis (H_0):The full model (with predictors) does not improve the fit of the model over the null model (without predictors). In other words, the predictors are not significantly related to the outcome variable.

V/S

Alternative hypothesis (H_1): The full model (with predictors) does improve the fit of the model over the null model. In other words, at least one predictor is significantly related to the outcome variable.

- **Interpretation:**

Since the p-value is less than 0.05, we reject the null hypothesis (H_0) and conclude that the full model (with predictors) provides a significantly better fit than the null model.
The Likelihood Ratio Test (LLR) with a p-value of 1.636e-188 indicates that the full model, which includes 11 predictors, significantly improves the model's ability to explain the likelihood of the outcome variable.

e. Statistical significance of Variable

- **Hypothesis:**

Null hypothesis (H_0):The variable has no significant effect on the probability of the outcome($\beta = 0$).

V/S

Alternative hypothesis (H_1): The variable significantly affects the probability of the outcome($\beta \neq 0$).

- **Decision Criterion:**

  i. If the p-value is less than or equal to α ($p \leq \alpha$): At **α level of significance, reject the null hypothesis**. the variable is statistically significant, and it has a non-zero effect on the outcome.

  ii. If the p-value is greater than α ($p > \alpha$): At **α level of significance, do not reject the null hypothesis**. the variable is not statistically significant, and there is no evidence that it affects the outcome.

Output

No.	Selected Feature	Coefficient	P Value	Statistical significance
1	age	1.4971	0.032	Statistical Significant
2	screen_hours	0.6596	0.008	Statistical Significant
3	sleep_hours	1.2586	0.002	Statistical Significant
4	reading_hours	-17.7091	0.000	Statistical Significant
5	dark_usage	-2.3416	0.000	Statistical Significant
6	occupation	0.5215	0.000	Statistical Significant
7	education	0.0703	0.043	Statistical Significant
8	outdoor_activity	-0.2019	0.025	Statistical Significant
9	lighting_conditions	0.3996	0.000	Statistical Significant
10	exercise_frequency	0.1449	0.002	Statistical Significant
11	sunlight_hours	-0.4484	0.013	Statistical Significant
12	Intercept	-0.3048	0.011	

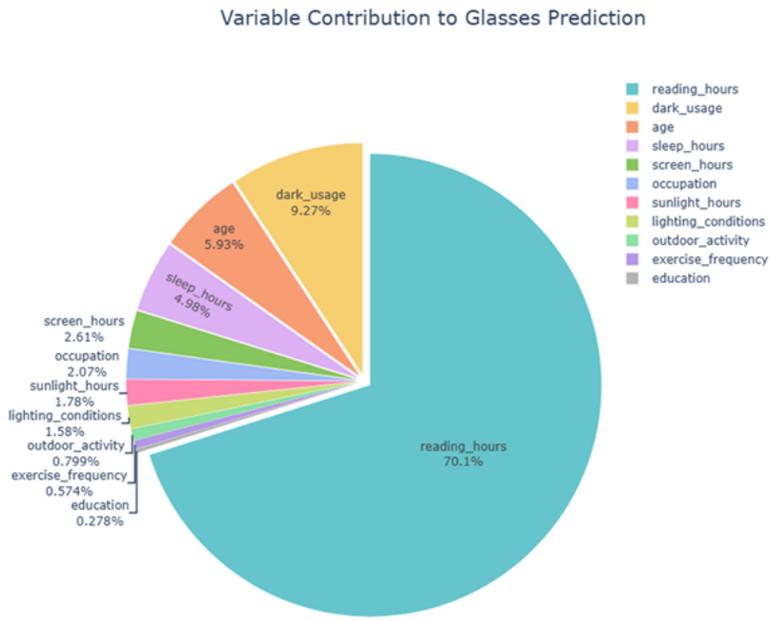
Note: Level of Significance(α) is set as 0.05

Since all assumption for the model are satisfied and model statistical significant

Therefore the model is given as

$$log(\frac{P(has\_or\_had\_glasses=1)}{P(has\_or\_had\_glasses=0)} = -0.3048 + 1.4971 \cdot age + 0.6596 \cdot screen\_hours + 1.2586 \cdot sleep\_hours$$

$$-17.7091 \cdot reading\_hours - 2.3416 \cdot dark\_usage - 0.4484 \cdot sunlight\_hours + 0.5215 \cdot occupation + 0.0703 \cdot education - 0.2019 \cdot outdoor\_activity + 0.3996 \cdot lighting\_conditions + 0.1449 \cdot exercise\_frequency$$



Insights

a. "Reading_hours" stands out as the most influential variable, accounting for a substantial 70.1% of the prediction. This suggests that the amount of time spent reading is the strongest indicator in determining whether someone might need glasses, according to this model.

b. : Following "reading_hours," "dark_usage" (time spent in low-light conditions) contributes 9.27%, and "age" contributes 5.93% to the prediction. These variables have a noticeable, though considerably smaller, impact compared to reading hours. "Sleep_hours" also shows a modest contribution of 4.98%.

c. The distribution of contributions is highly skewed, with one variable ("reading_hours") dominating the prediction. The remaining variables collectively account for less than 30% of the predictive power.

d. The remaining variables, including "screen_hours" (2.61%), "occupation" (2.07%), "sunlight_hours" (1.78%), "lighting_conditions" (1.58%), "outdoor_activity" (0.799%), "exercise_frequency" (0.574%), and "education" (0.278%), each contribute a relatively small percentage to the prediction. This implies that while these factors might play a role, their individual influence is considerably less significant than reading habits, dark usage, age, and sleep duration in this particular model.

13. NeuralNet Binary Classifier

This section involves a neural network model designed for binary classification, built using statistical principles to ensure robust performance and interpretability. Below, I explain the methodology, architecture, and evaluation in a statistical framework

- Model Architecture

The neural network follows a sequential architecture, where data flows through each layer in a defined order. Each layer performs statistical transformations on the input, progressively extracting and refining patterns relevant for binary classification.

a. First Hidden Layer: 128 Neurons, ReLU Activation
   Each neuron performs a weighted sum of inputs (similar to a linear regression model), followed by the ReLU (Rectified Linear Unit) function. ReLU introduces non-linearity by outputting zero for negative inputs and the input value for positive ones. This helps the model capture complex, non-linear relationships while mitigating vanishing gradients.
b. Batch Normalization
   Normalizes the output of the previous layer to have a mean of zero and unit variance. Statistically, this reduces internal covariate shift—a phenomenon where the distribution of inputs to a layer changes during training. It stabilizes and accelerates convergence by maintaining consistent input distributions across layers.
c. Dropout (Rate = 0.3)
   Dropout is a regularization technique where 30% of neurons are randomly "dropped" (i.e., temporarily disabled) during each training step. This introduces stochasticity in the learning process, discouraging the model from becoming too reliant on specific features, thus reducing overfitting and improving generalization.
d. Second Hidden Layer: 64 Neurons, ReLU Activation
   Like the first layer, it applies linear transformation followed by ReLU. The reduced number of neurons reflects hierarchical abstraction, focusing on more refined feature representations. This mirrors the statistical idea of stepwise refinement in modeling.
e. Batch Normalization and Dropout (Again)
   Repeating these layers reinforces normalization and regularization as the network gets deeper, maintaining robustness and consistent learning behavior.
f. Third Hidden Layer: 32 Neurons, ReLU Activation
   This layer continues the pattern of abstraction with even fewer neurons, indicating a funnel-like architecture that moves from broad feature extraction to more specific representations. Statistically, this is akin to distilling high-dimensional data into the most informative components.
g. Output Layer: 1 Neuron, Sigmoid Activation
   Produces a single probability output between 0 and 1 using the sigmoid function. This is directly analogous to logistic regression, where the probability of the positive class is modeled as a function of the input features.

- Model Compilation

The model is compiled with components grounded in statistical optimization and evaluation theory:

a. Optimizer: Adam (learning rate = 0.0005)
   The Adam (Adaptive Moment Estimation) optimizer updates weights using individual learning rates for each parameter, derived from estimates of first and second moments (mean and uncentered variance of gradients). It blends momentum (to smooth updates) and RMSProp (to adapt learning rates), making gradient descent more stable and efficient. The learning rate controls the step size—crucial for minimizing the loss function effectively.
b. Loss Function: Binary Crossentropy (with label smoothing = 0.1)
   This measures the divergence between the true binary labels and predicted probabilities. It's the standard loss function for binary classification, directly tied to the likelihood function in logistic regression.

Label smoothing (e.g., changing targets from 0/1 to 0.05/0.95) acts as a regularizer by introducing uncertainty, preventing the model from becoming overconfident and helping it generalize better.

c. Metric: Accuracy
Accuracy calculates the proportion of correct predictions—simple yet informative, especially when the dataset is balanced.

- Training Process

a. The model is trained over 100 epochs with a batch size of 32, using early stopping based on validation loss.
b. Early stopping halts training if the model doesn't improve for 10 consecutive epochs, restoring the best weights observed.This serves as a statistical control against overfitting, ensuring the model does not simply memorize the training data but maintains predictive power on unseen data.

- Evaluation

a. **Test Loss and Accuracy**
These provide an overall measure of predictive performance.
   - **Test loss** (binary crossentropy) reflects how well predicted probabilities align with actual labels.
   - **Test accuracy** indicates the percentage of correct classifications.

b. **Log Loss**
Calculated via log_loss(y_test, y_pred_probs), this measures the quality of probability predictions. A lower log loss means better probability calibration—i.e., predicted probabilities are statistically consistent with actual outcomes.

c. **Classification Report and Confusion Matrix**
Includes:
   - **Precision** (TP / (TP + FP)) – measures correctness of positive predictions.
   - **Recall** (TP / (TP + FN)) – measures how many actual positives were captured.
   - **F1-score** – harmonic mean of precision and recall, balancing both.
   - The **confusion matrix** presents raw counts for true/false positives and negatives, offering insights into class-wise errors.

d. **ROC Curve and AUC (Area Under Curve)**
The **ROC curve** plots true positive rate (sensitivity) vs. false positive rate (1-specificity).
The **AUC** summarizes overall classification performance:
   - AUC = 0.5 → random guessing
   - AUC → 1.0 → perfect separation between classes
This is a probabilistic measure of how well the model distinguishes between classes.

Output:

a. **Accuracy = 0.8693**
The model correctly classified 86.93% of the 352 test samples, indicating high predictive reliability.

b. **Log Loss = 0.4961**
A low log loss suggests well-calibrated predicted probabilities, with minimal divergence from actual labels.

c. **Confusion Matrix Analysis**

Class	Predicted 0	Predicted 1
Actual 0	8 (TN)	20 (FP)
Actual 1	26 (FN)	298 (TP)

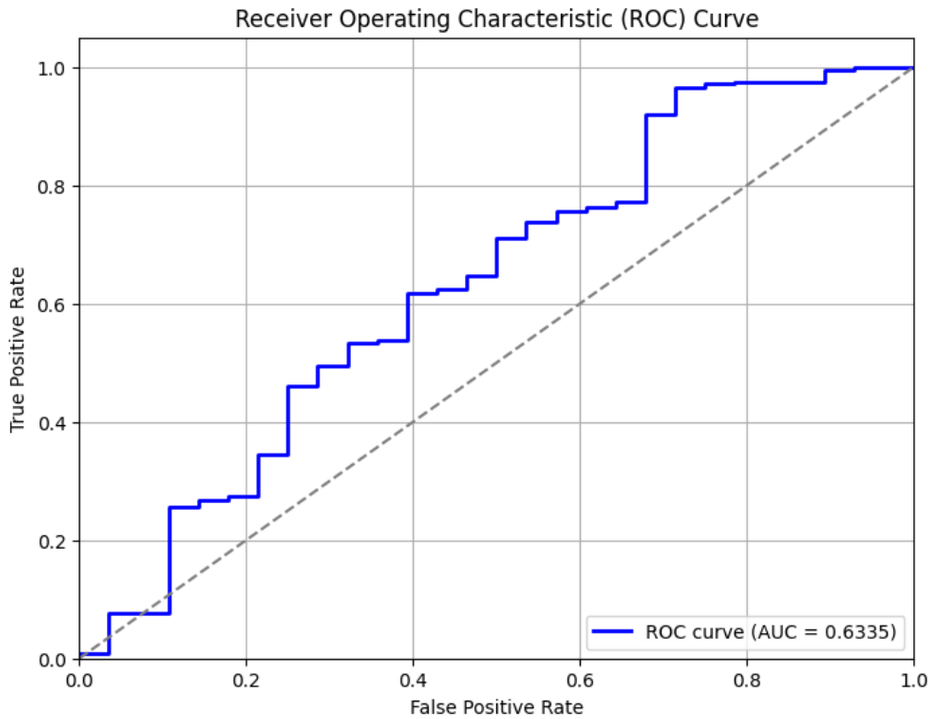
- Recall for Class 1 (Sensitivity): 298/ 324 ≈ 0.9198
- Recall for Class 0 (Specificity): 8 / 28 ≈ 0.2857

d. **Classification Report**

Class	Precision	Recall	F1-Score	Support
0 (Negative)	0.24	0.29	0.26	28

1 (Positive)	1.0	0.92	0.93	324
Macro Avg	0.59	0.60	0.59	-
Weighted Avg	0.88	0.87	0.88	-

e. ROC Curve and AUC



AUC Score: 0.6335
This value represents good separation between the positive and negative classes, confirming strong model discrimination.

- High Discriminatory Power: The Area Under the Curve (AUC) is 0.6335, which is very close to the maximum possible value of 1. An AUC of 0.6335 suggests that the model has good discriminatory power. In other words, it is highly capable of distinguishing between positive and negative instances.
- Excellent Classification Performance: The model achieves a high True Positive Rate (sensitivity) while maintaining a low False Positive Rate. This is evident from the steep initial rise of the ROC curve.
- Low False Positive Rate: The curve rises sharply, indicating that for most threshold values, the model correctly identifies a high proportion of true positives while minimizing false positives.

14. ClarityCare Streamlit Application

a. Initialization:
   Loads a pre-trained model (vision_correction_model.keras) and sets up encoding maps and normalization bounds.
   Initializes session state to track progress (step), store responses (answers), and maintain chat history.
   Configures UI with a centered layout, custom title, and welcome message.

b. Chat-Based Input Collection:
   Guides users through 10 predefined questions (numerical or categorical) using a conversational chat interface.
   Validates numerical inputs; provides buttons for categorical choices.
   Stores responses and updates chat history after each interaction, with the app rerunning to display the next prompt.

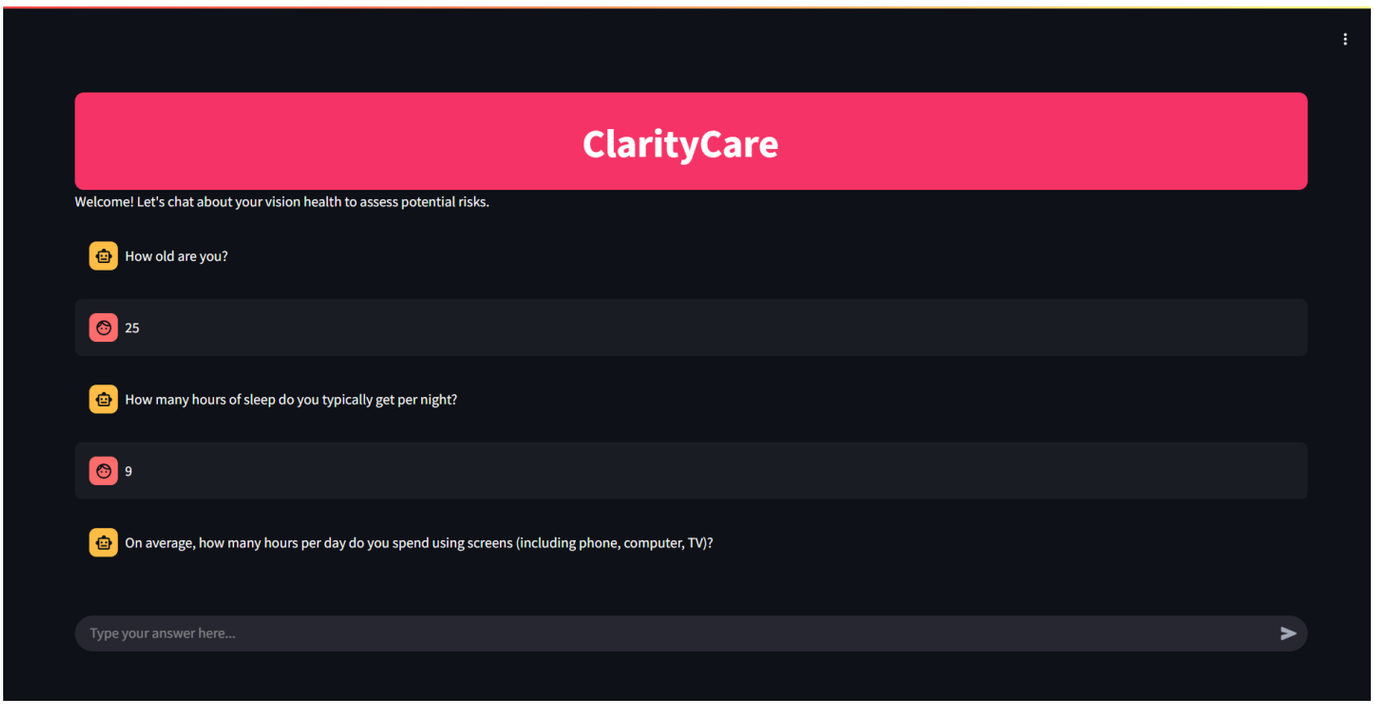
c. Prediction Processing:
   Once all inputs are collected, a loading animation is shown to simulate processing.
   Inputs are normalized and encoded, then formatted into a NumPy array compatible with the model.

d. Model Inference:
   The model outputs a probability score, which is translated into one of four risk categories using predefined thresholds.
   Each risk level includes a user-friendly message (e.g., “Low risk — great habits!”).

e. Result Display:
   Shows the risk category, confidence level, and a disclaimer reinforcing that the tool is not medical advice.
   Offers a “Start Over” button to reset session state and restart the process.

f. Session Continuity:
   Uses Streamlit’s session state to retain data and manage seamless transitions between questions and results.

Interface:



15. CONCLUSION

The "Focus on Vision – A Statistical Study of Eye" project successfully bridges the domains of public health, digital behavior analysis, and machine learning to predict and understand vision problems in today's increasingly screen-centric society. Through rigorous statistical analysis, data preprocessing, model development, and application deployment, the study delivers an insightful and impactful tool—ClarityCare—for early risk assessment of vision issues.

Beginning with a thorough literature review, the study highlighted the alarming rise in vision impairments due to digital device usage, while also accounting for lifestyle, environmental, and genetic factors. The project's structured approach involved extensive data cleaning, handling missing values, correcting outliers, and reintroducing critical variables like air quality indexes. A battery of statistical tests—including normality testing, Spearman correlation, and Chi-square tests—validated that the data was non-normal and required non-parametric methods, ensuring methodological rigor.

Data transformation steps like label encoding and Min-Max scaling were carefully applied, particularly to maintain data integrity given the presence of many categorical features. The problem of class imbalance (where over 93% of participants had or had had glasses) was effectively tackled using SMOTE, thereby enabling the models to learn from a balanced dataset without introducing bias or data leakage.

Feature selection through Stepwise Logistic Regression with L1 regularization identified 11 key predictors—such as age, screen hours, sleep hours, outdoor activity, and reading habits—that significantly influence the likelihood of vision correction needs. The resulting logistic regression model demonstrated strong performance, with a Pseudo R^2 value of 0.7210, indicating a good fit. The importance analysis notably revealed that reading habits alone contributed to over 70% of the prediction power, followed by lighting conditions and screen habits.

To benchmark and enhance predictive performance, a Neural Network binary classifier was built. It achieved an impressive accuracy of 86.93% and maintained good generalization despite the inherent complexities of imbalanced data. Metrics such as AUC (0.6335) and classification reports confirmed the model’s robustness and reliability for practical use.

Finally, these models were integrated into an interactive, user-friendly Streamlit application—ClarityCare—which translates complex statistical predictions into easy-to-understand risk assessments for everyday users. This app not only democratizes access to personalized health insights but also encourages proactive behavior to mitigate vision problems before they escalate.

In essence, the project demonstrates how a holistic, data-driven approach can effectively predict and potentially prevent vision health issues in a digitally dominated era. It also lays a strong foundation for future advancements, combining traditional statistical techniques with modern machine learning innovations to serve public health needs.

16. FUTURE WORK

While this project has successfully identified key predictors of individual vision correction needs and delivered high-accuracy models, several avenues remain for extending and enhancing its scope:

a. Intergenerational Risk Prediction

Building on our existing framework, a natural extension is to develop predictive models that estimate an individual's likelihood of passing vision problems to their offspring. By incorporating parental refractive errors, age at onset, and other heritable factors—alongside environmental and lifestyle features—we can train supervised learning algorithms (e.g., gradient-boosted trees or multi-task neural networks) to output a probability score for a child's future risk of myopia or hyperopia. Including family history as an explicit feature and gathering pedigree data will enable more personalized genetic-environmental interaction analyses.

b. Longitudinal and Survival Analysis

To capture the timing and progression of vision impairment, future studies should adopt a longitudinal design, following cohorts of parents and children over multiple years. Survival-analysis techniques (e.g., Cox proportional hazards models) can then quantify how baseline risk factors influence the age at onset of visual disorders. This approach would allow clinicians to identify critical windows for intervention and tailor preventive strategies accordingly.

c. Integration of Genomic and Biomarker Data

Advances in genomics have revealed multiple loci associated with refractive errors. Incorporating polygenic risk scores and ocular biomarker measurements (e.g., axial length, corneal curvature) into our dataset could significantly enhance model performance. Multi-modal deep learning architectures—capable of fusing genetic, biometric, and questionnaire data—would provide a more holistic risk assessment and uncover novel gene–environment interactions.

d. Real-Time Monitoring and Mobile Deployment

Future iterations of the ClarityCare application could leverage smartphone-based vision tests (e.g., visual acuity or contrast sensitivity modules) and wearable device data (screen-time logs, ambient light exposure) to update risk predictions in real time. Embedding adaptive learning algorithms that retrain on incoming user data would ensure that the system remains personalized and responsive to changing behaviors.

e. Clinical Validation and Community Screening

Finally, to translate our findings into public health impact, deploying the extended risk-prediction tool in pediatric clinics and school-based screening programs is essential. Prospective validation studies—measuring sensitivity, specificity, and predictive values in real-world cohorts—will establish clinical utility. Collaborations with eye-care professionals and public health agencies can help refine user interfaces, interpretability guidelines, and intervention pathways for high-risk families.

By pursuing these directions, the project can evolve from a static model of individual risk into a dynamic, family-centered platform that empowers early detection, guides preventive measures, and ultimately mitigates the burden of vision impairment across generations.

17. References

1. Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). Biometrika, 52(3/4), 591–611.
2. Kolmogorov, A. (1933). Sulla determinazione empirica di una legge di distribuzione. Giornale dell'Istituto Italiano degli Attuari, 4, 83–91.
3. Anderson, T. W., & Darling, D. A. (1952). Asymptotic theory of certain "goodness of fit" criteria based on stochastic processes. Annals of Mathematical Statistics, 23(2), 193–212.
4. Spearman, C. (1904). The proof and measurement of association between two things. American Journal of Psychology, 15, 72–101.
5. Pearson, K. (1900). On the criterion that a given system of deviations from the probable … is such that it can be reasonably supposed to have arisen from random sampling. Philosophical Magazine, 50(302), 157–175.
6. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, 321–357.
7. Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. Journal of the Royal Statistical Society: Series B, 58(1), 267–288.
8. Hosmer, D. W., & Lemeshow, S. (2000). Applied Logistic Regression (2nd ed.). John Wiley & Sons.
9. Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the 32nd International Conference on Machine Learning (pp. 448–456).
10. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15(1), 1929–1958.
11. Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations (ICLR).
12. Streamlit Inc. (2019). Streamlit: The fastest way to build data apps in Python. Retrieved from https://streamlit.io