IronHack Data Analytics

WEEK 7 | DRY EYES MACHINE LEARNING

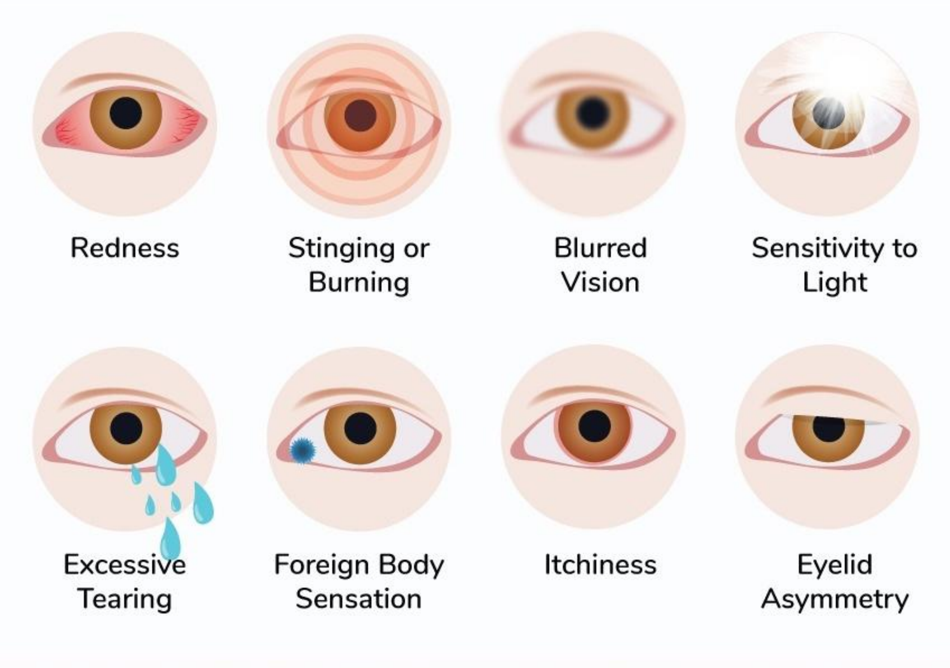
Agenda

- Project Overview: “The Dry Eye Syndrome”
- Data Selection
- Feature Engineering & Selection
- Model Building & Evaluation
- Hyperparameter Tuning & Model Optimization
- Key Findings & Insights
- Real World Implication & Impact
- Challenges & Learnings
- Future Work & Improvements

Project Overview: The Dry Eye Syndrome (DES)

- **Common condition = Eyes do not produce enough tears** or the tears evaporate too quickly
- **Causes:** Environmental factors, aging, medical conditions, screen time (!!)
- **Treatment & Prevention:** Artificial tears & eye drops, using using humidifiers, wearing blue-light filters

→ **As rising Data Analysts our heavy screen time puts us at risk of developing DES and we want to prevent that!**



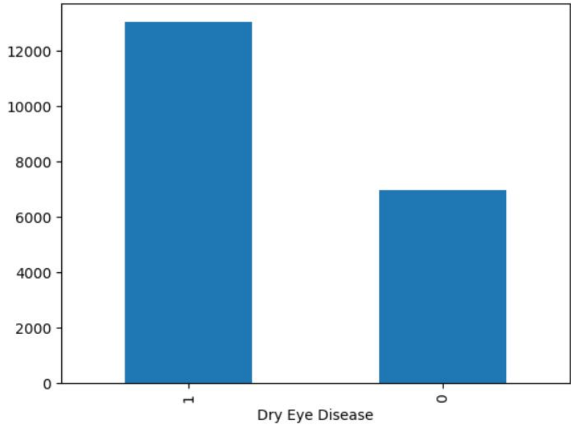
Data Selection

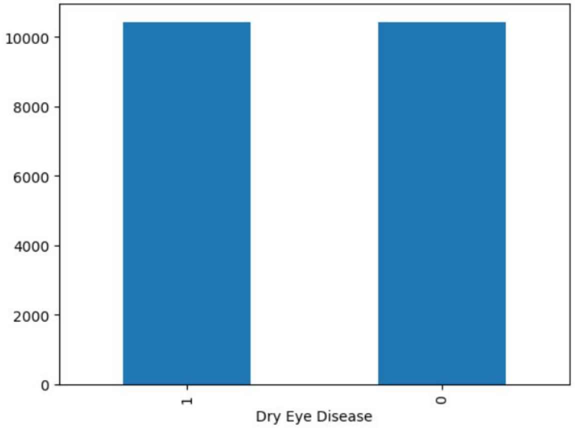
Dry Eyes

preprocessed_dry_eye_dataset

- Key Columns:
    - **Sleep Duration**
    - **Screen Time**
    - **Height/Weight (BMI**
    - **Age / Gender**
    - **…**
- Lists various factors that might contribute to DES

Oversampling





Feature Engineering

One Hot Encoding

Categorical columns

- Age (young adults, adults)
- Gender
- Sleep disorder
- Week up during night
- Feel sleepy during day
- Alcohol consumption
- Smoking

MinMax Scaler

Numerical columns

- Sleep duration
- Sleep quality
- Stress level
- Heart rate
- Daily steps
- Physical activity
- BMI

Model Building and Evaluation

- **Oversampling needed due to heavy data imbalance!**

Model Selection:

- **Logistic Regression Model**
  - Low accuracy (0.55)

Model Accuracy: 0.5520

Classification Report:

	precision	recall	f1-score	support
0	0.40	0.55	0.46	1397
1	0.70	0.55	0.62	2603
accuracy			0.55	4000
macro avg	0.55	0.55	0.54	4000
weighted avg	0.59	0.55	0.56	4000

- **Decision Tree**
  - Greater accuracy (0.70)

→ But prone to overfitting!

Accuracy score: 0.70

	precision	recall	f1-score	support
0	0.60	0.21	0.31	1307
1	0.71	0.93	0.80	2693
accuracy			0.70	4000
macro avg	0.65	0.57	0.56	4000
weighted avg	0.67	0.70	0.64	4000

Model Building and Evaluation

Model Selection:

- **Random Forest Classifier**
  - **Accuracy (0.69 → 0.70)**

→ Similar performance to decision tree

- **KNN**
  - Low **Accuracy (0.54)**

→ Also computationally expensive!

Model Accuracy: 0.6993

	precision	recall	f1-score	support
0	0.61	0.21	0.32	1307
1	0.71	0.93	0.81	2693
accuracy			0.70	4000
macro avg	0.66	0.57	0.56	4000
weighted avg	0.68	0.70	0.65	4000

Model Accuracy: 0.5445

Classification Report:

	precision	recall	f1-score	support
0	0.37	0.45	0.41	1397
1	0.67	0.59	0.63	2603
accuracy			0.54	4000
macro avg	0.52	0.52	0.52	4000
weighted avg	0.57	0.54	0.55	4000

Hyperparameter Tuning & Model Optimization

- Initial **Random Forest model** achieved **69.9% accuracy**, but we wanted to optimize it for better performance

Key hyperparameters tuned:

- `n_estimators`: Number of trees in the forest
- `max_depth`: Maximum depth of each tree
- `min_samples_split`: Minimum samples required to split a node
- `min_samples_leaf`: Minimum samples required at a leaf node
- `max_features`: Number of features considered for splitting

Optimized Random Forest Classifier:

- `n_estimators`: **386**
- `max_depth`: **36**
- `min_samples_split`: **13**
- `min_samples_leaf`: **20**
- `max_features`: **'log2'**

Result: Accuracy improved **slightly** from **69.9%** → **70.2%**.

Challenges & Learning

- **Data Preprocessing Uncertainty**: Mistakes were made due to confusion on whether to **normalize & one-hot encode before or after splitting** the data
- **Model Selection Confusion**: Various models were tested, and selecting the best-performing one required experimentation
- **Hyperparameter Tuning had limited Impact**: Despite optimizing the **Random Forest model**, the improvement was minor (~69.9% → 70.2% accuracy)
- **Imbalanced Dataset Required Oversampling**: Without balancing, models performed poorly in predicting **DES**

Key Findings & Insights

- **Best Performing Model**: **Random Forest Classifier** achieved the highest accuracy (**70.2%**) after hyperparameter tuning
- **Feature Importance Analysis**:
    - **Screen Time & Sleep Duration** had strong correlations with Dry Eye Disease
- **Age & Gender had limited predictive power** in our dataset
- **Logistic Models Underperformed**: **Logistic Regression** failed to capture complex interactions
- **Decision Trees** showed decent results but **Random Forest had the best balance of accuracy & robustness**

Future Work & Improvements

- **Preventing Data Leakage**: Ensure **normalization & encoding happen after the train-test split** to avoid unrealistic performance metrics
- **Feature Selection Optimization**: Use **correlation matrix & feature importance scores** to refine the most relevant predictors
- **Experiment with More Advanced Models**: E.g.**XGBoost, LightGBM, or Neural Networks** for potential performance gains
- **Collect More Data**: Expanding the dataset could improve model generalizability and performance

PROJECT Dry Eye Syndrome

THANKS !

Angel Vargas| Elohor Avwarute | Lukas Günther | Rebecca Woo