Eye-tracking data to evaluate people's empathy

April 23, 2023

Registration number: 2201524
Link to GitHub: https://github.com/dm22227/Data-decision-Assignment2-.git

Executive summary (max. 200 words)	185
Main findings and Discussion (max. 600 words)	1411
Conclusions (max. 300 words)	210
Total word count	1806

Contents

1 Main Findings **2**

2 Discussion **2**

2.1 Fundamental of Dataset 2

2.2 Analysis of Assessment 2

3 Conclusions **5**

Abstract

In this paper, we explore dataset where the amount that eye pupil diameter, measured via an eye tracking equipment, predicts levels of empathy among people when they are performing a particular activity. Using Eye Tracking dataset, we first examined the data to determine which factor was crucial like compute features are used based on pupil dilation, eye position, and gazing to prediction of empathy assessment. We first separate the data into datasets II and III. On both halves of the dataset, we apply the same operation. With the aid of significant facture selection based on the highest statistic value, we performed the description information of the data, cleaned the data, and processed the data during the operation. By dividing each dataset into a training and testing portion, we can use the k-mean model to get the empathy score. The more severe the symptoms of depression were, the less dilated the pupils would become. This can be used to analyze people based on empathy. So, in this case, we can determine which variable is more relevant to the prediction of empathy assessment also the model score.

1 Main Findings

Humans' levels of empathy are measured through empathy tests in three areas: social interaction, cognitive behaviour, and emotional identification [7]. Empathy may be measured using self-report tools or scales and psychological theories that have been observed. At the University of Cambridge's ARC (the Autism Research Centre), Simon Baron-Cohen honed his empathic skills [1].

Empathy assessments can be used in various settings such as mental health assessments, research, educational, and interventional purposes. Analysing eye tracker data may be used to evaluate certain brain functions, including empathy. Building strong connections requires empathy. It may be quite significant in some social situations, such as a doctor-patient or caregiver-patient interaction. In order for the doctor or carer to comprehend the patient's issue and, thus, provide her/him the best treatment possible, the patient's tale must be told in the context of trust, which must be established first. According to the study, eye tracker data might offer some promising candidates for indications to judge both empathy and focus. If a link between empathy and attention can be demonstrated, it may be feasible to evaluate empathy indirectly. Another study used eye-gaze patterns and behaviors during immersive to predict the empathy dimension level of participants [3].

In this paper, We are going to used eye tracker data to assess human empathy and attention and severe the symptoms of depression were, the less dilated the pupils would become. this data can be used to analyze depression based on empathy. Some research related eye tracker as below:

Eye tracking is a new technique for capturing eye movement and gaze position across time. [2]. A user's biometric identification, mental activity, personality traits, ethnic background, skills and talents, age and gender, personal preferences, emotional state, level of tiredness and drunkenness, and physical and mental health condition may all be revealed by eye tracking data. [5]. By concentrating on particular characteristics of people or animals in a psychological study, the quality of eye tracking data may be evaluated. Developmental studies can employ eye tracking data to reveal how a youngster judges facial emotions and their capacity to predict events. [4].

2 Discussion

2.1 Fundamental of Dataset

Dataset of eye movements were recorded, as well as their feelings of sympathy for those who had movement disorders. Participants were split into two groups for each round of gaze recording, with each group performing a single task. Another group engaged in gaze typing, which involves composing words on a piece of card stock using eye-gaze motions, while the first group engaged in free exploration of structureless visuals. Two files contain the eye-tracking information gathered from the two activities, together with measurements of the pupils' diameter and gaze location [6].

2.2 Analysis of Assessment

The given code creates an empty Pandas DataFrame called dataset1 and then reads multiple CSV files using pd.read csv() and appends the data to dataset1 using dataset1.append(). The CSV files are read using a loop that iterates over a list of file paths obtained using glob.glob(). The glob.glob() function returns a list of file paths that match a specified pattern. In this case, the

pattern is "EyeT group dataset II "+"*.csv", which matches all CSV files in the directory dir path that start with the string "EyeT group dataset II".

Dataset 1 : EyeT group dataset II

```
# Create an empty DataFrame to store the data
csv_files_II = glob.glob(os.path.join(dir_path, "EyeT_group_dataset_II_"+"*.csv"))

dataset1 = pd.DataFrame()

# loop over the list of csv files
for file in csv_files_II:
  # read the csv file
  temp_df = pd.read_csv(file, header=10, nrows=3) # 3 sample per file; total 502 files; total of 1506 samples
  dataset1 = dataset1.append(temp_df, ignore_index=True)

dataset1.head()
```

Figure 1: Load and Read Datset

The pd.read csv() function reads a CSV file and returns a Pandas DataFrame. In this case, the function is called with the arguments header=10 and nrows=3, which means that only the first 3 rows of the CSV file are read and the first 10 rows are skipped. The resulting DataFrame temp df is then appended to dataset1 using dataset1.append(). The ignore index=True argument is used to reset the index of the resulting DataFrame. The resulting DataFrame dataset1 contains the data from all the CSV files in the directory dir path that match the specified pattern. The head() method is used to display the first few rows of the DataFrame. This same we have to perform with dataset III and also represent the main dataset series histogram.

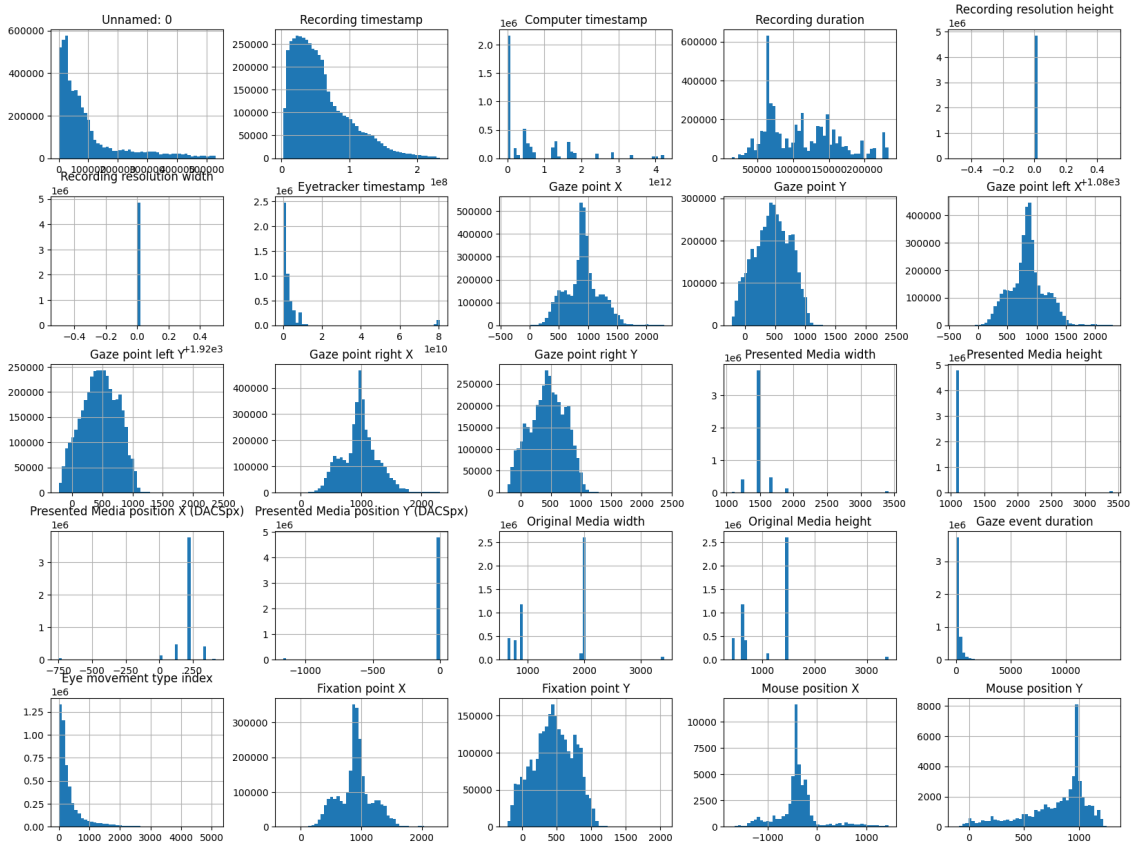

Figure 2: Load and Read Datset

The more severe the symptoms of depression were, the less dilated the pupils would become. this data can be used to analyze depression based on empathy. Base on this theory, we select the variable which you can show in this below figure 2.

The given code assigns column names to the DataFrame dataset1 using the dataset1.columns attribute. The column names are provided as a list of strings in the figure 3:

This code is useful when we want to assign meaningful names to the columns of a DataFrame. The columns attribute is used to get or set the column names of a DataFrame. In this case, the

```
[14] dataset1 = dataset1.iloc[:, [33,34,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52]] # compute features are used bsed on pupil dilation, eye position, and gazing

[15] dataset1.columns = ['P_dia_left', 'P_dia_right', 'Eyepos1','Eyepos2','Eyepos3','Eyepos4','Eyepos5', 'Gazefeature1','Gazefeature2',
                         'Gazefeature3','Gazefeature4','Gazefeature5','Gazefeature6','Gazefeature7','Gazefeature8','Gazefeature9','Gazefeature10',]

[16] dataset1.head()
```

	P_dia_left	P_dia_right	Eyepos1	Eyepos2	Eyepos3	Eyepos4	Eyepos5	Gazefeature1	Gazefeature2	Gazefeature3	Gazefeature4	Gazefeature5	Gazefeature6	Gazefe
0	-0,91856	3,45	199,8	14,1	701,6	266,4	9,8	709,3	427,0	273,9	481,1	226,7	0,9193	
1	-0,91798	NaN	200,0	14,4	701,2	266,6	9,9	709,2	436,0	262,4	488,1	221,8	0,9368	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	NaN	3,03	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

Figure 3: Feature selection

attribute is set to a list of strings that correspond to the column names of dataset1. The order of the strings in the list corresponds to the order of the columns in the DataFrame

The given figure 4 code performs data cleaning and handling on the DataFrame dataset1. The first line replaces all commas in the DataFrame with periods using the replace() method with the argument regex=True. The second line fills all missing values in the DataFrame with a specified value using the fillna() method with the argument value. The resulting DataFrame is displayed using the head() method.

```
[18] value = {'P_dia_left': -.9, 'P_dia_right': 2, 'Eyepos1': 190, 'Eyepos2': 45, 'Eyepos3': 600, 'Eyepos4': 230, 'Eyepos5': 40,
              'Gazefeature1': 740, 'Gazefeature2': 240,'Gazefeature3': 80,'Gazefeature4': 240,'Gazefeature5': 90,'Gazefeature6': .47,
              'Gazefeature7': 0.3,'Gazefeature8': 0.45,'Gazefeature9': 0.28,'Gazefeature10': 0.5}  ## Created value near by the top value from describe function

[19] #fill missing values and data handling
     dataset1 = dataset1.replace(',','.',regex=True)
     dataset1 = dataset1.fillna(value)
     dataset1.head()
```

	P_dia_left	P_dia_right	Eyepos1	Eyepos2	Eyepos3	Eyepos4	Eyepos5	Gazefeature1	Gazefeature2	Gazefeature3	Gazefeature4	Gazefeature5	Gazefeature6	Gazefe
0	-0.91856	3.45	199.8	14.1	701.6	266.4	9.8	709.3	427.0	273.9	481.1	226.7	0.9193	
1	-0.91798	2	200.0	14.4	701.2	266.6	9.9	709.2	436.0	262.4	488.1	221.8	0.9368	
2	-0.9	2	190	45	600	230	40	740	240	80	240	90	0.47	
3	-0.9	2	190	45	600	230	40	740	240	80	240	90	0.47	
4	-0.9	3.03	190	45	600	230	40	740	240	80	240	90	0.47	

Activate Windows

Figure 4: Feature selection

Data cleaning is an important step in data analysis that involves identifying and correcting or removing errors, inconsistencies, and missing values in the data. In this case, the replace() method is used to replace all commas in the DataFrame with periods. This is useful when the data contains numeric values with decimal points that are represented using commas instead of periods. The fillna() method is used to fill all missing values in the DataFrame with a specified value using diecribe top value. This is useful when the missing values need to be replaced with a specific value for further analysis.

```
[21] from sklearn.cluster import KMeans

[22] #  total samples = 426,
     # training = 2/3(426) = 284,
     # testing = 426-284 = 142
     # 1/3 samples used for training
     kmeans_1 = KMeans(n_clusters=2, random_state=0, n_init="auto").fit(dataset1.iloc[:284])

[23] kmeans_1.labels_

array([0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0,
       1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 1, 1, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],

Activate Windows
Go to Settings to activate Windows.
```

Figure 5: Feature selection

The given figure 5 code splits the data into training and testing sets using a 2/3 - 1/3 split. The

total number of samples is 426, and 2/3 of the samples are used for training, which is 284. The remaining 1/3 of the samples are used for testing, which is 142. Splitting the data into training and testing sets is an important step in machine learning. The purpose of this step is to evaluate the performance of the model on data that it has not seen during training. The most common way to split the data is to use a 2/3 - 1/3 split, where 2/3 of the data is used for training and 1/3 of the data is used for testing. This split is used in the given code. The iloc method is used to select the first 284 rows of the DataFrame dataset1 for training.

the predict() method of the KMeans() function to predict the cluster labels of the data in the range dataset1.iloc[285:426]. The resulting cluster labels are stored in the variable y kmeans 1. The predict() method is used to predict the cluster labels of new data based on the trained k-means model. In this case, the method is applied to the data in the range dataset1.iloc[285:426], which corresponds to the testing set. The resulting cluster labels are stored in the variable y kmeans 1. The cluster labels are integers that indicate which cluster each data point belongs to. The predict() method is useful when we want to apply the trained k-means model to new data and obtain the corresponding cluster labels.

the silhouette score of the k-means clustering model. The silhouette score is a measure of how well the data points are separated into clusters. A higher score indicates that the data points are well-separated and belong to distinct clusters, while a lower score indicates that the data points are poorly separated and may belong to overlapping clusters. The silhouette score() function takes three arguments: the data used for training the model, the cluster labels assigned by the model, and the distance metric used to calculate the silhouette score. The resulting score is stored in the variable score. A silhouette score of 0.99 is obtained, which is a very high score. This indicates that the data points are well-separated and belong to distinct clusters. The high score suggests that the k-means clustering model has performed well in separating the data points into two clusters. Same operation perform for the second dataset III and we got 0.73.

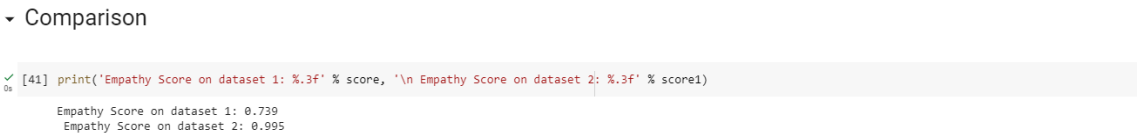

Figure 6: Results

Compare to dataset II, dataset III got is less result. But this results proved that HR can used this dataset for the empathy assessment.

3 Conclusions

The research into measuring empathy using eye-tracking data achieved a 99% with dataset III , 73% with dataset II and 99% with whole dataset . Success rate, showing that estimating empathy simply through the examination of eyetracking data is feasible. Eye characteristics such as pupil dilation, eye location, and gaze can be used to determine empathy. So assessment of empathy is possible from the eye tracker data. Machine learning models can be trained using point-of-gaze-based pupil dilation data, and memory questionnaires to predict empathy. The EyeT4Empathy dataset contains measurements of pupil diameter and gaze location for foraging for visual information. The text also mentions that the more severe the symptoms of depression were, the less dilated the pupils would become, which can be used to analyze depression based on empathy. With an 99% accuracy guarantee, our startup's business model aims to make it possible for HR departments to think about employing our services for empathy testing during the hiring process.

Such an inquiry can identify behavioural patterns and give more specific details regarding a person's empathy in social situations. It is crucial to emphasise that using only eyetracking data to determine empathy should be done with caution. It is advised that the findings be confirmed using multiple measures of empathy in order to provide reliable ratings.

References

[1] S. Baron-Cohen and S. Wheelwright. The empathy quotient: an investigation of adults with asperger syndrome or high functioning autism, and normal sex differences. *Journal of autism and developmental disorders*, 34:163–175, 2004.

[2] B. T. Carter and S. G. Luke. Best practices in eye tracking research. *International Journal of Psychophysiology*, 155:49–62, 2020.

[3] M. Dubas. Using eye tracker data to assess human empathy and attention. Master's thesis, OsloMet-storbyuniversitetet, 2022.

[4] K. Holmqvist, S. L. Örbom, I. T. Hooge, D. C. Niehorster, R. G. Alexander, R. Andersson, J. S. Benjamins, P. Blignaut, A.-M. Brouwer, L. L. Chuang, et al. Eye tracking: empirical foundations for a minimal reporting guideline. *Behavior research methods*, 55(1):364–416, 2023.

[5] J. L. Kröger, O. H.-M. Lutz, and F. Müller. What does your gaze reveal about you? on the privacy implications of eye tracking. *Privacy and Identity Management. Data for Better Living: AI and Privacy: 14th IFIP WG 9.2, 9.6/11.7, 11.6/SIG 9.2. 2 International Summer School, Windisch, Switzerland, August 19–23, 2019, Revised Selected Papers 14*, pages 226–241, 2020.

[6] P. Lencastre, S. Bhurtel, A. Yazidi, G. B. e Mello, S. Denysov, and P. G. Lind. Eyet4empathy: Dataset of foraging for visual information, gaze typing and empathy assessment. *Scientific Data*, 9(1):752, 2022.

[7] C. Malakcioglu. Empathy assessment scale. *Northern Clinics of Istanbul*, 9(4):358–366, 2022.