AlzheimersPredictionEDA

April 6, 2025

0.0.1 Hypothesis Testing

[2]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# read data into a data frame
df = pd.read_csv("alzheimers_prediction_dataset.csv")
df
```

[2]:
```
            Country  Age  Gender  Education Level   BMI  \
0             Spain   90    Male                1  33.0
1         Argentina   72    Male                7  29.9
2      South Africa   86  Female               19  22.9
3             China   53    Male               17  31.2
4            Sweden   58  Female                3  30.0
…               …  …     …                  …    …
74278        Russia   60  Female                3  22.6
74279            UK   58    Male               18  30.6
74280         Spain   57  Female               13  28.2
74281        Brazil   73  Female                7  29.0
74282        Norway   57  Female                1  31.7

      Physical Activity Level Smoking Status Alcohol Consumption Diabetes  \
0                      Medium          Never        Occasionally       No
1                      Medium         Former               Never       No
2                        High        Current        Occasionally       No
3                         Low          Never           Regularly      Yes
4                        High         Former               Never      Yes
…                         …             …                  …      …
74278                    High         Former               Never       No
74279                     Low          Never        Occasionally      Yes
74280                  Medium          Never           Regularly       No
74281                     Low          Never           Regularly       No
74282                     Low        Current           Regularly       No

      Hypertension  … Dietary Habits Air Pollution Exposure  \
0               No  …        Healthy                   High
```

```
1               No  …        Healthy                  Medium
2              Yes  …        Average                  Medium
3               No  …        Healthy                  Medium
4               No  …      Unhealthy                    High
…              …  …           …                        …
74278           No  …        Average                    High
74279           No  …        Average                  Medium
74280           No  …        Healthy                     Low
74281           No  …        Healthy                     Low
74282           No  …        Average                     Low

       Employment Status Marital Status Genetic Risk Factor (APOE- 4 allele)  \
0                Retired         Single                                   No
1             Unemployed        Widowed                                   No
2               Employed         Single                                   No
3                Retired         Single                                   No
4               Employed        Married                                   No
…                    …            …                                    …
74278         Unemployed        Widowed                                   No
74279         Unemployed         Single                                   No
74280           Employed         Single                                  Yes
74281           Employed        Widowed                                   No
74282         Unemployed         Single                                   No

      Social Engagement Level Income Level Stress Levels  \
0                         Low       Medium           High
1                        High          Low           High
2                         Low       Medium           High
3                        High       Medium            Low
4                         Low       Medium           High
…                        …           …             …
74278                  Medium         High         Medium
74279                  Medium         High           High
74280                    High          Low            Low
74281                     Low          Low           High
74282                     Low       Medium         Medium

      Urban vs Rural Living Alzheimer's Diagnosis
0                     Urban                    No
1                     Urban                    No
2                     Rural                    No
3                     Rural                    No
4                     Rural                    No
…                     …                   …
74278                 Rural                    No
74279                 Rural                    No
74280                 Rural                    No
```

```
74281                Rural                     No
74282                Urban                     No

[74283 rows x 25 columns]
```

[3]:
```
df.columns = df.columns.str.replace("’", "'")  # replace curly apostrophes with␣
 ↪straight ones
df = df.rename(columns={"Family History of Alzheimer's": "Family History"})
print(df.columns.tolist())
```

```
['Country', 'Age', 'Gender', 'Education Level', 'BMI', 'Physical Activity
Level', 'Smoking Status', 'Alcohol Consumption', 'Diabetes', 'Hypertension',
'Cholesterol Level', 'Family History', 'Cognitive Test Score', 'Depression
Level', 'Sleep Quality', 'Dietary Habits', 'Air Pollution Exposure', 'Employment
Status', 'Marital Status', 'Genetic Risk Factor (APOE- 4 allele)', 'Social
Engagement Level', 'Income Level', 'Stress Levels', 'Urban vs Rural Living',
"Alzheimer's Diagnosis"]
```

Split features into numerical and categorical columns.

[5]:
```
numerical_features = df.select_dtypes(include=['int64', 'float64']).columns
categorical_features = df.select_dtypes(include=['object', 'category', 'bool']).
 ↪columns
```

Loop through numerical features and run t-tests

[7]:
```
from scipy.stats import ttest_ind

# Convert Alzheimer's (ind. var.) to binary
df["Alzheimer's Diagnosis"] = df["Alzheimer's Diagnosis"].map({'Yes': 1, 'No':␣
 ↪0})

from scipy.stats import ttest_ind

for col in numerical_features:
    group1 = df[df["Alzheimer's Diagnosis"] == 1][col].dropna()
    group0 = df[df["Alzheimer's Diagnosis"] == 0][col].dropna()

    if len(group1) > 1 and len(group0) > 1 and group1.std() > 0 and group0.
 ↪std() > 0:
        t_stat, p = ttest_ind(group1, group0)
        print(f"{col}: p = {p:.4f}")
    else:
        print(f"{col}: cannot perform t-test (insufficient data or zero␣
 ↪variance)")
```

```
Age: p = 0.0000
Education Level: p = 0.3091
BMI: p = 0.6426
Cognitive Test Score: p = 0.7557
```

Age appears to be the only numerical feature that shows a statistically significant relationship with the binary Alzheimer's diagnosis variable ($p < 0.05$). In contrast, Education Level, BMI, and Cognitive Test Score all have high p-values, indicating no statistically significant difference in their means between individuals diagnosed with Alzheimer's and those not diagnosed. Therefore, Age may be a meaningful predictor, while the other features are less likely to contribute individually to classification performance.

Check for any present null values

[10]:
```
print(df[["Alzheimer's Diagnosis", "Age", "Education Level", "BMI", "Cognitive␣
↪Test Score"]].isnull().sum())
```

```
Alzheimer's Diagnosis    0
Age                      0
Education Level          0
BMI                      0
Cognitive Test Score     0
dtype: int64
```

There are no null values

Loop through categorical features and run chi-square tests

[13]:
```
from scipy.stats import chi2_contingency
import pandas as pd

for col in categorical_features:
    contingency_table = pd.crosstab(df[col], df["Alzheimer's Diagnosis"])
    chi2, p_value, dof, expected = chi2_contingency(contingency_table)
    print(f"{col}: p = {p_value:.4f}")
```

```
Country: p = 0.0000
Gender: p = 0.7156
Physical Activity Level: p = 0.7007
Smoking Status: p = 0.5682
Alcohol Consumption: p = 0.2818
Diabetes: p = 0.4721
Hypertension: p = 0.7544
Cholesterol Level: p = 0.5719
Family History: p = 0.0000
Depression Level: p = 0.7476
Sleep Quality: p = 0.9543
Dietary Habits: p = 0.4662
Air Pollution Exposure: p = 0.4601
Employment Status: p = 0.2761
Marital Status: p = 0.9126
Genetic Risk Factor (APOE- 4 allele): p = 0.0000
Social Engagement Level: p = 0.6845
Income Level: p = 0.1653
Stress Levels: p = 0.3603
```

```
Urban vs Rural Living: p = 0.2665
Alzheimer's Diagnosis: p = 0.0000
```

Chi-square tests indicate that Country, Family History, and the Genetic Risk Factor (APOE- 4 allele) are significantly associated with Alzheimer's diagnosis ($p < 0.05$). This suggests that these factors could be important predictors for identifying individuals at risk. On the other hand, variables such as Gender, Physical Activity Level, and Income Level do not show significant associations with Alzheimer's, and are unlikely to contribute much to predictive models based on this dataset.

0.0.2 Feature Selection

So far, we have identified important variables as: Age, Country, Family History, and the Genetic Risk Factor (APOE- 4 allele)

0.0.3 Prepare data for modeling

[36]:
```
from sklearn.preprocessing import StandardScaler

# One-hot encoding for categorical variables
df_encoded = pd.get_dummies(df, drop_first=True)  # drop_first to avoid␣
 ↪multicollinearity

scaler = StandardScaler()
df_encoded[['Age', 'BMI']] = scaler.fit_transform(df_encoded[['Age', 'BMI']])
```

0.0.4 Model Data

[43]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split

# Split the data into training and testing sets
X = df_encoded.drop("Alzheimer's Diagnosis", axis=1)  # Features
y = df_encoded["Alzheimer's Diagnosis"]  # Target

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
 ↪random_state=42)

# Initialize the model and fit it (max iterations = 1000)
model = LogisticRegression(max_iter=1000)
model.fit(X_train, y_train)

# Evaluate the model
print(f"Accuracy: {model.score(X_test, y_test):.4f}")
```

```
Accuracy: 0.7145
```