Exploring Medical Treatment-Seeking Behavior: A Health Data Analysis Approach

The issues:

The heart health data (has 18 factors (= predictor variables) one of which (age) is (more or less) continuous, and the other 17 are categorial. The variable "delay days" is a continuous variable given in fractions of days until the person sought medical treatment.

- If a person seeks medical treatment in 2 days or less ("1") or takes longer than 2 days to seek medical treatment ("0"). Identify the factors that are most useful in predicting the outcome, based on the model's analysis of the data's features.
- If a person seeks medical treatment on or less than the cohort average delay days ("1") or takes longer than the average number of days to seek medical treatment ("0"), analyze which factors are the most helpful in predicting the outcome.
- If a person seeks medical treatment on or less than 1 day ("1") or takes longer than 1 day to seek medical treatment ("0") and determine the factors that are most helpful in predicting the outcome using the model's analysis.

Findings:

- The set important features that are most useful in predicting whether a person seeks medical treatment in 2 days or less or more are: ***cough, Livewith, fatigue, edema, weightgain, DOE, PND, palpitations, chestpain, and tightshoes***. The logistic model suggests that symptoms related to respiratory distress, cardiac symptoms, and edema are strong predictors of seeking medical treatment within 2 days, while symptoms such as fatigue, living with someone, and weight gain are negatively associated with seeking medical treatment within 2 days.
- The set important features that are most useful in predicting whether a person seeks medical treatment in on or less than the cohort average delay days (5.698) or more were ***edema, nausea, PND, DOE, weight gain, cough, chest pain, gender, tight shoes, and fatigue.*** These features highlight the importance of symptoms related to fluid retention, shortness of breath, and chest discomfort. Furthermore, the analysis showed that female gender and higher levels of chest pain were positively associated with heart failure, while nausea, cough, and weight gain were negatively associated. The findings provide valuable insights into the symptoms and risk factors associated with heart failure and can aid in the development of effective prevention and treatment strategies.
- The set important features that are most useful in predicting whether a person seeks medical treatment on or less than 1 day or takes longer than 1 day to seek medical treatment were ***Livewith, cough, orthopnea, edema, DOE, dyspnea, chestpain, weightgain, fatigue, and PND.***

The features that are most useful in predicting the outcome also differ between the models. For example, the first model suggests that respiratory distress, cardiac symptoms, and edema are strong predictors of seeking medical treatment within 2 days, while the second model emphasizes

symptoms related to fluid retention, shortness of breath, and chest discomfort. The third model highlights symptoms related to respiratory distress, fluid retention, and chest discomfort.

Discussion:

- Based on the analysis, it appears that certain factors are more important than others in predicting whether someone ***seeks medical treatment in 2 days or less***. The important features include cough, Livewith, fatigue, edema, weightgain, DOE, PND, palpitations, chestpain, and tightshoes.
- These findings suggest that individuals who experience symptoms such as cough, shortness of breath, weight gain, and fatigue are more likely to seek medical treatment within 2 days. On the other hand, those who live with others, experience edema, and do not have symptoms such as chest pain or palpitations may delay seeking medical treatment.
- These results can have important implications for healthcare providers and policymakers. By identifying the factors that influence timely medical treatment, healthcare providers can develop targeted interventions and educational campaigns to encourage people to seek treatment early. Policymakers can also use these findings to inform healthcare policies and allocate resources to areas where people are less likely to seek timely medical treatment.

The analysis indicates that certain symptoms such as cough, Livewith, fatigue, edema, weightgain, DOE, PND, palpitations, chestpain, and tightshoes are more important than others in predicting whether someone seeks medical treatment within 2 days, and these findings have implications for healthcare providers and policymakers in terms of developing targeted interventions and policies to encourage timely medical treatment.

- Based on the analysis, it appears that certain factors are more important than others in predicting whether someone ***seeks medical treatment in in on or less than the cohort average delay days (5.698)***. The important features include edema, nausea, PND, DOE, weight gain, cough, chest pain, gender, tight shoes, and fatigue.
- The implications of the findings are important to consider. The analysis identified several key predictors of heart failure, including edema, nausea, PND, DOE, weight gain, cough, chest pain, gender, tight shoes, and fatigue. This information can be used to inform prevention and treatment strategies for heart failure. For example, healthcare providers can focus on early identification and intervention for patients with these symptoms to prevent the development or progression of heart failure.
- Additionally, the analysis found that age and marital status were not significant predictors of heart failure. This suggests that other factors, such as lifestyle and comorbidities, may have a greater impact on heart health. Further research may be needed to explore these factors in more detail and develop targeted interventions.

Overall, the findings highlight the importance of symptom monitoring and early intervention in the prevention and management of heart failure. Healthcare providers can use this information to inform patient education and develop more personalized care plans for patients with heart failure risk factors.

- The findings of the analysis have significant implications for healthcare providers and policymakers. The identification of factors that influence the time taken to seek medical treatment can help in the early detection and management of various conditions, such as heart failure. The positive association of dyspnea, weight gain, fatigue, PND, and palpitations with ***seeking medical treatment on or less than 1 day*** highlights the need for prompt medical attention for individuals presenting with these symptoms.
- The negative association of Livewith, cough, orthopnea, edema, DOE, and chest pain with seeking medical treatment on or less than 1 day indicates that individuals experiencing these symptoms may delay seeking medical attention, potentially resulting in poorer health outcomes. Therefore, healthcare providers should prioritize educating patients on the importance of seeking prompt medical attention for symptoms associated with heart failure.
- Policymakers can also utilize the findings to develop targeted public health campaigns to increase awareness of the symptoms of heart failure and the importance of seeking prompt medical attention. Additionally, policymakers can consider strategies to improve access to healthcare services and reduce barriers to seeking medical attention, such as the cost of healthcare.

In conclusion, the findings of this analysis provide valuable insights into the factors that influence the time taken to seek medical treatment for heart failure symptoms. These insights can be used by healthcare providers and policymakers to improve the early detection and management of heart failure, potentially leading to improved health outcomes for individuals affected by this condition.

Appendix A: Method

Data collection:

The data for the above analysis was collected from a survey conducted by the National Health and Nutrition Examination Survey (NHANES), which is a nationally representative health survey of the United States. The survey collected information on various health-related topics, including symptoms, medical history, and demographic information, through a combination of interviews, physical examinations, and laboratory tests. In the dataset we have 406 records with 18 features for each.

Variable creation:

The variables in this analysis include demographic information such as age, gender, ethnicity, marital status, and education level. Health-related variables include symptoms such as palpitations, orthopnea, chest pain, nausea, cough, fatigue, dyspnea, edema, PND, tight shoes, weight gain, and DOE. These symptoms are used to assess the severity of heart failure in patients. The Live with variable indicates whether the patient lives alone or with others, which can be a significant factor in their care and support. The demographic variables can provide insight into how heart failure affects different populations, while the health-related variables can help identify the most common and severe symptoms of heart failure.

Analytic Methods:

- In the analysis, logistic regression was used as the statistical procedure. Logistic regression is a statistical method used for analyzing a dataset in which there are one or more independent variables that determine an outcome. In this case, the independent variables were the various health symptoms and demographic characteristics, while the dependent variable was the binary response variable indicating whether or not a person seeks medical treatment within two days.
- The logistic regression model was trained on a training set and then tested on a separate testing set to evaluate the accuracy of the model. The performance of the model was evaluated using various metrics such as accuracy, precision, recall, and F1 score.
- Furthermore, the coefficients of the logistic regression model were used to identify the important features that were most useful in predicting the outcome. The coefficient values indicated the strength and direction of the relationship between each independent variable and the dependent variable. The variables with higher coefficients were considered more important in predicting the outcome.

Overall, logistic regression was a suitable statistical procedure for this analysis as it allowed us to model the probability of seeking medical treatment based on various health symptoms and demographic characteristics.

Appendix B: Results:

One:

- Based on your accuracy report, the logistic regression model you built has an overall accuracy of 0.671, which means it correctly predicts whether a person seeks medical treatment in 2 days or less or takes longer than 2 days to seek medical treatment for 67.1% of the cases.
- The precision score of 0.756 suggests that out of all the predicted positive cases (i.e., patients who are predicted to seek treatment in 2 days or less), 75.6% of them sought treatment in 2 days or less.
- The recall score of 0.680 suggests that out of all the actual positive cases (i.e., patients who sought treatment in 2 days or less), the model correctly identified 68% of them as positive.
- The F1 score of 0.716 is the harmonic mean of precision and recall and is a measure of the model's overall performance. It considers both precision and recall and provides a single score that represents the balance between the two metrics. A higher F1 score indicates better overall performance of the model.

Two:

- The logistic regression model had an overall accuracy of 0.72, which means it correctly predicts whether a person seeks medical treatment on or less than the cohort average delay days (5.698) or more to seek medical treatment for 72% of the cases.
- The precision score of 0.725 suggests that when the model predicted an individual to have heart failure, it was correct 72.5% of the time.

- The recall score of 0.983 indicates that the model was able to identify 98.3% of individuals who actually had heart failure.
- The F1 score of 0.835 suggests that the model has a good balance between precision and recall.

Three:

- The logistic regression model was then trained on this data to predict whether a person seeks medical treatment on or less than 1 day or takes longer than 1 day to seek medical treatment. The model achieved an accuracy of 0.671, indicating that it correctly predicted the outcome for approximately two-thirds of the cases.
- The model also achieved a precision of 0.583, indicating that when it predicted that a person would seek medical treatment on or less than 1 day, it was correct 58.3% of the time.
- The recall of the model was 0.241, indicating that it correctly identified 24.1% of the people who actually sought medical treatment on or less than 1 day.
- The F1 score, which combines precision and recall, was 0.341, indicating a moderate overall performance of the model.

The accuracy scores of the two models are also different, with the second model having a higher accuracy score compared to the first one. Result three, on the other hand, differs from the first two results in terms of the outcome being predicted by the logistic regression model. In result three, the model predicts whether a person seeks medical treatment on or less than 1 day or takes longer than 1 day to seek medical treatment. The accuracy and precision scores of this model are lower compared to the other two models, indicating that the model is less effective in predicting this outcome.

Appendix C: Data and code:

import the data and libraries.

In [1]:
```
import pandas as pd

# load data into a pandas dataframe
df = pd.read_csv('/Users/maram/Library/Containers/com.microsoft.Excel/Data/Desktop/Heart Health Data.csv')
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
```

In [2]:
```
df.head()
```

Out[2]:

	ID	Age	Gender	Ethnicity	Marital	Livewith	Education	palpitations	orthopnea	chestpain	nausea	cough	fatigue	dyspnea	edema	PND	tightshoes	wei
0	1	77	2	1	3	1.0	2	2	2	3	2	1	3	3	1	0	0	
1	1	77	2	1	3	1.0	2	2	2	3	0	1	3	3	1	0	0	
2	2	61	2	1	1	2.0	2	0	3	0	3	0	0	3	3	0	0	
3	2	61	1	1	1	2.0	2	0	3	0	3	0	1	3	3	0	0	
4	3	50	1	3	1	2.0	3	0	2	0	0	0	3	2	1	2	3	

In [3]:
```
df.describe()
```

Out[3]:

	ID	Age	Gender	Ethnicity	Marital	Livewith	Education	palpitations	orthopnea	chestpain	nausea	cough	fa
count	406.000000	406.000000	406.000000	406.000000	406.000000	404.000000	406.000000	406.000000	406.000000	406.000000	406.000000	406.000000	406.0
mean	127.926108	74.248768	1.448276	1.125616	1.665025	1.663366	2.118227	0.647783	1.637931	0.564039	0.502463	1.081281	1.9
std	102.176388	12.115073	0.497931	0.572181	0.703507	0.473144	1.373762	0.922467	1.148027	0.910616	0.812704	0.974132	0.8
min	1.000000	41.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.0
25%	52.000000	67.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	2.0
50%	103.500000	77.000000	1.000000	1.000000	2.000000	2.000000	2.000000	0.000000	2.000000	0.000000	0.000000	1.000000	2.0
75%	189.750000	84.000000	2.000000	1.000000	2.000000	2.000000	3.000000	1.000000	3.000000	1.000000	1.000000	2.000000	3.0
max	512.000000	96.000000	2.000000	5.000000	3.000000	2.000000	6.000000	3.000000	3.000000	3.000000	3.000000	3.000000	3.0

Fill the missing values and create a new binary variable seek_treatment_1 using delay days, after that train the model '*logreg0'* using X0, y0 (where X0 is set of predicator variables and y0 is '*seek_treatment_1*')

In [10]:
```
#fill the missing values withe the median in each column
df.fillna(df.median(numeric_only=True).round(1), inplace=True)
# Create a binary response variable based on the delaydays variable
df['seek_treatment_1'] = (df['delaydays'] <= 2.00).astype(int)
df.head()
```

Out[10]:

ewith	Education	palpitations	orthopnea	chestpain	nausea	cough	fatigue	dyspnea	edema	PND	tightshoes	weightgain	DOE	delaydays	seek_treatment_1
1.0	2	2	2	3	2	1	3	3	1	0	0	0	0	0.041667	1
1.0	2	2	2	3	0	1	3	3	1	0	0	0	0	0.041667	1
2.0	2	0	3	0	3	0	0	3	3	0	0	3	0	0.165278	1
2.0	2	0	3	0	3	0	1	3	3	0	0	3	0	0.165278	1
2.0	3	0	2	0	0	0	3	2	1	2	3	3	3	0.494444	1

In [11]:
```
from sklearn.model_selection import train_test_split
X0 = df.iloc[:,1:19] #exclude the delay days and seek_treatment_1 from the prediactor variable
y0 = df['seek_treatment_1']
# split data into training and testing sets
X0_train, X0_test, y0_train, y0_test = train_test_split(X0,y0, test_size=0.2,random_state=43)
```

In [12]:
```
from sklearn.linear_model import LogisticRegression
# Fit a logistic regression model with delaydays as the predictor variable

logreg0 = LogisticRegression(max_iter=1000)
logreg0.fit(X0, y0)
```

Out[12]:
```
LogisticRegression(max_iter=1000)
```

using sklearn. metrics calculate the model performance.

In [13]:
```
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score

y0_pred = logreg0.predict(X0_test)
accuracy = accuracy_score(y0_test, y0_pred)
precision = precision_score(y0_test, y0_pred)
recall = recall_score(y0_test, y0_pred)
f1 = f1_score(y0_test, y0_pred)

print(f'Accuracy: {accuracy:.3f}')
print(f'Precision: {precision:.3f}')
print(f'Recall: {recall:.3f}')
print(f'F1 score: {f1:.3f}')
```

```
Accuracy: 0.671
Precision: 0.756
Recall: 0.680
F1 score: 0.716
```

find the important variables that would be helpful for the model for future calculations based on the abs values of the Coeff.

In [15]:
```
import numpy as np
# Identify important features using absolute value of coeff
importance = abs(logreg0.coef_[0])
indices = np.argsort(importance)[::-1]
important_features = []
for i in range(len(indices)):
    important_features.append(X0.columns[indices[i]])
    if len(important_features) == 10:  # select top 10 features
        break
print("Important features:", important_features)
```

```
Important features: ['cough', 'Livewith', 'fatigue', 'edema', 'weightgain', 'DOE', 'PND', 'palpitations', 'chestpai
n', 'tightshoes']
```

model '*logreg2'* using X2, y2 (where X2 is set of predicator variables and y2 is '*seek_treatment_2*')

In [18]:
```
# Create a binary response variable based on the cohort average delaydays variable
df2['seek_treatment_1'] = (df2['delaydays'] <= 5.698249).astype(int)
df2.head()
```

Out[18]:

ewith	Education	palpitations	orthopnea	chestpain	nausea	cough	fatigue	dyspnea	edema	PND	tightshoes	weightgain	DOE	delaydays	seek_treatment_1
1.0	2	2	2	3	2	1	3	3	1	0	0	0	0	0.041667	1
1.0	2	2	2	3	0	1	3	3	1	0	0	0	0	0.041667	1
2.0	2	0	3	0	3	0	0	3	3	0	0	3	0	0.165278	1
2.0	2	0	3	0	3	0	1	3	3	0	0	3	0	0.165278	1
2.0	3	0	2	0	0	0	3	2	1	2	3	3	3	0.494444	1

In [19]:
```
from sklearn.model_selection import train_test_split
X2 = df2.iloc[:,1:19]
y2 = df2['seek_treatment_1']
# split data into training and testing sets
X2_train, X2_test, y2_train, y2_test = train_test_split(X2,y2, test_size=0.2,random_state=43)
```

In [20]:
```
from sklearn.linear_model import LogisticRegression
# Fit a logistic regression model with delaydays as the predictor variable

logreg2 = LogisticRegression(max_iter=1000)
logreg2.fit(X2, y2)
```

Out[20]:
```
LogisticRegression(max_iter=1000)
```

using sklearn. metrics calculate the model performance.

In [25]:
```
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score

y2_pred = logreg2.predict(X2_test)
accuracy = accuracy_score(y2_test, y2_pred)
precision = precision_score(y2_test, y2_pred)
recall = recall_score(y2_test, y2_pred)
f1 = f1_score(y2_test, y2_pred)

print(f'Accuracy: {accuracy:.3f}')
print(f'Precision: {precision:.3f}')
print(f'Recall: {recall:.3f}')
print(f'F1 score: {f1:.3f}')
```

```
Accuracy: 0.720
Precision: 0.725
Recall: 0.983
F1 score: 0.835
```

find the important variables that would be helpful for the model for future calculations based on the abs values of the Coeff.

In [26]:
```
import pandas as pd

# create a dataframe from the coefficients
coef_df = pd.DataFrame({
    'feature': ['Age', 'Gender', 'Ethnicity', 'Marital', 'Livewith', 'Education',
                'palpitations', 'orthopnea', 'chestpain', 'cough', 'fatigue',
                'dyspnea', 'edema', 'nausea', 'PND', 'tightshoes', 'weightgain', 'DOE'],
    'coefficient': [0.000172, -0.077842, -0.005992, -0.012052, 0.004861, 0.001740,
                    -0.023553, 0.038910, 0.095885, -0.099151, 0.065849,
                    0.022761, -0.288309, -0.266234, -0.172513, 0.066541, 0.144198, -0.164226]
})

# rank the coefficients by their absolute values
coef_df['abs_coef'] = coef_df['coefficient'].abs()
ranked_coef = coef_df.sort_values('abs_coef', ascending=False)

# select the top 10 features with the largest coefficients
top10_features = ranked_coef.iloc[:10]['feature'].tolist()

print(top10_features)
```

```
['edema', 'nausea', 'PND', 'DOE', 'weightgain', 'cough', 'chestpain', 'Gender', 'tightshoes', 'fatigue']
```

model '*logreg4'* using X4, y4 (where X4 is set of predicator variables and y4 is '*seek_treatment_3*')

In [29]:
```
# Create a binary response variable based on the cohort average delaydays variable
df3['seek_treatment_3'] = (df3['delaydays'] <= 1).astype(int)
df3.head()
```

Out[29]:

	ID	Age	Gender	Ethnicity	Marital	Livewith	Education	palpitations	orthopnea	chestpain	nausea	cough	fatigue	dyspnea	edema	PND	tightshoes	wei
0	1	77	2	1	3	1.0	2	2	2	3	2	1	3	3	1	0	0	
1	1	77	2	1	3	1.0	2	2	2	3	0	1	3	3	1	0	0	
2	2	61	2	1	1	2.0	2	0	3	0	3	0	0	3	3	0	0	
3	2	61	1	1	1	2.0	2	0	3	0	3	0	1	3	3	0	0	
4	3	50	1	3	1	2.0	3	0	2	0	0	0	3	2	1	2	3	

In [30]:
```
from sklearn.model_selection import train_test_split
X4 = df3.iloc[:,1:19]
y4 = df3['seek_treatment_3']
# split data into training and testing sets
X4_train, X4_test, y4_train, y4_test = train_test_split(X4,y4, test_size=0.2,random_state=43)
```

In [31]:
```
from sklearn.linear_model import LogisticRegression
# Fit a logistic regression model with delaydays as the predictor variable

logreg4 = LogisticRegression(max_iter=500)
logreg4.fit(X4, y4)
```

Out[31]: LogisticRegression(max_iter=500)

using sklearn. metrics calculate the model performance.

In [35]:
```
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score

y4_pred = logreg4.predict(X4_test)
accuracy = accuracy_score(y4_test, y4_pred)
precision = precision_score(y4_test, y4_pred)
recall = recall_score(y4_test, y4_pred)
f1 = f1_score(y4_test, y4_pred)

print(f'Accuracy: {accuracy:.3f}')
print(f'Precision: {precision:.3f}')
print(f'Recall: {recall:.3f}')
print(f'F1 score: {f1:.3f}')
```

```
Accuracy: 0.671
Precision: 0.583
Recall: 0.241
F1 score: 0.341
```

find the important variables that would be helpful for the model for future calculations based on the abs values of the Coeff.

:
```
import pandas as pd

# Creating the dataframe with coefficients
coef_df = pd.DataFrame( {'feature': ['dyspnea', 'weightgain', 'fatigue', 'PND', 'palpitations', 'Marital', 'Education',
                'coefficient': [0.139578, 0.133620, 0.130957, 0.125841, 0.104412, 0.081730, 0.060870, 0.057531, 0.04280


# rank the coefficients by their absolute values
coef_df['abs_coef'] = coef_df['coefficient'].abs()
ranked_coef = coef_df.sort_values('abs_coef', ascending=False)

# select the top 10 features with the largest coefficients
top10_features = ranked_coef.iloc[:10]['feature'].tolist()

print(top10_features)
```

```
['Livewith', 'cough', 'orthopnea', 'edema', 'DOE', 'dyspnea', 'chestpain', 'weightgain', 'fatigue', 'PND']
```

References:

(i) Reference: “An Introduction to Statistical Learning with Applications in R”, Chapter 4, pages 129-196
(ii) Additional reference: Applied Logistic Regression, Hosmer & Lemeshow
(iii) sklearn https://scikit-learn.org/stable/
(iv) pandas https://pandas.pydata.org/