Absenteeism_Data

October 23, 2024

[26]:
```
import pandas as pd
df= pd.read_csv('C:/Users/User/Downloads/Absenteeism-data.csv')
df.head()
```

[26]:

	ID	Reason for Absence	Date	Transportation Expense	\
0	11	26	07/07/2015	289	
1	36	0	14/07/2015	118	
2	3	23	15/07/2015	179	
3	7	7	16/07/2015	279	
4	11	23	23/07/2015	289	

	Distance to Work	Age	Daily Work Load Average	Body Mass Index	Education	\
0	36	33	239.554	30	1	
1	13	50	239.554	31	1	
2	51	38	239.554	31	1	
3	5	39	239.554	24	1	
4	36	33	239.554	30	1	

	Children	Pets	Absenteeism Time in Hours
0	2	1	4
1	1	0	0
2	0	0	2
3	2	0	4
4	2	1	2

[19]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 700 entries, 0 to 699
Data columns (total 12 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   ID                         700 non-null    int64
 1   Reason for Absence         700 non-null    int64
 2   Date                       700 non-null    object
 3   Transportation Expense     700 non-null    int64
 4   Distance to Work           700 non-null    int64
 5   Age                        700 non-null    int64
```

```
 6   Daily Work Load Average    700 non-null    float64
 7   Body Mass Index            700 non-null    int64
 8   Education                  700 non-null    int64
 9   Children                   700 non-null    int64
 10  Pets                       700 non-null    int64
 11  Absenteeism Time in Hours  700 non-null    int64
dtypes: float64(1), int64(10), object(1)
memory usage: 65.8+ KB
```

[5]:
```
df.describe()
```

[5]:

	ID	Reason for Absence	Transportation Expense
count	700.000000	700.000000	700.000000
mean	17.951429	19.411429	222.347143
std	11.028144	8.356292	66.312960
min	1.000000	0.000000	118.000000
25%	9.000000	13.000000	179.000000
50%	18.000000	23.000000	225.000000
75%	28.000000	27.000000	260.000000
max	36.000000	28.000000	388.000000

	Distance to Work	Age	Daily Work Load Average	Body Mass Index
count	700.000000	700.000000	700.000000	700.000000
mean	29.892857	36.417143	271.801774	26.737143
std	14.804446	6.379083	40.021804	4.254701
min	5.000000	27.000000	205.917000	19.000000
25%	16.000000	31.000000	241.476000	24.000000
50%	26.000000	37.000000	264.249000	25.000000
75%	50.000000	40.000000	294.217000	31.000000
max	52.000000	58.000000	378.884000	38.000000

	Education	Children	Pets	Absenteeism Time in Hours
count	700.000000	700.000000	700.000000	700.000000
mean	1.282857	1.021429	0.687143	6.761429
std	0.668090	1.112215	1.166095	12.670082
min	1.000000	0.000000	0.000000	0.000000
25%	1.000000	0.000000	0.000000	2.000000
50%	1.000000	1.000000	0.000000	3.000000
75%	1.000000	2.000000	1.000000	8.000000
max	4.000000	4.000000	8.000000	120.000000

[6]:
```
df.shape
```

[6]:
```
(700, 12)
```

[7]:
```
df['ID'].value_counts()
```

[7]:
```
3     113
28     74
34     48
20     42
22     41
11     39
15     36
36     32
24     30
14     27
33     24
10     22
1      22
17     19
5      18
18     16
13     14
25     10
27      7
30      7
6       7
23      7
7       6
9       6
2       5
29      5
32      5
26      5
12      3
31      3
19      3
21      2
8       1
16      1
Name: ID, dtype: int64
```

[27]:
```
# Map the 'Reason for Absence' column. Group the categorical nominal value into␣
 ↪Health for cases registered in the
#international classification of diseases(ICD), Personal for cases not␣
 ↪registered in the (ICD) and present for individuals
# who are fully available
reason_mapping = {
    **dict.fromkeys(range(1, 22), 'Health'),  # Reasons 1 to 21 mapped to␣
 ↪'Health'
    22: 'Personal',
    23: 'Personal',
    24: 'Personal',
```

```
    25: 'Personal',
    26: 'Personal',
    27: 'Personal',
    28: 'Personal',
    0:'Present'
}


df['Reason for Absence'] = df['Reason for Absence'].map(reason_mapping)
df
```

[27]:

	ID	Reason for Absence	Date	Transportation Expense	\
0	11	Personal	07/07/2015	289	
1	36	Present	14/07/2015	118	
2	3	Personal	15/07/2015	179	
3	7	Health	16/07/2015	279	
4	11	Personal	23/07/2015	289	
..	..	…	…	…	
695	17	Health	23/05/2018	179	
696	28	Health	23/05/2018	225	
697	18	Health	24/05/2018	330	
698	25	Personal	24/05/2018	235	
699	15	Personal	31/05/2018	291	

	Distance to Work	Age	Daily Work Load Average	Body Mass Index	\
0	36	33	239.554	30	
1	13	50	239.554	31	
2	51	38	239.554	31	
3	5	39	239.554	24	
4	36	33	239.554	30	
..	…	…	…	…	
695	22	40	237.656	22	
696	26	28	237.656	24	
697	16	28	237.656	25	
698	16	32	237.656	25	
699	31	40	237.656	25	

	Education	Children	Pets	Absenteeism Time in Hours
0	1	2	1	4
1	1	1	0	0
2	1	0	0	2
3	1	2	0	4
4	1	2	1	2
..	…	…	…	…
695	2	2	0	8
696	1	1	2	3
697	2	0	0	8

```
698          3         0     0                          2
699          1         1     1                          2

[700 rows x 12 columns]
```

[28]:
```
# Map the 'Reason for Absence' column. Group the categorical nominal value into␣
 ↪Health for cases registered in the
#international classification of diseases(ICD), Personal for cases not␣
 ↪registered in the (ICD) and present for individuals
# who are fully available
education_mapping = {1: 'High School', 2: 'Graduate', 3: 'Graduate', 4:␣
 ↪'Graduate'}


df['Education'] = df['Education'].map(education_mapping)
df
```

[28]:
```
     ID Reason for Absence        Date  Transportation Expense  \
0    11           Personal  07/07/2015                     289
1    36            Present  14/07/2015                     118
2     3           Personal  15/07/2015                     179
3     7             Health  16/07/2015                     279
4    11           Personal  23/07/2015                     289
..   ..                ...         ...                     ...
695  17             Health  23/05/2018                     179
696  28             Health  23/05/2018                     225
697  18             Health  24/05/2018                     330
698  25           Personal  24/05/2018                     235
699  15           Personal  31/05/2018                     291

     Distance to Work  Age  Daily Work Load Average  Body Mass Index  \
0                  36   33                  239.554               30
1                  13   50                  239.554               31
2                  51   38                  239.554               31
3                   5   39                  239.554               24
4                  36   33                  239.554               30
..                ...  ...                      ...              ...
695                22   40                  237.656               22
696                26   28                  237.656               24
697                16   28                  237.656               25
698                16   32                  237.656               25
699                31   40                  237.656               25

       Education  Children  Pets  Absenteeism Time in Hours
0    High School         2     1                          4
1    High School         1     0                          0
2    High School         0     0                          2
```

```
3    High School         2     0                          4
4    High School         2     1                          2
..           …       …   …                        …
695     Graduate         2     0                          8
696  High School         1     2                          3
697     Graduate         0     0                          8
698     Graduate         0     0                          2
699  High School         1     1                          2

[700 rows x 12 columns]
```

[29]:
```
# Extracting the day of the week (0=Monday, 1=Tuesday, ..., 6=Sunday)
df['Date'] = pd.to_datetime(df['Date'],format='%d/%m/%Y')
df['Day of Week'] = df['Date'].dt.dayofweek
df
```

[29]:
```
     ID Reason for Absence       Date  Transportation Expense  \
0    11           Personal 2015-07-07                     289
1    36            Present 2015-07-14                     118
2     3           Personal 2015-07-15                     179
3     7             Health 2015-07-16                     279
4    11           Personal 2015-07-23                     289
..   ..                …        …                       …
695  17             Health 2018-05-23                     179
696  28             Health 2018-05-23                     225
697  18             Health 2018-05-24                     330
698  25           Personal 2018-05-24                     235
699  15           Personal 2018-05-31                     291

     Distance to Work  Age  Daily Work Load Average  Body Mass Index  \
0                  36   33                  239.554               30
1                  13   50                  239.554               31
2                  51   38                  239.554               31
3                   5   39                  239.554               24
4                  36   33                  239.554               30
..                 …  …                        …                …
695                22   40                  237.656               22
696                26   28                  237.656               24
697                16   28                  237.656               25
698                16   32                  237.656               25
699                31   40                  237.656               25

       Education  Children  Pets  Absenteeism Time in Hours  Day of Week
0    High School         2     1                          4            1
1    High School         1     0                          0            1
2    High School         0     0                          2            2
3    High School         2     0                          4            3
```

```
4    High School          2     1                          2            3
..           …       …    …                        …          …
695     Graduate          2     0                          8            2
696  High School          1     2                          3            2
697     Graduate          0     0                          8            3
698     Graduate          0     0                          2            3
699  High School          1     1                          2            3

[700 rows x 13 columns]
```

[30]:
```
# Check for outliers by calculating Q1 (25th percentile) and Q3 (0.75th␣
 ↪percentile).
Q1 = df['Absenteeism Time in Hours'].quantile(0.25)
Q3 = df['Absenteeism Time in Hours'].quantile(0.75)

# Interquartile Range (IQR)
IQR = Q3 - Q1

# Define outlier boundaries
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR

print(f'Lower Bound: {lower_bound}, Upper Bound: {upper_bound}')
```

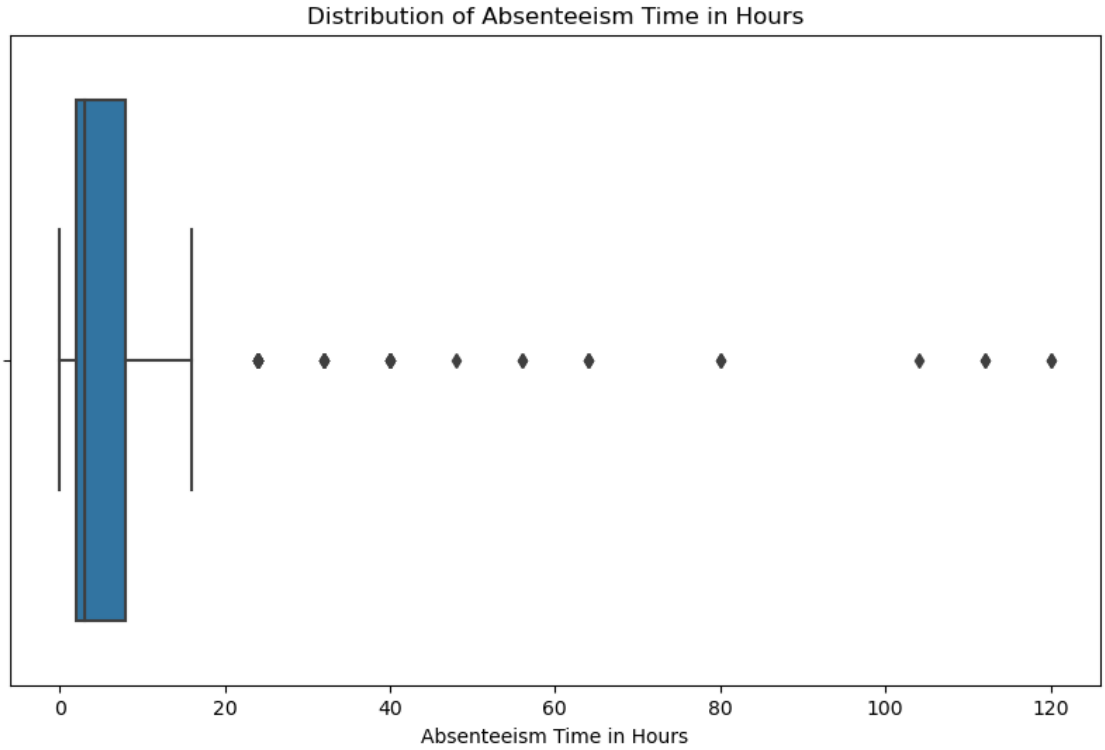
```
Lower Bound: -7.0, Upper Bound: 17.0
```

[31]:
```
#individual with extreme hours of absenteeism
filt = df[df['Absenteeism Time in Hours'] == 120]
filt
```

[31]:
```
     ID Reason for Absence       Date  Transportation Expense  \
323  14             Health 2016-11-14                     155
420  36             Health 2017-04-19                     118

     Distance to Work  Age  Daily Work Load Average  Body Mass Index  \
323                12   34                  284.031               25
420                13   50                  239.409               31

       Education  Children  Pets  Absenteeism Time in Hours  Day of Week
323  High School         2     0                        120            0
420  High School         1     0                        120            2
```

[32]:
```
# Use a boxlot to see the effect of outliers
import seaborn as sns
import matplotlib.pyplot as plt

plt.figure(figsize=(10,6))
sns.boxplot(x=df['Absenteeism Time in Hours'])
```

```
plt.title('Distribution of Absenteeism Time in Hours')
plt.show()
```



[33]:
```
# Makes sense that people with health issues might have extreme hours of␣
 ↪absenteeism I decided not to cap or remove the outliers
# Instead flag individuals who exceed 17 hours of absenteeism for investigation
df['Flag_Exceeds_17_Hours'] = df['Absenteeism Time in Hours'].apply(lambda x: 1␣
 ↪if x > 17 else 0)
df
```

[33]:

	ID	Reason for Absence	Date	Transportation Expense	\
0	11	Personal	2015-07-07	289	
1	36	Present	2015-07-14	118	
2	3	Personal	2015-07-15	179	
3	7	Health	2015-07-16	279	
4	11	Personal	2015-07-23	289	
..	..	…	…	…	
695	17	Health	2018-05-23	179	
696	28	Health	2018-05-23	225	
697	18	Health	2018-05-24	330	
698	25	Personal	2018-05-24	235	
699	15	Personal	2018-05-31	291	

```
     Distance to Work  Age  Daily Work Load Average  Body Mass Index  \
0                  36   33                  239.554               30
1                  13   50                  239.554               31
2                  51   38                  239.554               31
3                   5   39                  239.554               24
4                  36   33                  239.554               30
..                 …  …                    …                 …
695                22   40                  237.656               22
696                26   28                  237.656               24
697                16   28                  237.656               25
698                16   32                  237.656               25
699                31   40                  237.656               25

       Education  Children  Pets  Absenteeism Time in Hours  Day of Week  \
0    High School         2     1                          4            1
1    High School         1     0                          0            1
2    High School         0     0                          2            2
3    High School         2     0                          4            3
4    High School         2     1                          2            3
..           …       …   …                        …          …
695     Graduate         2     0                          8            2
696  High School         1     2                          3            2
697     Graduate         0     0                          8            3
698     Graduate         0     0                          2            3
699  High School         1     1                          2            3

     Flag_Exceeds_17_Hours
0                        0
1                        0
2                        0
3                        0
4                        0
..                     …
695                      0
696                      0
697                      0
698                      0
699                      0

[700 rows x 14 columns]
```

[45]:
```
df.describe()
```

[45]:
```
                ID  Transportation Expense  Distance to Work         Age  \
count  700.000000              700.000000        700.000000  700.000000
mean    17.951429              222.347143         29.892857   36.417143
```

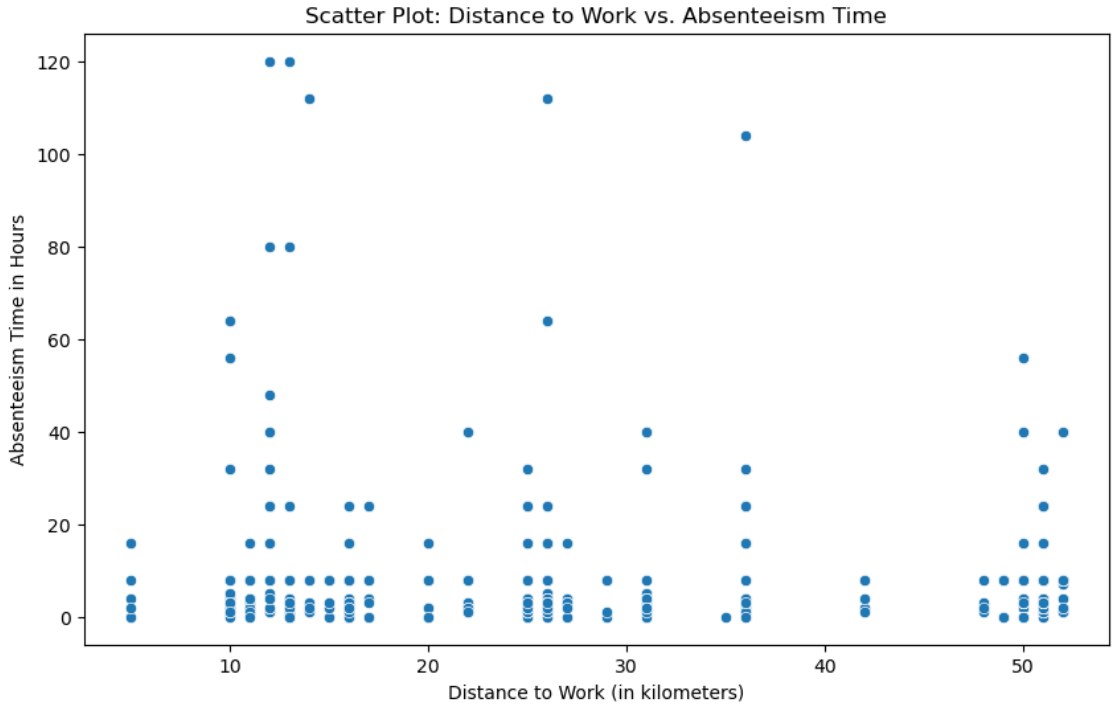
```
std     11.028144              66.312960         14.804446    6.379083
min      1.000000             118.000000          5.000000   27.000000
25%      9.000000             179.000000         16.000000   31.000000
50%     18.000000             225.000000         26.000000   37.000000
75%     28.000000             260.000000         50.000000   40.000000
max     36.000000             388.000000         52.000000   58.000000

       Daily Work Load Average  Body Mass Index    Children        Pets  \
count               700.000000       700.000000  700.000000  700.000000
mean                271.801774        26.737143    1.021429    0.687143
std                  40.021804         4.254701    1.112215    1.166095
min                 205.917000        19.000000    0.000000    0.000000
25%                 241.476000        24.000000    0.000000    0.000000
50%                 264.249000        25.000000    1.000000    0.000000
75%                 294.217000        31.000000    2.000000    1.000000
max                 378.884000        38.000000    4.000000    8.000000

       Absenteeism Time in Hours  Day of Week  Flag_Exceeds_17_Hours
count                 700.000000   700.000000             700.000000
mean                    6.761429     2.011429               0.058571
std                    12.670082     1.480396               0.234989
min                     0.000000     0.000000               0.000000
25%                     2.000000     1.000000               0.000000
50%                     3.000000     2.000000               0.000000
75%                     8.000000     3.000000               0.000000
max                   120.000000     6.000000               1.000000
```

[38]:
```
# Find IDs with high absenteeism. Those with an absenteeism higher than the 75␣
 ↪percentile
#are regarded to have high absenteeism
high_absenteeism = df.groupby('ID')['Absenteeism Time in Hours'].sum().
 ↪reset_index()

high_absenteeism_threshold = high_absenteeism['Absenteeism Time in Hours'].
 ↪quantile(0.75)
high_absenteeism_ids = high_absenteeism[high_absenteeism['Absenteeism Time in␣
 ↪Hours'] > high_absenteeism_threshold]


high_absenteeism_ids
```

[38]:
```
    ID  Absenteeism Time in Hours
2    3                        482
9   11                        442
12  14                        466
13  15                        251
18  20                        306
```

```
22  24                    254
26  28                    338
32  34                    314
33  36                    284
```

[40]:
```
# Analyzing the relationship between distance from work and absenteeism
import matplotlib.pyplot as plt
import seaborn as sns


plt.figure(figsize=(10, 6))
sns.scatterplot(x='Distance to Work', y='Absenteeism Time in Hours', data=df)
plt.title('Scatter Plot: Distance to Work vs. Absenteeism Time')
plt.xlabel('Distance to Work (in kilometers)')  #
plt.ylabel('Absenteeism Time in Hours')
plt.show()

# Calculate Correlation Coefficient
correlation = df['Distance to Work'].corr(df['Absenteeism Time in Hours'])
print(f"Correlation Coefficient: {correlation}")
```




```
Correlation Coefficient: -0.0805932243539209
```

[42]:
```
# exploring the relationship further by performing a simple linear regression␣
 ↪analysis
import statsmodels.api as sm

# Prepare the data for regression
X = df['Distance to Work']  # Independent variable
y = df['Absenteeism Time in Hours']  # Dependent variable

# Add a constant to the model
X = sm.add_constant(X)

# Fit the model
model = sm.OLS(y, X).fit()


model.summary()
```

[42]:
```
<class 'statsmodels.iolib.summary.Summary'>
"""
                            OLS Regression Results
==============================================================================
=====
Dep. Variable:     Absenteeism Time in Hours   R-squared:
0.006
Model:                                 OLS   Adj. R-squared:
0.005
Method:                      Least Squares   F-statistic:
4.563
Date:                     Wed, 23 Oct 2024   Prob (F-statistic):
0.0330
Time:                             14:28:30   Log-Likelihood:
-2767.9
No. Observations:                      700   AIC:
5540.
Df Residuals:                          698   BIC:
5549.
Df Model:                                1
Covariance Type:                 nonrobust
==============================================================================
====
                       coef    std err          t      P>|t|      [0.025
0.975]
------------------------------------------------------------------------------
----
const                8.8233      1.077      8.193      0.000       6.709
10.938
Distance to Work    -0.0690      0.032     -2.136      0.033      -0.132
```

```
-0.006
==============================================================================
Omnibus:                      804.854   Durbin-Watson:                   1.981
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            49632.690
Skew:                           5.697   Prob(JB):                         0.00
Kurtosis:                      42.647   Cond. No.                         75.3
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
"""
```

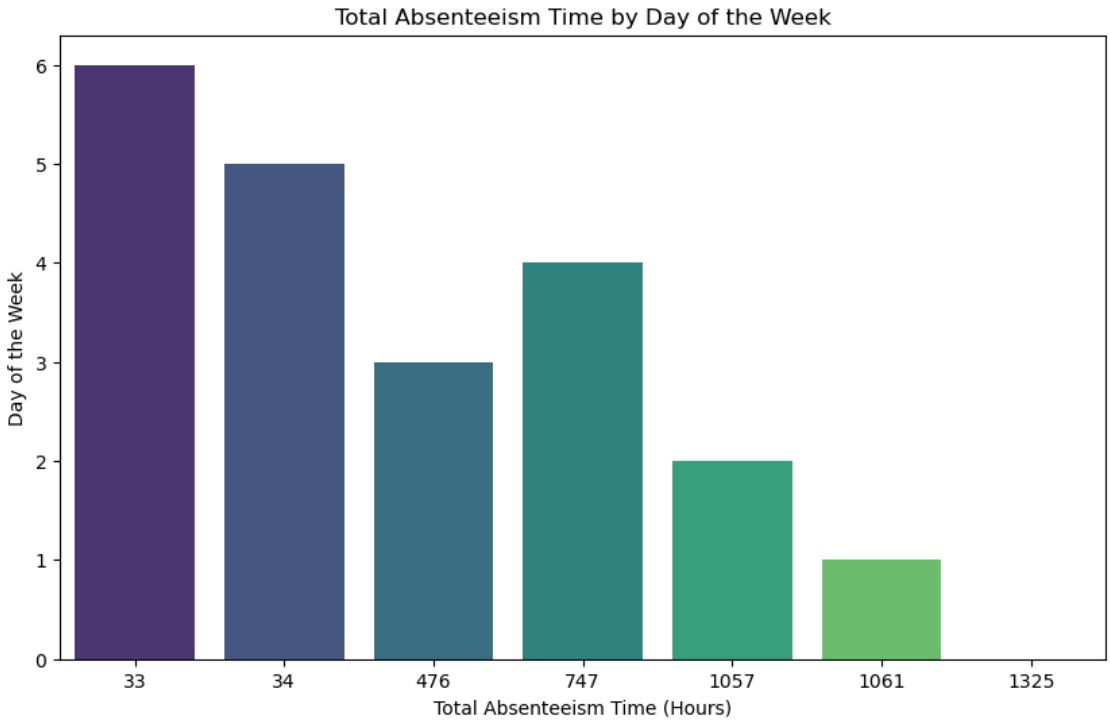
[43]:
```
# exploring for any evidence for preferred days of absence
from scipy import stats


# Count occurrences of absenteeism on each day of the week
day_counts = df['Day of Week'].value_counts().sort_index()

# Perform Chi-Square Test
chi2, p_value = stats.chisquare(day_counts)

print(f"Chi-Square Test Statistic: {chi2}")
print(f"P-Value: {p_value}")

if p_value < 0.05:
    print("There is evidence to suggest absenteeism is not evenly distributed␣
 ↪across days of the week.")
else:
    print("No statistical evidence to suggest a preferred day of the week for␣
 ↪absenteeism.")
```

```
Chi-Square Test Statistic: 251.6
P-Value: 1.8663354205897134e-51
There is evidence to suggest absenteeism is not evenly distributed across days
of the week.
```

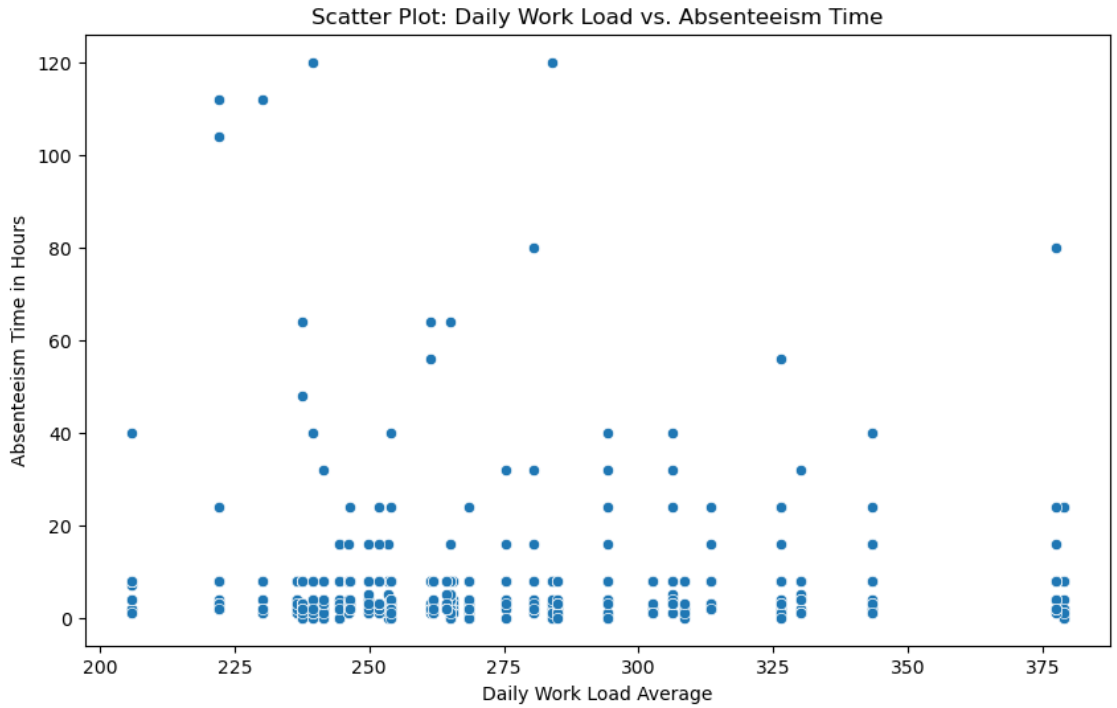
[44]:
```
#Breakdown of absenteeism by day of the week
# Group by 'DayOfWeek' and sum the absenteeism time
absenteeism_by_day = df.groupby('Day of Week')['Absenteeism Time in Hours'].
 ↪sum().reset_index()

# Sort by absenteeism time to identify days with higher absenteeism
absenteeism_by_day = absenteeism_by_day.sort_values(by='Absenteeism Time in␣
 ↪Hours', ascending=False)
```

```python
# Display the results
print(absenteeism_by_day)

# Visualization: Bar chart of absenteeism by day
import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(10, 6))
sns.barplot(x='Absenteeism Time in Hours', y='Day of Week',␣
 ↪data=absenteeism_by_day, palette='viridis')
plt.title('Total Absenteeism Time by Day of the Week')
plt.xlabel('Total Absenteeism Time (Hours)')
plt.ylabel('Day of the Week')
plt.show()
```

	Day of Week	Absenteeism Time in Hours
0	0	1325
1	1	1061
2	2	1057
4	4	747
3	3	476
5	5	34
6	6	33

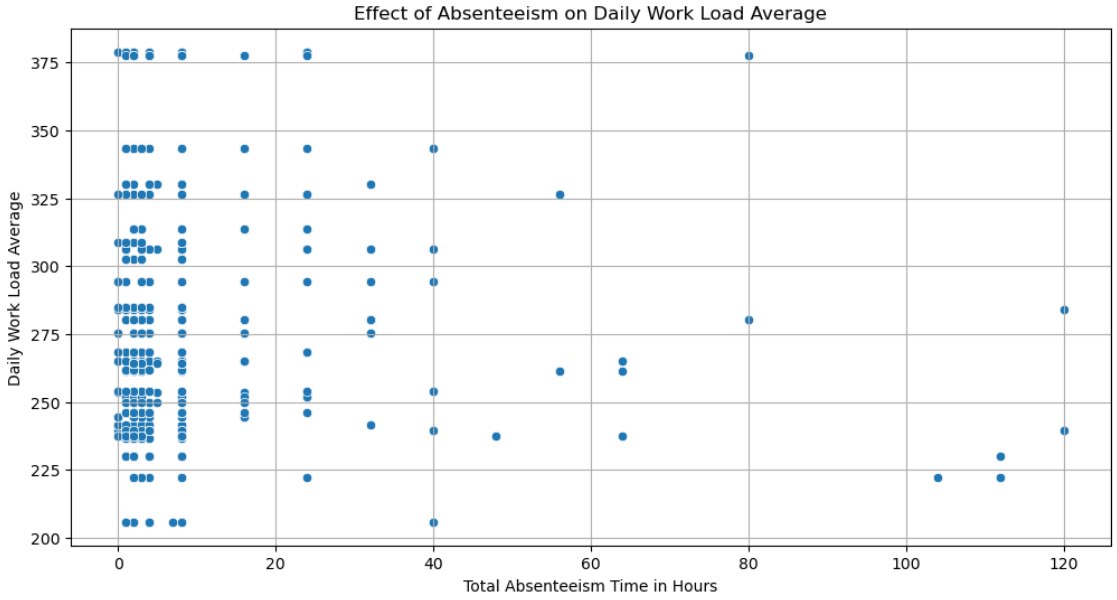


[46]:
```
#exploring if there's a relationship between Daily Work Load Average and␣
 ↪Absenteeism Time in Hours

# Scatter Plot
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Daily Work Load Average', y='Absenteeism Time in Hours',␣
 ↪data=df)
plt.title('Scatter Plot: Daily Work Load vs. Absenteeism Time')
plt.xlabel('Daily Work Load Average')
plt.ylabel('Absenteeism Time in Hours')
plt.show()

# Calculating Correlation Coefficients
pearson_corr = df['Daily Work Load Average'].corr(df['Absenteeism Time in␣
 ↪Hours'], method='pearson')
spearman_corr = df['Daily Work Load Average'].corr(df['Absenteeism Time in␣
 ↪Hours'], method='spearman')

print(f"Pearson Correlation: {pearson_corr}")
print(f"Spearman Correlation: {spearman_corr}")
```




```
Pearson Correlation: 0.029609303294737647
Spearman Correlation: 0.011543655380549979
```

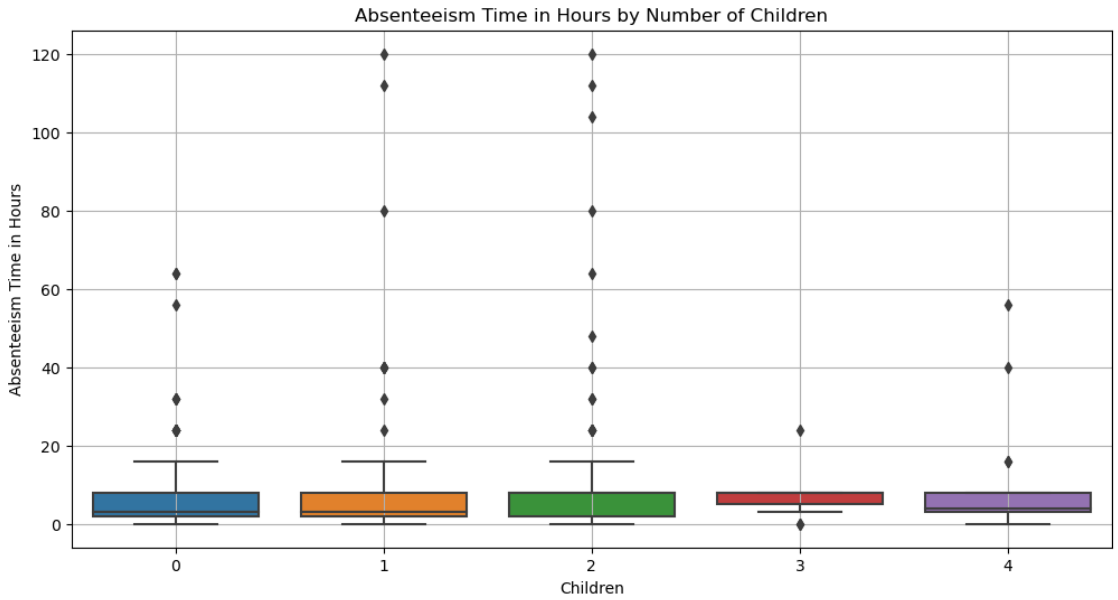
[47]:
```
#  Effect of absenteeism on Daily Work Load Average
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm


# Plotting the relationship
plt.figure(figsize=(12, 6))
sns.scatterplot(data=df, x='Absenteeism Time in Hours', y='Daily Work Load␣
 ↪Average')
plt.title('Effect of Absenteeism on Daily Work Load Average')
plt.xlabel('Total Absenteeism Time in Hours')
plt.ylabel('Daily Work Load Average')
plt.grid()
plt.show()

# Regression Analysis
# Adding a constant for the intercept
X = sm.add_constant(df['Absenteeism Time in Hours'])
y = df['Daily Work Load Average']

# Fit regression model
model = sm.OLS(y, X).fit()


print(model.summary())
```



```
                            OLS Regression Results
==============================================================================
===
Dep. Variable:     Daily Work Load Average   R-squared:
0.001
Model:                                OLS   Adj. R-squared:
-0.001
Method:                     Least Squares   F-statistic:
0.6125
Date:                    Wed, 23 Oct 2024   Prob (F-statistic):
0.434
Time:                            15:07:58   Log-Likelihood:
-3575.0
No. Observations:                     700   AIC:
7154.
Df Residuals:                         698   BIC:
7163.
Df Model:                               1
Covariance Type:                 nonrobust
==============================================================================
=============
                                 coef    std err          t      P>|t|
[0.025      0.975]
------------------------------------------------------------------------------
-------------
const                        271.1694      1.715    158.084      0.000
267.802     274.537
Absenteeism Time in Hours      0.0935      0.120      0.783      0.434
-0.141       0.328
==============================================================================
Omnibus:                       78.666   Durbin-Watson:                   0.049
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              102.969
Skew:                           0.916   Prob(JB):                     4.37e-23
Kurtosis:                       3.417   Cond. No.                         16.3
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

[49]:
```
# analyzing how the number of children affects absenteeism in hours
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm
```

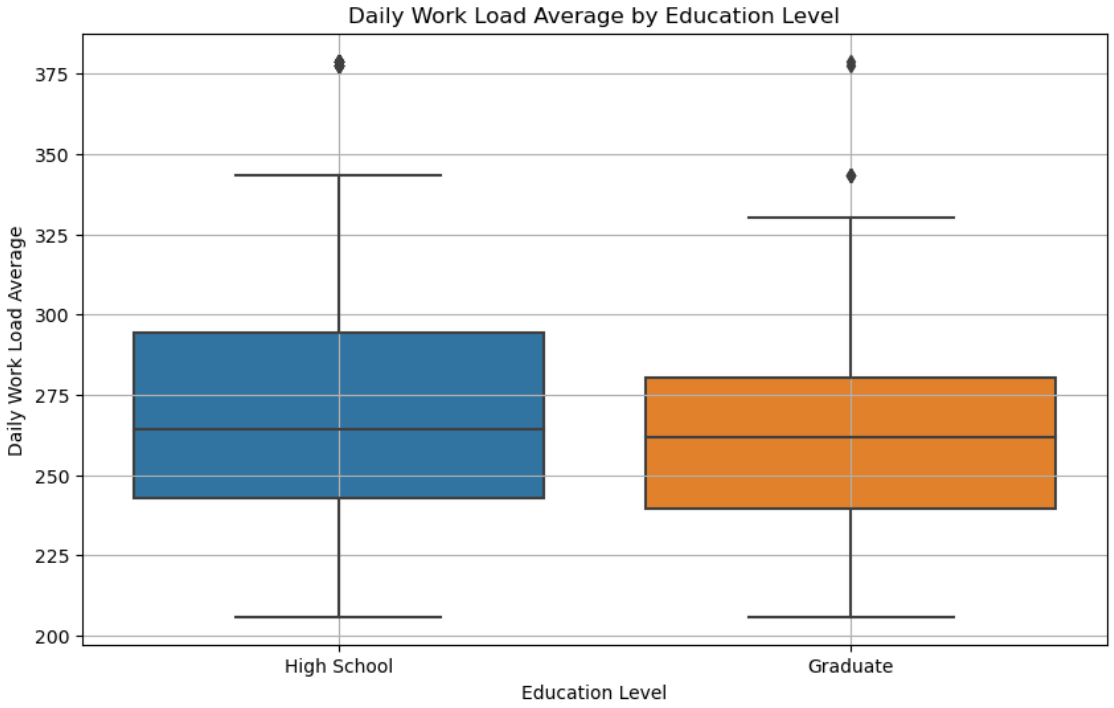
```
# Visualize the relationship
plt.figure(figsize=(12, 6))
sns.boxplot(data=df, x='Children', y='Absenteeism Time in Hours')
plt.title('Absenteeism Time in Hours by Number of Children')
plt.xlabel('Children')
plt.ylabel('Absenteeism Time in Hours')
plt.grid()
plt.show()

# Calculate correlation coefficients
correlation_pearson = df['Children'].corr(df['Absenteeism Time in Hours'],␣
 ↪method='pearson')
correlation_spearman = df['Children'].corr(df['Absenteeism Time in Hours'],␣
 ↪method='spearman')

print(f'Pearson Correlation: {correlation_pearson}')
print(f'Spearman Correlation: {correlation_spearman}')

# Step 3: Regression Analysis
X = sm.add_constant(df['Children'])  # Adding a constant for the intercept
y = df['Absenteeism Time in Hours']

# Fit regression model
model = sm.OLS(y, X).fit()

# Display regression results
print(model.summary())
```



```
Pearson Correlation: 0.09366082776165138
Spearman Correlation: 0.13704004577840043
                            OLS Regression Results
==============================================================================
=====
Dep. Variable:     Absenteeism Time in Hours   R-squared:
0.009
Model:                                 OLS   Adj. R-squared:
0.007
Method:                      Least Squares   F-statistic:
6.177
Date:                     Wed, 23 Oct 2024   Prob (F-statistic):
0.0132
Time:                             15:16:22   Log-Likelihood:
-2767.1
No. Observations:                     700   AIC:
5538.
Df Residuals:                         698   BIC:
5547.
Df Model:                               1
Covariance Type:                 nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          5.6716      0.648      8.752      0.000       4.399       6.944
Children       1.0670      0.429      2.485      0.013       0.224       1.910
==============================================================================
Omnibus:                      810.115   Durbin-Watson:                   2.012
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            51081.824
Skew:                           5.753   Prob(JB):                         0.00
Kurtosis:                      43.237   Cond. No.                         2.56
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

[53]:
```
#analyzing the correlation between education level (grouped into high school␣
 ↪and graduate) and workload
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels.api as sm

# Sample data (replace this with your actual DataFrame)
```

```
# df should contain 'Education Level' (as str) and 'Daily Work Load Average' columns

# Step 1: Convert the 'Education Level' column to numeric
df['Education Level'] = df['Education'].map({'High School': 1, 'Graduate': 2})

# Step 2: EDA - Visualize the relationship
plt.figure(figsize=(10, 6))
sns.boxplot(data=df, x='Education', y='Daily Work Load Average')
plt.title('Daily Work Load Average by Education Level')
plt.xlabel('Education Level')
plt.ylabel('Daily Work Load Average')
plt.xticks(ticks=[0, 1], labels=['High School', 'Graduate'])
plt.grid()
plt.show()

# Step 3: Calculate correlation coefficients
correlation_pearson = df['Education Level'].corr(df['Daily Work Load Average'], method='pearson')
correlation_spearman = df['Education Level'].corr(df['Daily Work Load Average'], method='spearman')

print(f'Pearson Correlation: {correlation_pearson}')
print(f'Spearman Correlation: {correlation_spearman}')

# Step 4: Regression Analysis
X = sm.add_constant(df['Education Level'])  # Adding a constant for the intercept
y = df['Daily Work Load Average']

# Fit regression model
model = sm.OLS(y, X).fit()

# Display regression results
print(model.summary())
```




```
Pearson Correlation: -0.05899037868736083
Spearman Correlation: -0.05549872645004542
                            OLS Regression Results
==============================================================================
===
Dep. Variable:     Daily Work Load Average   R-squared:
0.003
Model:                                OLS   Adj. R-squared:
0.002
Method:                     Least Squares   F-statistic:
2.437
Date:                    Wed, 23 Oct 2024   Prob (F-statistic):
0.119
Time:                            15:37:02   Log-Likelihood:
-3574.1
No. Observations:                     700   AIC:
7152.
Df Residuals:                         698   BIC:
7161.
Df Model:                               1
Covariance Type:                nonrobust
==============================================================================
===
                     coef    std err          t      P>|t|      [0.025
```

```
0.975]
------------------------------------------------------------------------------
---
const              279.1819      4.963     56.255      0.000     269.438
288.926
Education Level     -6.3232      4.050     -1.561      0.119     -14.275
1.629
==============================================================================
Omnibus:                       77.843   Durbin-Watson:                   0.054
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              101.602
Skew:                           0.911   Prob(JB):                     8.66e-23
Kurtosis:                       3.409   Cond. No.                         6.55
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

[57]:
```
df.drop(columns=['Education Level'],inplace=True)
```

[58]:
```
df
```

[58]:

	ID	Reason for Absence	Date	Transportation Expense	\
0	11	Personal	2015-07-07	289	
1	36	Present	2015-07-14	118	
2	3	Personal	2015-07-15	179	
3	7	Health	2015-07-16	279	
4	11	Personal	2015-07-23	289	
..	..	…	…	…	
695	17	Health	2018-05-23	179	
696	28	Health	2018-05-23	225	
697	18	Health	2018-05-24	330	
698	25	Personal	2018-05-24	235	
699	15	Personal	2018-05-31	291	

	Distance to Work	Age	Daily Work Load Average	Body Mass Index	\
0	36	33	239.554	30	
1	13	50	239.554	31	
2	51	38	239.554	31	
3	5	39	239.554	24	
4	36	33	239.554	30	
..	…	…	…	…	
695	22	40	237.656	22	
696	26	28	237.656	24	
697	16	28	237.656	25	
698	16	32	237.656	25	
699	31	40	237.656	25	

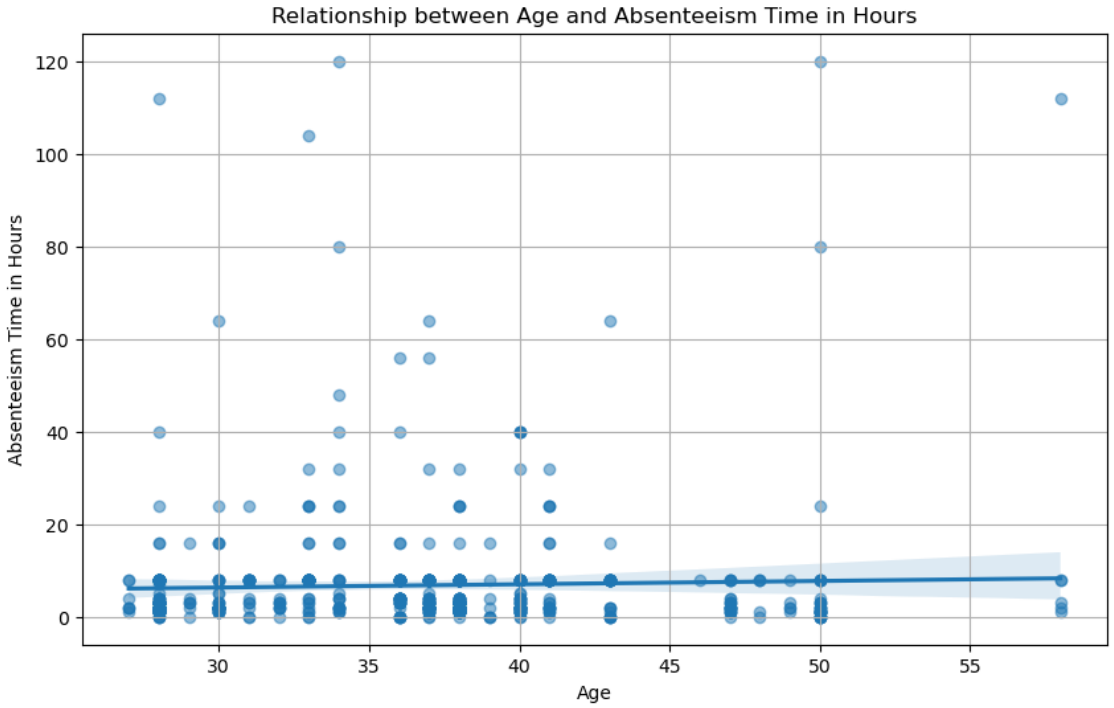
```
       Education  Children  Pets  Absenteeism Time in Hours  Day of Week  \
0    High School         2     1                          4            1
1    High School         1     0                          0            1
2    High School         0     0                          2            2
3    High School         2     0                          4            3
4    High School         2     1                          2            3
..           …       …   …                        …          …
695     Graduate         2     0                          8            2
696  High School         1     2                          3            2
697     Graduate         0     0                          8            3
698     Graduate         0     0                          2            3
699  High School         1     1                          2            3

     Flag_Exceeds_17_Hours
0                        0
1                        0
2                        0
3                        0
4                        0
..                     …
695                      0
696                      0
697                      0
698                      0
699                      0

[700 rows x 14 columns]
```

[61]:
```
#Exploring relationship between age and absenteeism

# Calculate Pearson and Spearman correlations between age and absenteeism
pearson_corr = df['Age'].corr(df['Absenteeism Time in Hours'], method='pearson')
spearman_corr = df['Age'].corr(df['Absenteeism Time in Hours'],␣
 ↪method='spearman')
print("Pearson Correlation:", pearson_corr)
print("Spearman Correlation:", spearman_corr)

#Regression analysis

import statsmodels.api as sm

# Define the dependent (y) and independent (X) variables
X = df['Age']
y = df['Absenteeism Time in Hours']
```

```
# Add a constant to the independent variable (to include the intercept in the␣
 ↪regression)
X = sm.add_constant(X)

# Fit the OLS model
model = sm.OLS(y, X).fit()
print(model.summary())

#Visualizing the relationship

import seaborn as sns
import matplotlib.pyplot as plt

# Scatter plot with regression line
plt.figure(figsize=(10, 6))
sns.regplot(x='Age', y='Absenteeism Time in Hours', data=df,␣
 ↪scatter_kws={'alpha':0.5})
plt.title('Relationship between Age and Absenteeism Time in Hours')
plt.xlabel('Age')
plt.ylabel('Absenteeism Time in Hours')
plt.grid(True)
plt.show()
```

```
Pearson Correlation: 0.035784412437051535
Spearman Correlation: -0.07341726109138923
                            OLS Regression Results
==============================================================================
=====
Dep. Variable:     Absenteeism Time in Hours   R-squared:
0.001
Model:                                 OLS   Adj. R-squared:
-0.000
Method:                      Least Squares   F-statistic:
0.8950
Date:                     Wed, 23 Oct 2024   Prob (F-statistic):
0.344
Time:                             16:40:29   Log-Likelihood:
-2769.8
No. Observations:                     700   AIC:
5544.
Df Residuals:                         698   BIC:
5553.
Df Model:                               1
Covariance Type:                 nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
```

```
const          4.1731      2.778       1.502       0.133      -1.280       9.627
Age            0.0711      0.075       0.946       0.344      -0.076       0.219
==============================================================================
Omnibus:                      808.581   Durbin-Watson:                   1.999
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            50529.557
Skew:                           5.738   Prob(JB):                         0.00
Kurtosis:                      43.009   Cond. No.                         215.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```



0.1 Summary report on absenteeism and related factors analysis

The purpose of the analysis was to investigate the factors that affect absenteeism in a work environment, paying particular emphasis to the relationship between absenteeism and variables including workload, distance to work, educational level, number of children, and preferred days off.

Findings and Recommendations

1. **Correlation Between Workload and Absenteeism**: Findings: -Pearson Correlation: 0.0296 (very weak positive correlation). -Spearman Correlation: 0.0115 (very weak rank correlation). -Regression Analysis: The R-squared value of 0.001 shows that daily workload has no significant impact on absenteeism. A p-value of 0.434 shows the relationship is not

statistically significant. Recommendations: -Monitor Workload: Regularly assess workload distribution among employees to avoid any potential imbalances or overloadssigns of overload that could lead to absenteeism.

2. **Absenteeism and Distance to Work**: Findings: -Pearson Correlation: -0.0806 (indicating a weak negative correlation). -Regression Analysis: The model shows an R-squared value of 0.006. No meaningful relationship between distance to work and absenteeism. A p-value of 0.033 indicates a weak significance, suggesting further investigation may be needed

3. **Absenteeism by Day of the Week**: Findings: -Absenteeism is not evenly distributed across the week, with Mondays showing the highest absenteeism. This suggests a tendency for employees to take leave at the beginning of the week, warranting attention to potential underlying causes. Recommendations: -Investigate Mondays: Conduct surveys or focus groups to understand the reasons for higher absenteeism on at the beginning of the week.

4. **Impact of Number of Children on Absenteeism**: Findings: -Pearson Correlation: 0.0937 (weak positive correlation). -Spearman Correlation**: 0.1370 (indicating a slightly stronger relationship). -Regression Analysis: The p-value of 0.0132 indicates a statistically significant relationship, with an increase of approximately 1.07 hours of absenteeism for each additional child, suggesting that family responsibilities may impact absenteeism. Recommendations: -Family Support Programs: There is a need family support initiatives, such as childcare assistance, flexible scheduling for parents, or family leave policies to help employees manage their responsibilities effectively.

5. **Education Level and Workload**: Findings: -Pearson Correlation: -0.0590 (very weak negative correlation). -Spearman Correlation: -0.0555 (weak rank correlation). -Regression Analysis: The R-squared value of 0.003 shows education level has negligible impact on daily workload. The p- value of 0.119 suggests that this relationship is not statistically significant.

6. **Correlation Between Workload and Absenteeism**: Findings -Pearson Correlation: 0.036 -Spearman Correlation:-0.073 -Pearson's coefficient indicates a very weak positive correlation between age and absenteeism. However, the Spearman's coefficient shows a weak negative correlation, suggesting no meaningful relationship between age and absenteeism.

7. **Regression Analysis**: -R-squared - 0.001, indicating that age explains only 0.1% of the variance in absenteeism time. Coefficient for Age: 0.0711 with a p-value of 0.344, the regression results confirm that age does not significantly affect absenteeism hours.

Conclusion The analysis revealed important areas that require more research and focused efforts to increase employee attendance and overall productivity, offering insightful information on the dynamics of workplace absenteeism. By putting the suggestions into practice, a more encouraging workplace may be developed, which would eventually lower absenteeism and raise employee satisfaction.

[59]:
```
df.to_csv('C:/Users/User/Downloads/Final_Absenteeism-data.csv', index=False)
```