```python
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

import warnings
warnings.filterwarnings('ignore')
```

1.Fetching Data set from Kaggle

```python
# Install dependencies as needed:
# pip install kagglehub[pandas-datasets]
import kagglehub
from kagglehub import KaggleDatasetAdapter

# Set the path to the file you'd like to load
file_path = "global_cancer_patients_2015_2024.csv"

# Load the latest version
df = kagglehub.load_dataset(
  KaggleDatasetAdapter.PANDAS,
  "zahidmughal2343/global-cancer-patients-2015-2024",
  file_path,
)
```

```python
df
```

```python
df.info()
```

```python
# checking the duplicates
df.duplicated().sum()
```

Data Profiling

```
%pip install ydata-profiling --quiet
```

```
from ydata_profiling import ProfileReport

# generate the profile report
profile=ProfileReport(df,title='My Data Profile')

# Save the report as html file
# profile.to_file('Cancer Data Anlysis.html')

# Display the report directly in the notebook
profile.to_notebook_iframe()
```

2.EDA

2.1 Ploting for Age column

Double-click (or enter) to edit

```
plt.figure(figsize=(10,4))

plt.subplot(1,2,1)
sns.kdeplot(df["Age"],fill=True,color='lightgreen')
plt.title("Kde Plot for Age")

plt.subplot(1,2,2)
sns.histplot(df['Age'], bins=30, kde=False, color='cyan')
plt.title("Histgram for Age")

plt.tight_layout()
plt.show()
```

```
df['Age'].describe()
```

Age Remark

Age Remark

Range : 20-89 years

Mean Age : 54.42 years

standard devaition: 20.22

Interquartile Range(IQR): Q1---> 37 Q3---> 72

This suggest that a broad representation of both young and elder patients in the data whisch supports age based camparative anlysis

2.2 Gender Column

```
df['Gender'].value_counts()
```

```
sns.barplot(x =df['Gender'].value_counts().index,
            y = df['Gender'].value_counts().values,
            palette=['SkyBlue','Pink','LightGreen']
            )
for i, v in enumerate(df['Gender'].value_counts()):
  plt.text(i,v, str(v),ha='center',va='bottom')

plt.title("Gender Count")
plt.xlabel("Gender")
plt.ylabel('Count')
plt.show()
```

Gender Remark

- So this data set contains 3 gender category : Male , Female, Others
- With most common being male(16,796)

So Gender Distribution is sufficient to evaluate gender specific survival trends and severity outcomes

2.3 For Country Column

```
country_counts= df['Country_Region'].value_counts()

plt.figure(figsize=(5,5))
plt.pie(x=country_counts.values,
        labels=country_counts.index,
        autopct='%1.1f%%')
plt.title("Country Distribution")
plt.show()
```

Country Remarks

- Patients come from 10 different countries
- with Australia with most patient (5,092)
- Number of datas from each country is almost the same

This diversity enables cross country comparison of cancer outcomes and treatment economics

2.4 Cancer Types

```
df['Cancer_Type'].value_counts()

sns.barplot(x=df['Cancer_Type'].value_counts().index,
            y=df['Cancer_Type'].value_counts().values)

for i,v in enumerate(df['Cancer_Type'].value_counts()):
  plt.text(i,v, str(v), ha='center', va='bottom')

plt.title("cancer type count")
plt.xlabel("Cancer Type")
plt.ylabel("count")

plt.show()
```

Cancer types remarks

- There are 8 types of Cancer

- all have approx same no of dats
- most common is " colon cancer "
- And then Prostate cancer

2.5 Cancer Stages

```
df['Cancer_Stage'].value_counts()

sns.barplot(x=df['Cancer_Stage'].value_counts().index,
            y=df['Cancer_Stage'].value_counts().values)
for i, v in enumerate(df['Cancer_Stage'].value_counts()):
  plt.text(i,v, str(v), ha='center', va='bottom')

plt.title("Cancer Stage Count")
plt.xlabel("Cancer Stage")
plt.ylabel("Count")
```

Cancer Stage Remarks

- There are 5 stages of cancer
- with vlaues ranging from satge 0 to 4
- stage 2 : most common stage (most people are diagonsied in this phase)
- each stage have almost same no of dtas

```
df.info()
```

2.6 Treatment cost

```
# treatment cost vs age
plt.figure(figsize=(10,4))

plt.subplot(1,2,1)
sns.kdeplot(df["Treatment_Cost_USD"], fill=True, color="lightgreen")
```

```
plt.title("KDE plot for Age")

plt.subplot(1,2,2)
sns.histplot(df
["Treatment_Cost_USD"], bins=30,  kde=False, color="cyan")
plt.title("Histogram plot for Age")

plt.tight_layout()
plt.show()
```

```
df["Treatment_Cost_USD"].describe()
```

Treatment cost (usd) remark

- there is no skeewness
- there are almost same number of data points under each bin as observd from histogram

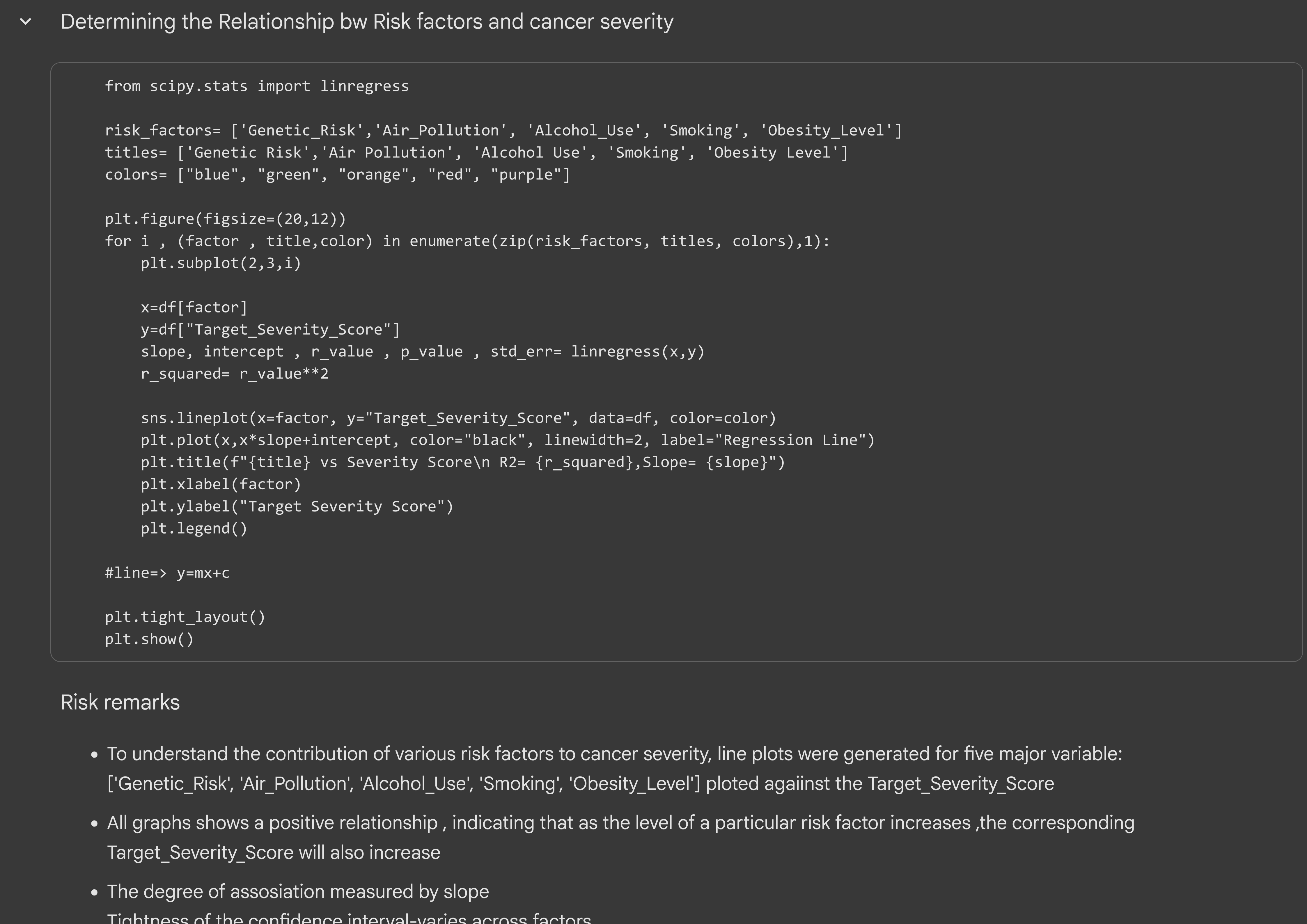
2.7 Analysing The Risk Factor

```
df.info()
```

```
column_of_interest=['Genetic_Risk','Air_Pollution', 'Alcohol_Use', 'Smoking', 'Obesity_Level']

summary= df[column_of_interest].agg(['mean','std','min','max'])
summary
```

Risk Factor Analysis

- All these variable have nearly same means ~ 5 & standard deviation ~ 2.88
- This indicate that they were likely designed on the same standardized scale
- They are essential in studying interaction effects on survival
- Also from the profile data report from the corelation graph we can see that these factors affect the Target_Severity_Score



Determining the Relationship bw Risk factors and cancer severity

```
from scipy.stats import linregress

risk_factors= ['Genetic_Risk','Air_Pollution', 'Alcohol_Use', 'Smoking', 'Obesity_Level']
titles= ['Genetic Risk','Air Pollution', 'Alcohol Use', 'Smoking', 'Obesity Level']
colors= ["blue", "green", "orange", "red", "purple"]

plt.figure(figsize=(20,12))
for i , (factor , title,color) in enumerate(zip(risk_factors, titles, colors),1):
    plt.subplot(2,3,i)

    x=df[factor]
    y=df["Target_Severity_Score"]
    slope, intercept , r_value , p_value , std_err= linregress(x,y)
    r_squared= r_value**2

    sns.lineplot(x=factor, y="Target_Severity_Score", data=df, color=color)
    plt.plot(x,x*slope+intercept, color="black", linewidth=2, label="Regression Line")
    plt.title(f"{title} vs Severity Score\n R2= {r_squared},Slope= {slope}")
    plt.xlabel(factor)
    plt.ylabel("Target Severity Score")
    plt.legend()

#line=> y=mx+c

plt.tight_layout()
plt.show()
```

Risk remarks

- To understand the contribution of various risk factors to cancer severity, line plots were generated for five major variable: ['Genetic_Risk', 'Air_Pollution', 'Alcohol_Use', 'Smoking', 'Obesity_Level'] ploted agaiinst the Target_Severity_Score
- All graphs shows a positive relationship , indicating that as the level of a particular risk factor increases ,the corresponding Target_Severity_Score will also increase
- The degree of assosiation measured by slope
  Tightness of the confidence interval-varies across factors

- Tightness of the confidence interval varies across factors

Relations

- Genetic Risk vs Target_Severity_Score
    - R^2= .23 : A weak linear relationship
    - only 23% of variability in Target_Severity_Score can be explained by Genetic_Risk
    - this means that other factors likely play a larger role in infulencing the Target_Severity_Score
    - slope= 0.20: positive slope means Genetic_Risk Increses the Target_Severity_Score
        - for each unit increase in Genetic Risk ,Target_Severity_Score increses by 0.20 units
        - however the R^2 score is relatively low
- Air Polution vs Target_Severity_Score
    - R^2 = 0.13: A very weak relationship.
        - Only 13% of the variance in Target_Severity_Score can be explained by Air_Pollution
        - meaning that this factor has a limited effect on the target variable.
    - Slope = 0.15: positive slope
        - means that as air pollution increases, the severity score slightly increases
        - But, due to the low R^2, this relationship is weak and unreliable as a predictor for the target severity
- Alcohol Use vs Target_Severity_Score
    - R^2 = 0.13: Weak Relation
        - Only 13% of the variation in the target score is explained by alcohol use
    - Slope = 0.15: positive slope
        - means increased alcohol use correlates with a slight increase in target severity.
        - However, like air pollution, the weak R^2 suggests other factors have a much stronger influence on the target.
- Smoking vs Target_Severity_Score
    - R^2 = 0.23: weak relationship
        - similar to Genetic_Risk

        - similar to Genetic_Risk
        - Smoking explains only 23% of the variance in the target score, leaving the majority of the variation to be explained by other factors
    - Slope = 0.20: positive slope
        - this implies that as smoking increases, the target severity score increases as well
        - This relationship is similar to that of genetic risk, but with a weak linear association (low R^2)
- Obesity Level vs Target Severity Score
    - R^2 = 0.06: weakest relationship among all factors
        - Only 6% of the variation in the target score is explained by obesity level, suggesting that obesity has a minimal effect on the target variable.
    - Slope = 0.10: A positive slope
        - indicating a slight increase in the severity score as obesity level increases.
        - However, due to the very low R^2, this is a weak and unreliable relationship.

Key Takeaways

1. Weak Linear Relationships:
    - The R^2 values for all the risk factors are relatively low , ranging from 0.06 - 0.23
    - this indicates that there is some relationship b/w these risk factor and Target_Severity_Score but it is weak
    - These factors alone dont explain much variation in the Target_Severity_Score
2. Positive slope:
    - All slopes are positive, indicating that each risk factors increases the Target_Severity_Score
    - As R^2 is very low ,the incese in not strongly consistent across all data points
3. Others:
    - Low R^2 values implies that ,some other unmeasured factors likely contributing to the variation in Target_Severity_Score
    - what we took as risk factors are too weak and not reliable predictors on their own

Next Steps

- As most of the risk factor we took were too weak , we should explore other variables or more complex models that could account for more of variation in the Target_Severity_Score
- this colud inculde interactions b/w risk factors,adding new features, or applying more sophisticated regression techniques

2.8 Analyzing the proportion of earlyy stage diagnoses by cancer type

```
df['Cancer_Type'].unique()
```

```
# Lung Cancer

stage_count=df[df['Cancer_Type']=='Lung']['Cancer_Stage'].value_counts()
early_stage_counts=stage_count.get("Stage 0",0)+stage_count.get("Stage I",0)
total_sum=stage_count.sum()
Percentage_1=(early_stage_counts/total_sum)*100
print(f'Proportion of lung cancer diagnosed at stage 0 and I :{Percentage_1}')
```

```
# Leukemia Cancer

stage_count=df[df['Cancer_Type']=='Leukemia']['Cancer_Stage'].value_counts()
early_stage_counts=stage_count.get("Stage 0",0)+stage_count.get("Stage I",0)
total_sum=stage_count.sum()
Percentage_2=(early_stage_counts/total_sum)*100
print(f'Proportion of lung cancer diagnosed at stage 0 and I :{Percentage_2}')
```

```
# Breast Cancer
stage_count=df[df['Cancer_Type']=='Breast']['Cancer_Stage'].value_counts()
early_stage_counts=stage_count.get("Stage 0",0)+stage_count.get("Stage I",0)
total_sum=stage_count.sum()
Percentage_3=(early_stage_counts/total_sum)*100
print(f'Proportion of lung cancer diagnosed at stage 0 and I :{Percentage_3}')
```

```
# Colon Cancer
stage_count=df[df['Cancer_Type']=='Colon']['Cancer_Stage'].value_counts()
early_stage_counts=stage_count.get("Stage 0",0)+stage_count.get("Stage I",0)
```

```
early_stage_counts=stage_count.get("Stage 0",0)+stage_count.get("Stage I",0)
total_sum=stage_count.sum()
Percentage_4=(early_stage_counts/total_sum)*100
print(f'Proportion of lung cancer diagnosed at stage 0 and I :{Percentage_4}')
```

```
# Skin Cancer
stage_count=df[df['Cancer_Type']=='Skin']['Cancer_Stage'].value_counts()
early_stage_counts=stage_count.get("Stage 0",0)+stage_count.get("Stage I",0)
total_sum=stage_count.sum()
Percentage_5=(early_stage_counts/total_sum)*100
print(f'Proportion of lung cancer diagnosed at stage 0 and I :{Percentage_5}')
```

```
# Cervical Cancer
stage_count=df[df['Cancer_Type']=='Cervical']['Cancer_Stage'].value_counts()
early_stage_counts=stage_count.get("Stage 0",0)+stage_count.get("Stage I",0)
total_sum=stage_count.sum()
Percentage_6=(early_stage_counts/total_sum)*100
print(f'Proportion of lung cancer diagnosed at stage 0 and I :{Percentage_6}')
```

```
# Prostate Cancer
stage_count=df[df['Cancer_Type']=='Prostate']['Cancer_Stage'].value_counts()
early_stage_counts=stage_count.get("Stage 0",0)+stage_count.get("Stage I",0)
total_sum=stage_count.sum()
Percentage_7=(early_stage_counts/total_sum)*100
print(f'Proportion of lung cancer diagnosed at stage 0 and I :{Percentage_7}')
```

```
# Liver Cancer
stage_count=df[df['Cancer_Type']=='Liver']['Cancer_Stage'].value_counts()
early_stage_counts=stage_count.get("Stage 0",0)+stage_count.get("Stage I",0)
total_sum=stage_count.sum()
Percentage_8=(early_stage_counts/total_sum)*100
print(f'Proportion of lung cancer diagnosed at stage 0 and I :{Percentage_8}')
```

```
early_stage_proportions = {
    'Lung': Percentage_1,
    'Leukemia': Percentage_2,
    'Breast': Percentage_3,
    'Colon': Percentage_4,
    'Skin': Percentage_5,
    'Cervical': Percentage_6,
    'Prostate': Percentage_7,
    'Liver': Percentage_8
}

early_stage_df = pd.DataFrame.from_dict(early_stage_proportions, orient='index', columns=['Early Stage Diagnoses %'])
early_stage_df = early_stage_df.reset_index().rename(columns={'index': 'Cancer Type'})
display(early_stage_df)
```

Remarks

- Early diagnosis (Stage 0 or Stage I) for different cancers is fairly common, happening for about 38% to 41% of cases
- Liver cancer is the most often caught early, while Lung cancer is the least
- This means our current ways of finding cancer early are working pretty well, but we can do better, especially for Lung cancer
- To find more cancers early and help patients live longer, we should look into better screening tests, starting treatment sooner, and using newer ways to spot cancer
- Since the early diagnosis rates are similar for most cancers, it seems helpful to improve early detection across the board. But we should pay special attention to cancers like lung cancer where the rate is a bit lower

3.Feature Selection

3.1 Indentify the key predictors of cancer severity and survival years

```
df.columns
```

```
features= ["Age", 'Genetic_Risk','Air_Pollution', 'Alcohol_Use', 'Smoking', 'Obesity_Level']

targets = ["Survival_Years","Target_Severity_Score"]

#calculate correlations
pearson_corr= df[features+targets].corr(method="pearson")
spearman_corr= df[features+targets].corr(method="spearman")
```

```
# slice out only thr relationship with target variables
pearson_results= pearson_corr[targets]
spearman_results=spearman_corr[targets]

#combine both
correlation_df= pd.concat([pearson_results,spearman_results], axis=1, keys=["Pearson","Spearman"])
correlation_df
```

4.Data Splting and Model Training

```
# import Library
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import r2_score
```

```
# converting categorical columns to numerical
categorical_cols= ["Gender","Country_Region","Cancer_Type","Cancer_Stage"]
label_encoder= LabelEncoder()

for col in categorical_cols:
    df[col]= label_encoder.fit_transform(df[col])
```

```
# preparing features and input
X= df.drop(columns=['Patient_ID','Survival_Years','Target_Severity_Score','Treatment_Cost_USD'])
y= df['Target_Severity_Score']
```

```
# train test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
# model training
model= RandomForestRegressor(n_estimators=200, max_depth=None, min_samples_leaf=1, random_state=42)
model.fit(X_train,y_train)
```

```
▾ RandomForestRegressor
RandomForestRegressor(n_estimators=200, random_state=42)
```

```
# evaluate the model performance

train_r2_severity= r2_score(y_train , model.predict(X_train))
test_r2_severity= r2_score(y_test , model.predict(X_test))

print(train_r2_severity)
print(test_r2_severity)
```

```
# feature importance

feature_importance_severity= pd.Series(model.feature_importances_, index= X.columns, ).sort_values(ascending=True)
feature_importance_severity
```

```
# plotting of important features
plt.figure(figsize=(10,6))
sns.barplot(x=feature_importance_severity, y= feature_importance_severity.index)
plt.title('Feature Importance for Target Severity Score(Random Forest)')
plt.xlabel("Feature Importance")
plt.ylabel("Features")
```

Feature Importance Interpretation

- Smoking (0.2336):
  - The most important factor in predicting cancerseverity
  - Higher smoking levels are linked to higher severity scores.
- Genetic Risk (0.2286):
  - A strong genetic pre-disposition is almost insignificant as smoking in influencing severity
- Treatment Cost (USD) (0.2133):
  - Higher treatment costs tend to be associated with more severe cancer conditions
- Alcohol Use (0.1291):

- Alcohol Use (0.1291):
    - Alcohol consumption also has a notable impact oncancer severity.
- Air Pollution (0.1271):
    - This environmental factor is important, withhigher pollution levels linked to worse severity scores
- Obesity Level (0.0573):
    - While it has an effect, obesity has a muchsmaller impact on severity compared to other factors
- Age and Gender (< 0.01):
    - These factors have very low importance and donot explain much of the variation in severity scores.

In summary, Smoking, Genetic Risk, Treatment Cost, Alcohol Use, and Air Pollution are the major factors influencing cancer severity. This information can help identify areas where interventions could potentially reduce severity.

4.2 Taking Important features

```
# Random Forest for target severity score
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.preprocessing import  LabelEncoder
from sklearn.metrics import r2_score
```

```
# converting categorical column to numerical column
categorical_cols= ["Gender","Country_Region","Cancer_Type","Cancer_Stage"]
label_encoder= LabelEncoder()

for col in categorical_cols:
    df[col]= label_encoder.fit_transform(df[col])
```

```
# feature imput
X= df.drop(columns=['Patient_ID','Survival_Years','Target_Severity_Score','Treatment_Cost_USD'])
y= df['Target_Severity_Score']
```

```
# train test split
# train test split
X_train, X_test, y_train,y_test= train_test_split(X,y, test_size=0.2, random_state=42)
```

```
param_grid={
    'n_estimators':[100,200],
    'max_depth':[None,5,10],
    'min_samples_split': [2, 5],
    'min_samples_leaf':[1,2]

}
```

```
# train model
model= RandomForestRegressor(random_state=40)
GSC= GridSearchCV(model, param_grid, cv=3, scoring='r2', n_jobs=-1)
GSC.fit(X_train,y_train)
```

```
best_rf_severity= GSC.best_estimator_
```

```
# model evaluation
train_r2_severity= r2_score(y_train, best_rf_severity.predict(X_train))
test_r2_severity= r2_score(y_test, best_rf_severity.predict(X_test))

print(train_r2_severity)
print(test_r2_severity)
```

checking relation with survival years

```
import seaborn as sns
sns.histplot(df['Survival_Years'], kde=True)
```

```
df.corr(numeric_only=True)['Survival_Years'].sort_values(ascending=True)
```

- The information for survial years doesnt help model to figure out how long someone survive

5. Exploring the economic burden of cancer treatment across different demographics and countries

```
# Install dependencies as needed:
# pip install kagglehub[pandas-datasets]
import kagglehub
from kagglehub import KaggleDatasetAdapter

# Set the path to the file you'd like to load
file_path = "global_cancer_patients_2015_2024.csv"

# Load the latest version
df = kagglehub.load_dataset(
  KaggleDatasetAdapter.PANDAS,
  "zahidmughal2343/global-cancer-patients-2015-2024",
  file_path,
)
```

```
df
```

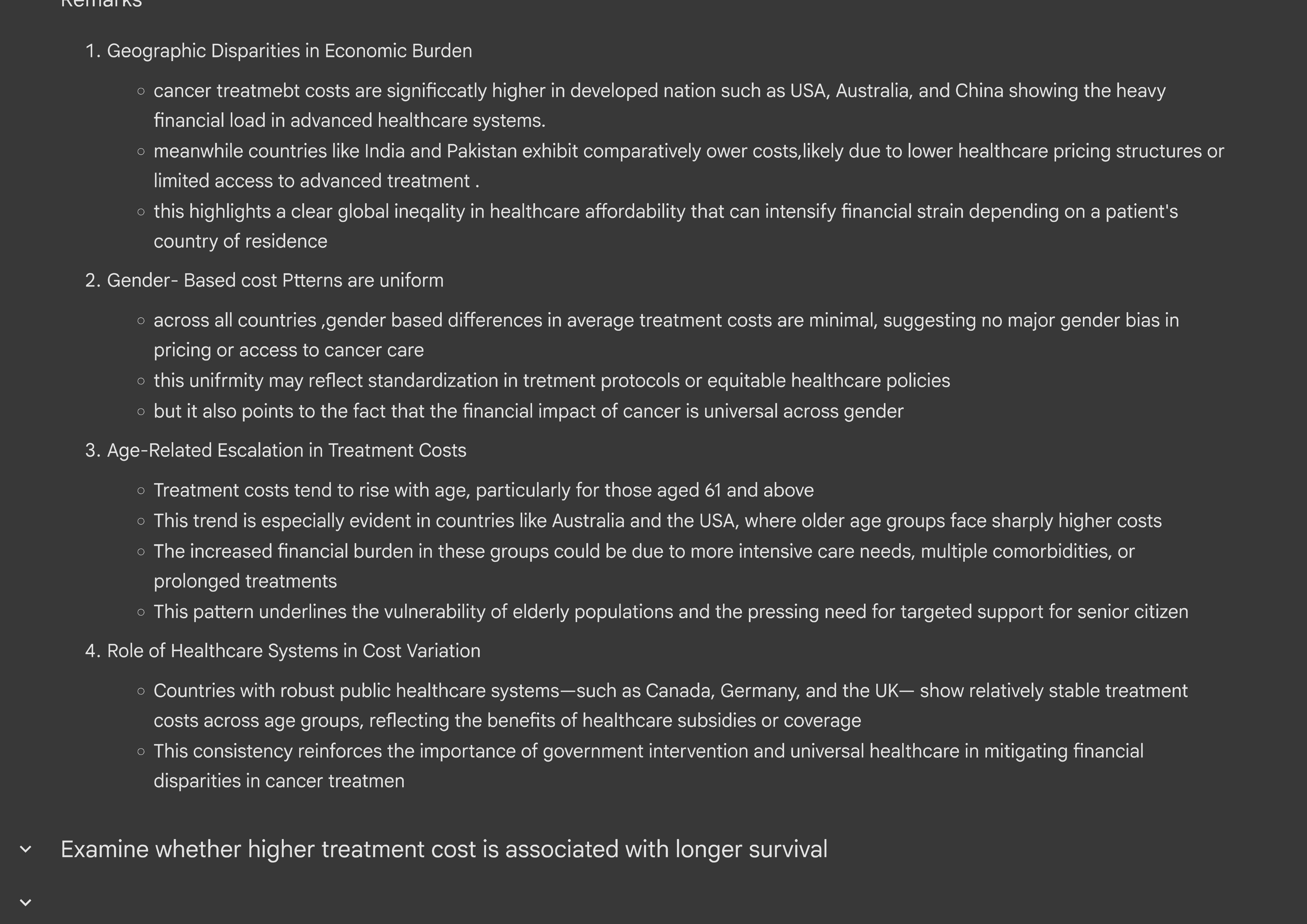
```
df['Age_Group']= pd.cut(df['Age'], bins=[0,30,45,60,75,100], labels=['0-30', '31-45','46-60','61-75','76+'])
```

```
country_age_cost= df.groupby(['Country_Region', 'Age_Group','Gender'])

plt.figure(figsize=(10,6))
sns.barplot(data=df, x='Country_Region', y='Treatment_Cost_USD', hue='Gender', estimator='mean')
plt.title("Average cancer treatment cost by country and gender")
plt.show
```

```
country_age_cost= df.groupby(['Country_Region', 'Age_Group'])["Treatment_Cost_USD"].mean().reset_index()
```

```
heatmap_data= country_age_cost.pivot(index='Age_Group', columns='Country_Region', values='Treatment_Cost_USD')

plt.figure(figsize=(10,6))
sns.heatmap(heatmap_data, annot=True, fmt='.0f')
plt.title("Average treatment cost by age group and country")
plt.show()
```

Remarks



Remarks

1. Geographic Disparities in Economic Burden
    - cancer treatmebt costs are significcatly higher in developed nation such as USA, Australia, and China showing the heavy financial load in advanced healthcare systems.
    - meanwhile countries like India and Pakistan exhibit comparatively ower costs,likely due to lower healthcare pricing structures or limited access to advanced treatment .
    - this highlights a clear global ineqality in healthcare affordability that can intensify financial strain depending on a patient's country of residence
2. Gender- Based cost Ptterns are uniform
    - across all countries ,gender based differences in average treatment costs are minimal, suggesting no major gender bias in pricing or access to cancer care
    - this unifrmity may reflect standardization in tretment protocols or equitable healthcare policies
    - but it also points to the fact that the financial impact of cancer is universal across gender
3. Age-Related Escalation in Treatment Costs
    - Treatment costs tend to rise with age, particularly for those aged 61 and above
    - This trend is especially evident in countries like Australia and the USA, where older age groups face sharply higher costs
    - The increased financial burden in these groups could be due to more intensive care needs, multiple comorbidities, or prolonged treatments
    - This pattern underlines the vulnerability of elderly populations and the pressing need for targeted support for senior citizen
4. Role of Healthcare Systems in Cost Variation
    - Countries with robust public healthcare systems—such as Canada, Germany, and the UK— show relatively stable treatment costs across age groups, reflecting the benefits of healthcare subsidies or coverage
    - This consistency reinforces the importance of government intervention and universal healthcare in mitigating financial disparities in cancer treatmen

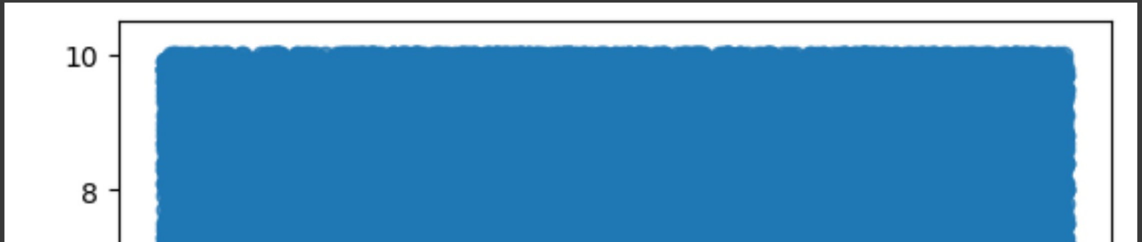
Examine whether higher treatment cost is associated with longer survival

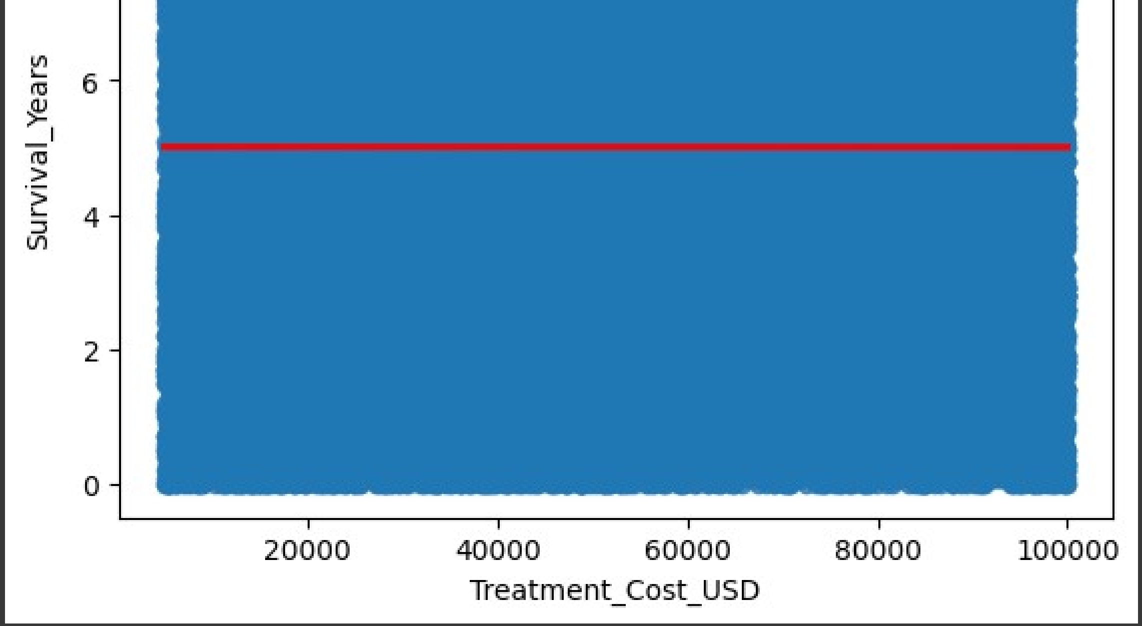
- Null Hypothesis(Ho) : There is no correlation between treatment cost and survival years
- Alternative Hypothesis(H1) : There is a correlation (positive or negetive) between treatmentcost and survival years

```
Start coding or generate with AI.
```

```
from scipy.stats  import pearsonr, spearmanr
x= df["Treatment_Cost_USD"]
y=df["Survival_Years"]

# performing pearson correlation test
pearson_corr, pearson_p = pearsonr(x,y)
print(f"Pearson Correlation Coefficient: {pearson_corr}")
print(f"Pearson P-Value : {pearson_p}")

# Spearman correlation test
spearman_corr, spearman_p = spearmanr(x,y)
print(f"spearman Correlation Coefficient: {spearman_corr}")
print(f"spearman P-Value : {spearman_p}")

alpha=0.05

def interpret_corr(corr, p, method):
    if p<alpha:
        print(f"{method}, we  reject thr hull hypothesis")
    else:
        print(f"{method}, we failed to reject null hypothesis")

interpret_corr(pearson_corr,pearson_p,"Pearson")
interpret_corr(spearman_corr,spearman_p,"Spearman")
```

```
sns.regplot(x=x, y=y, line_kws={"color":"red"})
plt.show()
```





Remarks

- there is no relation between Treatment cost and survival years

Examine if higher cancer stages lead to greater treatment costs and reducede survival years

```
df_new= df.copy()
```

```
stage_order=['Stage 0','Stage I','Stage II','Stage III','Stage IV']
```

```
grouped_stats= df_new.groupby("Cancer_Stage")[["Treatment_Cost_USD", "Survival_Years"]].mean().reset_index()
grouped_stats
```

Remarks

1. Treatment cost vs Cancer stage
    - Null hypothesis(Ho) : The avg treatment cost is same across all cancer stages

- Alternative Hypothesis(H1) : At least one stage has a different average cost

2. Survival Years vs. Cancer Stage

    - Null Hypothesis (H_0): The average survival years are the same across all cancer stages.
    - Alternative Hypothesis (H_1): At least one stage has a different survival duration.

```
grouped_costs=[]
grouped_survival=[]



for  stage in stage_order:
    stage_data= df[df["Cancer_Stage"]==stage]
    cost= stage_data["Treatment_Cost_USD"]
    survival= stage_data["Survival_Years"]
    grouped_costs.append(cost)
    grouped_survival.append(survival)
```

```
len(grouped_costs)
```

```
# check for normaility
from scipy.stats import  shapiro, f_oneway
normal_cost=0
normal_survival=0

for i in range (len(stage_order)):
    cost_p= shapiro(grouped_costs[i]).pvalue
    surv_p= shapiro(grouped_survival[i]).pvalue
    print(f" cost {cost_p} for group {i}")
    print(f"Survival {surv_p} for group {i}")
    if cost_p<0.05:
        normal_cost+=1
    if surv_p<0.05:
        normal_survival+=1
```

```
print(normal_cost)
print(normal_survival)
```

```
from scipy.stats import kruskal
```

```
kusrkal_cost =kruskal(*grouped_costs)
kurkal_survival= kruskal(*grouped_survival)

p_cost= kusrkal_cost.pvalue
p_survival= kurkal_survival.pvalue
```

```
print(f'p_cost     :',{p_cost})
print(f'p_survival :',{p_survival})
```

Remarks

- Kruskal-Wallis Test: Treatment Cost across Cancer Stages
  - P-value: 0.4254
  - Conclusion: No significant difference in treatment costs among cancer stages.
- Kruskal-Wallis Test: Survival Years across Cancer Stages
  - P-value: 0.6033
  - Conclusion: No significant difference in survival years among cancer stages.

Examine whether higher genetic risk amplifies the negative effects of smoking on cancer severity

- Null Hypothesis (H_0): The interaction effect between genetic risk and smoking on cancer Severity is not significant. (Genetic risk does not amplify or alter the effect of smoking.)
- Alternative Hypothesis (H_1): The interaction effect between genetic risk and smoking on cancer severity is significant. (Genetic risk does amplify or alter the effect of smoking.)

```
import statsmodels.formula.api as smf

model = smf.ols("Target_Severity_Score ~ Genetic_Risk*Smoking", data=df).fit()
model = smf.ols("Target_Severity_Score ~ Genetic_Risk*Smoking", data=df).fit()

model.summary2().tables[1].loc["Genetic_Risk:Smoking"]
```

```
p_value=0.628255
```

Interpretation:

- The interaction coefficient is negative but very small: -0.000228
- The p-value = 0.628, which is greater than 0.05, so we fail to reject the null hypothesis

🧪 Conclusion (Statistical):

✅ Test Used: Multiple Linear Regression with interaction term

📌 Null Hypothesis (H_0): No interaction effect between genetic risk and smoking

📌 Alternative Hypothesis (H_1): There is an interaction effect

- The interaction effect between Genetic Risk and Smoking on Target Severity Score is not statistically significant (p = 0.628 > 0.05).
- This means that based on your data, there is no evidence that Genetic Risk amplifies or reduces the effect of Smoking on the Target Severity Score.

In other words, smoking and genetic risk may each have independent effects (or none), but they do not interact in a way that significantly changes the outcome.