Staffing Standards: analysing the impact of the healthcare workforce count on healthcare quality in Singapore

1 Background, importance and objectives

1.1 Background and importance

The quality of public healthcare is essential to a nation's wellbeing, and thus the healthcare workforce is a critical area of study. Singapore boasts one of the strongest healthcare systems in the world, ranking first in the Legatum Prosperity Index in terms of health [1]. While much attention has been given to policies and funding, how the healthcare workforce count affects the healthcare quality remains underexplored despite numerous efforts to bolster the numbers in the healthcare workforce.

As Singapore experiences a rapidly aging population, with nearly 1 in 5 citizens aged 65 and above [2], it is imperative we focus on workforce adequacy to meet the needs of an aging population. Examining how workforce dynamics influence public healthcare quality over time can pinpoint the gaps in Singapore's healthcare system.

The research combines healthcare workforce count data with healthcare quality indicators. By appliyng data analysis techniques, I will investigate trends and correlations to understand the relationship between staffing levels and public healthcare quality in Singapore.

1.2 Objectives

By the end of this project, I intend to achieve:

1. Thorough analysis of the relationship between the healthcare workforce and the quality of public healthcare in Singapore.
2. Identification of trends in staffing levels across key healthcare professions like nurses and therapists, and their correlation with healthcare quality indicators over time.
3. Evaluation of the adequacy of workforce growth in meeting Singapore's healthcare demands in the context of evolving healthcare needs

The requirements to achieve the objectives:

1. Data collection and preparation
    - Gather and process historical data on the healthcare workforce count in Singapore for allied health professionals.

    - Gather and process data on healthcare quality indicators in Singapore.
2. Trend Analysis
    - Analyse staffing trends across different healthcare professions over time.
    - Identify changes in healthcare quality and align changes with the workforce dynamics.
3. Insights
    - Investigate correlations between workforce trends and public healthcare quality indicators.
    - Explore possible causal links and identify key factors driving changes in healthcare quality.
    - Provide insights into long term effects.
4. Reflections
    - Reflect on the limitations of the dataset and analysis techniques used.
    - Discuss potential areas for further research.

1.3 Research Questions and Hypotheses

Outlined key research questions to guide the analysis:

1. How have the healthcare workforce count changed over time across key healthcare professions?
2. Is there a measurable relationship between changes in workforce allocation and healthcare quality indicators?
3. What long-term trends or impacts can be projected from the current data?

Proposed hypotheses:

1. There exists a negative correlation between the increase in healthcare workers entering the private sector and the worsening healthcare quality indicators.
2. Workforce imbalances across professions lead to disparities in healthcare quality outcomes.

2 Data and methodology

2.1 Datasets Overview

This section will describe key datasets used in the research, including sources, time range and content.

Healthcare workforce data is sourced from https://data.gov.sg/ for the following professions:

- Advanced Practice Nurses
- Nurses and Midwives

- Occupational Therapists
- Physiotherapists
- Speech Therapists
- Pharmacists

Healthcare quality data is based on the Healthcare Access and Quality (HAQ) Index sourced from The Lancet Global Health and Institute for Health Metrics and Evaluation.

Analysis of these datasets will provide a comprehensive basis for exploring the link between the healthcare workforce and healthcare quality in Singapore

2.2 Analytical Framework and Tools

The analytical process and tools used in this paper include:

1. Data processing - Cleaning datasets for missing, inconsistent or invalid entries, ensuring compatibility for analysis. Selecting necessary data and excluding unnecessary data.
2. Exploratory Data Analysis - Visualising workforce count and healthcare quality trends using libraries such as pandas, numpy and matplotlib.
3. Webscraping - Obtain data from HTML elements using Selenium for elements loaded by Javascript

2.3 Limitations and constraints of data

The varying time intervals of the data collected represent a significant limitation in analysing the impact of healthcare workforce count trends on healthcare quality.

2.3.1 Inconsistent Data Granularity

Considering the data intervals for the HAQ Index is irregular while the data intervals for the healthcare workforce count is yearly, it can lead to challenges in aligning trends across different datasets. The irregular data intervals for the HAQ Index mean that we are unable to observe the immediate impact of the healthcare workforce on the healthcare quality for certain years.

To combat this limitation, only the years that are present across all data sets will be used for this research.

2.3.2 Delayed Reporting

The mismatch in the data intervals, where healthcare workforce count and healthcare quality data are collected at different time points, it could lead to significant challenges in interpreting the relationship between the healthcare workforce count and healthcare quality. This inconsistency may cause the analysis to overlook long term impacts.

This limitation can be countered by considering the immediate effects of the healthcare workforce count on the healthcare quality in Singapore.

2.4 Licensing of data

All data used in this research project is open for non-commercial uses from data.gov.sg [3] and Institute for Health Metrics and Evaluation [4]

2.5 Ethical considerations

Data in the datasets used are completely anonymous and mitigates any biases as well as avoids harmful assumptions.

The findings of this project do not aim to crticise Singapore's healthcare system but rather understand how the healthcare workforce count impacts the qualty of healthcare in Singapore.

2.6 Guidelines for reuse

Individuals intending to use the source data must comply with the specific terms and conditions established by each publishing entity at the time of use and independently secure the necessary permissions where applicable.

3 Analysis of workforce trends

This section will explore the trends and implications of the healthcare workforce. Datasets from the healthcare workforce for these professions were obtained from the data.gov.sg website.

The first part of section 3 will initialise the libraries used in order to import the CSV files and print out a graph plot for the analysed data. Following that, analysis of the different healthcare professions and their numbers will take place. The first profession group would be nurses which include advanced practice nurses, nurses and midwives. The next profession group would be therapists which include speech therapists, occupational therapists and physiotherapists.

The steps for analysing the counts for nursing profession group are
Step 1: Import the CSV containing the data
Step 2: Separate the profession counts into sectors - public, private and inactive
Step 3: According to the sectors, plot a graph of healthcare workforce count against the years

Step 4: Observe and understand the graph trends, finding further research on possible causes for the trends

The steps for analysing the counts for therapists profession group are
Step 1: Import the CSV containing the data
Step 2: Separate the profession counts into sectors - public, private and inactive
Step 3: Once all professions (speech therapists, occupational therapists and physiotherapists) have been seperated into sectors, combine each data set for the seperated sectors
Step 4: According to the combine sectors, plot a graph of healthcare workforce count against the years
Step 5: Observe and understand the graph trends, finding further research on possible causes for the trends

All datasets used for this section are in the form of CSV files and will be imported and processed based on the requirements of the sub-sections.

3.1 Importing data from CSV files

To import data from a CSV file, we require the pandas library.
To install the pandas library, we need to run the following code in a terminal:
"pip install pandas"
We will import the pandas library as "pd" for cleaner code formatting with the following code:
"import pandas as pd"

In [39]:
```
import pandas as pd
```

To plot and graph the dataframe, we require the matplotlib and searborn library. matplotlib is a plotting library with sufficient functionality for the purposes of this research. seaborn is a interface data visualisation library based on matplotlib for the purpose of accessibility.

To install the pandas library, we need to run the following code in a terminal:
"pip install matplotlib"
"pip install seaborn"
We will import the matplotlib library as "plt" and seaborn as "sns" for cleaner code formatting with the following code:
"import matplotlib.pyplot as plt"
"import seaborn as sns"

In [42]:
```
import matplotlib.pyplot as plt
import seaborn as sns

# reset colour palette to colour-blind friendly for accessibility
sns.reset_orig()
```

```
palette = sns.color_palette("colorblind")
plt.style.use("seaborn-v0_8-colorblind")
```

3.2 Advanced practice nurses, nurses and midwives

3.2.1 Advanced practice nurses

Import and display "NumberofAdvancedPracticeNurses.csv" using pandas library

In [297…
```
advanced_practice_nurses_df = pd.read_csv("NumberofAdvancedPracticeNurses.csv")

# for the purposes of shortening the output, print the head of the dataframe
advanced_practice_nurses_df.head(5)
```

Out[297…

	year	sector	count
0	2008	Public Sector	15
1	2008	Private Sector	2
2	2008	Not in Active Practice	0
3	2009	Public Sector	35
4	2009	Private Sector	2

Assign each sector to independent variables for graph plotting

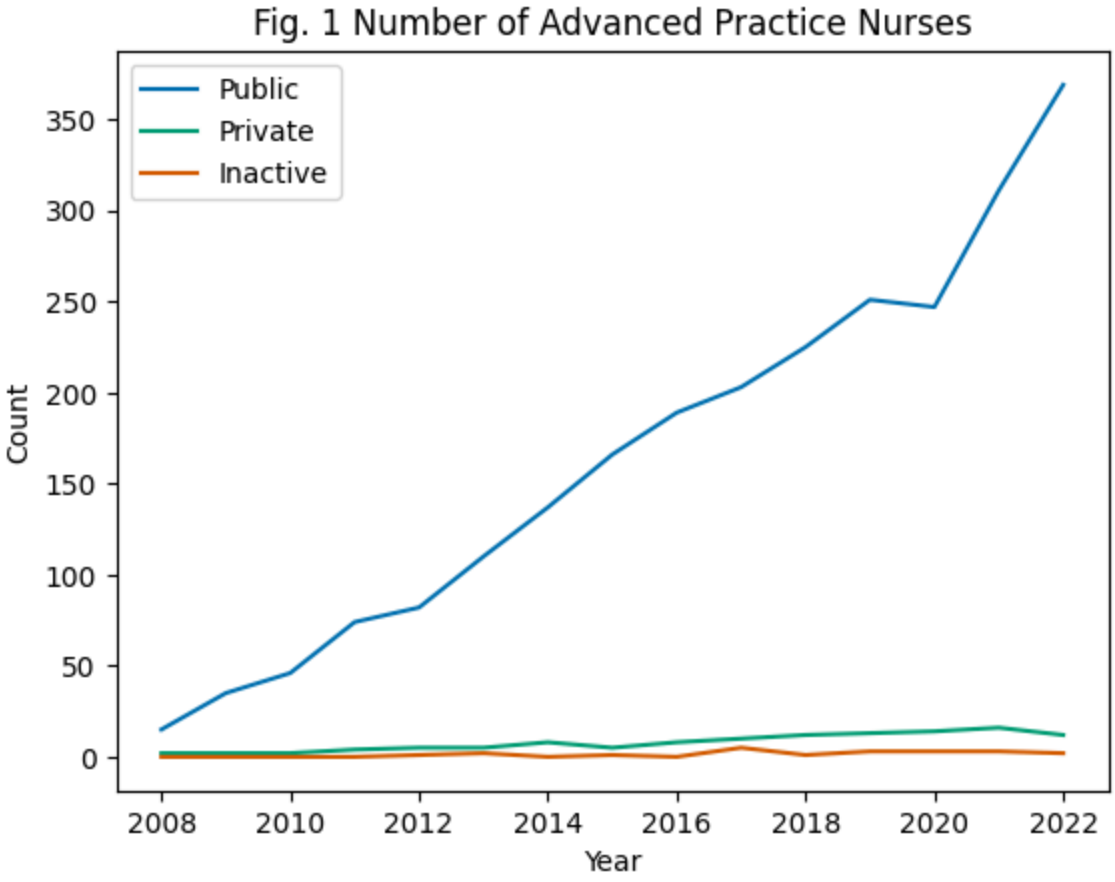
In [48]:
```
apn_public_sector = advanced_practice_nurses_df[advanced_practice_nurses_df['sector
apn_private_sector = advanced_practice_nurses_df[advanced_practice_nurses_df['secto
apn_inactive = advanced_practice_nurses_df[advanced_practice_nurses_df['sector'] ==
```

Plot dataframe into a Line Plot using matplotlib library

In [50]:
```
fig, advanced_practice_nurses_plot = plt.subplots()
advanced_practice_nurses_plot.plot(apn_public_sector['year'],apn_public_sector['cou
advanced_practice_nurses_plot.plot(apn_private_sector['year'],apn_private_sector['c
advanced_practice_nurses_plot.plot(apn_inactive['year'],apn_inactive['count'], labe
advanced_practice_nurses_plot.legend()

advanced_practice_nurses_plot.set_xlabel("Year")
advanced_practice_nurses_plot.set_ylabel("Count")
advanced_practice_nurses_plot.set_title("Fig. 1 Number of Advanced Practice Nurses"
```

Out[50]:
```
Text(0.5, 1.0, 'Fig. 1 Number of Advanced Practice Nurses')
```



3.2.2 Nurses and midwives

Import and display "NumberofNursesandMidwives.csv" using pandas library

In [293…
```
nurses_midwives_df = pd.read_csv("NumberofNursesandMidwives.csv")

# for the purposes of shortening output, print the head of dataframe
nurses_midwives_df.head(5)
```

Out[293…

	year	type	sector	count
0	2006	Registered Nurses	Public Sector	8495
1	2006	Registered Nurses	Private Sector	4566
2	2006	Registered Nurses	Not in Active Practice	2391
3	2006	Enrolled Nurses	Public Sector	2956
4	2006	Enrolled Nurses	Private Sector	1484

Assign each sector to independent variables for graph plotting.
Group different types of nurses and midwives by sector instead of type.

In [295…
```
nm_public_sector = nurses_midwives_df[nurses_midwives_df['sector'] == 'Public Secto
nm_public_counts = nm_public_sector.groupby(['year', 'sector'],as_index=False)['cou
```

```
# for the purposes of shortening output, print the head of dataframe
nm_public_counts.head(5)
```

Out[295…

	index	year	sector	count
0	0	2006	Public Sector	11574
1	1	2007	Public Sector	12294
2	2	2008	Public Sector	13711
3	3	2009	Public Sector	15675
4	4	2010	Public Sector	17613

In [299…

```
nm_private_sector = nurses_midwives_df[nurses_midwives_df['sector'] == 'Private Sec
nm_private_counts = nm_private_sector.groupby(['year', 'sector'],as_index=False)['c
# for the purposes of shortening output, print the head of dataframe
nm_private_counts.head(5)
```

Out[299…

	index	year	sector	count
0	0	2006	Private Sector	6109
1	1	2007	Private Sector	6112
2	2	2008	Private Sector	6224
3	3	2009	Private Sector	6463
4	4	2010	Private Sector	6965

In [301…

```
nm_inactive = nurses_midwives_df[nurses_midwives_df['sector'] == 'Not in Active Pra
nm_inactive_counts = nm_inactive.groupby(['year', 'sector'],as_index=False)['count'
# for the purposes of shortening output, print the head of dataframe
nm_inactive_counts.head(5)
```

Out[301…

	index	year	sector	count
0	0	2006	Not in Active Practice	3244
1	1	2007	Not in Active Practice	3926
2	2	2008	Not in Active Practice	4274
3	3	2009	Not in Active Practice	4654
4	4	2010	Not in Active Practice	4762

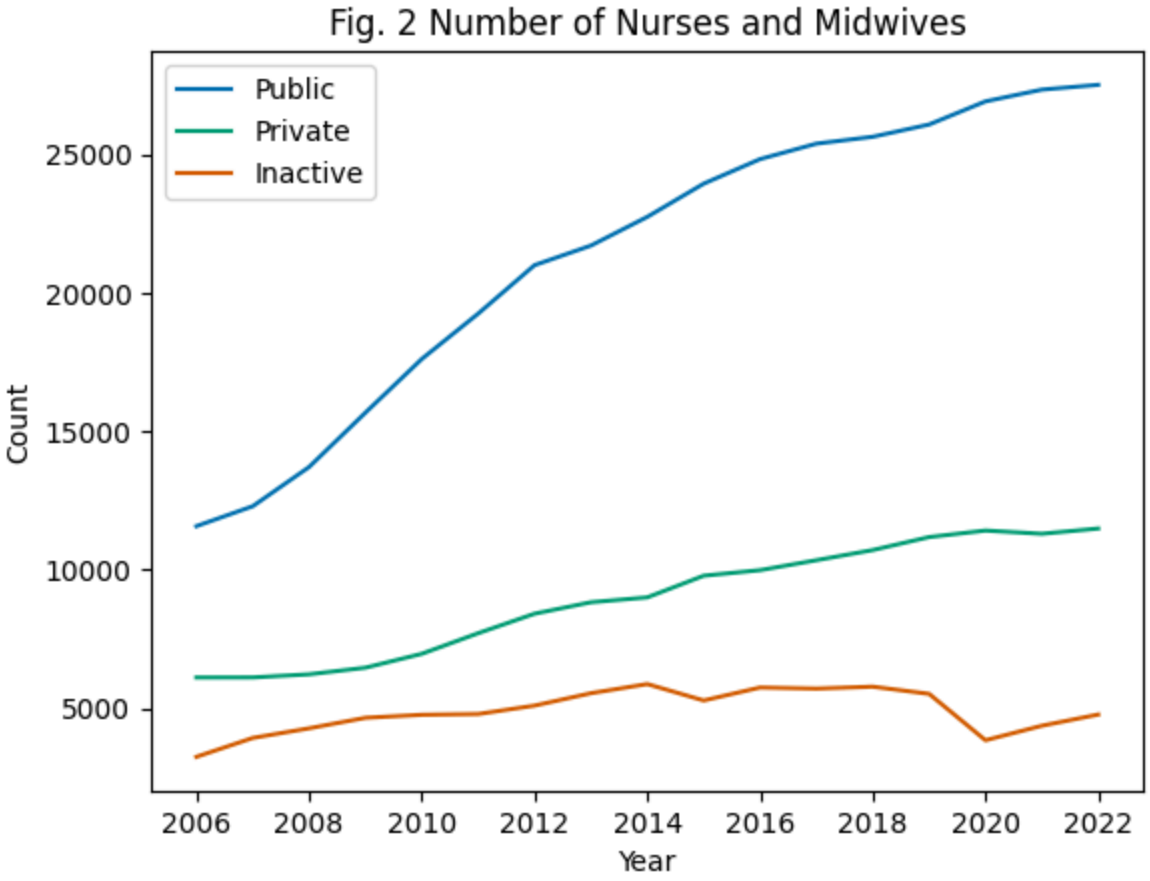
Plot dataframe into a Line Plot using matplotlib library

In [59]:

```
fig, nurses_midwives_plot = plt.subplots()
nurses_midwives_plot.plot(nm_public_counts['year'],nm_public_counts['count'], label
nurses_midwives_plot.plot(nm_private_counts['year'],nm_private_counts['count'], lab
nurses_midwives_plot.plot(nm_inactive_counts['year'],nm_inactive_counts['count'], l
nurses_midwives_plot.legend()
```

```
nurses_midwives_plot.set_xlabel("Year")
nurses_midwives_plot.set_ylabel("Count")
nurses_midwives_plot.set_title("Fig. 2 Number of Nurses and Midwives")
```

Out[59]: Text(0.5, 1.0, 'Fig. 2 Number of Nurses and Midwives')



3.2.3 Analysis of Advanced practice nurses, nurses and midwives

Analysis of the plotted line graphs reveals that the public sector has experienced a significantly greater increase in the number of advanced practice nurses, as well as registered nurses and midwives, compared to the private sector over the observed period. This trend suggests a more substantial expansion of the public healthcare workforce, potentially reflecting targeted policies or resource allocation favoring public sector healthcare services.

Further findings show that there was increased attention towards the nursing sector with new avenues of pursuing a career in nursing opened up. [5][6]

Besides the increased focus on the general career of nursing, more benefits have been provided to the nurses in the public sector. [7] This would contribute to the great rise in nurses in the public sector compared to the private sector.

3.3 Therapists

3.3.1 Speech Therapists

Import and display "NumberofSpeechTherapists.csv" using pandas library.

In [308…
```
speech_therapists_df = pd.read_csv("NumberofSpeechTherapists.csv")

# for purposes of shortening output, print the head of the dataframe
speech_therapists_df.head(5)
```

Out[308…

	year	sector	count
0	2014	Public Sector	193
1	2014	Private Sector	207
2	2014	Not in Active Practice	0
3	2015	Public Sector	212
4	2015	Private Sector	244

Assign each sector to independent variables for graph plotting.

In [69]:
```
st_public_sector = speech_therapists_df[speech_therapists_df['sector'] == 'Public S
st_private_sector = speech_therapists_df[speech_therapists_df['sector'] == 'Private
st_inactive = speech_therapists_df[speech_therapists_df['sector'] == 'Not in Active
```

3.3.2 Occupational Therapists

Import and display "NumberofOccupationalTherapists.csv" using pandas library.

In [330…
```
occupational_therapists_df = pd.read_csv("NumberofOccupationalTherapists.csv")

# for purposes of shortening output, print the head of the dataframe
occupational_therapists_df.head(5)
```

Out[330…

	year	sector	count
0	2014	Public Sector	501
1	2014	Private Sector	403
2	2014	Not in Active Practice	1
3	2015	Public Sector	498
4	2015	Private Sector	429

Assign each sector to independent variables for graph plotting.

In [73]:
```
ot_public_sector = occupational_therapists_df[occupational_therapists_df['sector']
ot_private_sector = occupational_therapists_df[occupational_therapists_df['sector']
ot_inactive = occupational_therapists_df[occupational_therapists_df['sector'] == 'N
```

3.3.3 Physiotherapists

Import and display "NumberofPhysiotherapists.csv" using pandas library.

In [332…
```
physiotherapists_df = pd.read_csv("NumberofPhysiotherapists.csv")

# for purposes of shortening output, print the head of the dataframe
physiotherapists_df.head(5)
```

Out[332…

	year	sector	count
0	2014	Public Sector	861
1	2014	Private Sector	531
2	2014	Not in Active Practice	2
3	2015	Public Sector	877
4	2015	Private Sector	606

Assign each sector to independent variables for graph plotting.

In [77]:
```
pt_public_sector = physiotherapists_df[physiotherapists_df['sector'] == 'Public Sec
pt_private_sector = physiotherapists_df[physiotherapists_df['sector'] == 'Private S
pt_inactive = physiotherapists_df[physiotherapists_df['sector'] == 'Not in Active P
```

Combine the dataframes for public, private and inactive, matching the years and summing the counts

3.3.4 Therapists in public sector

In [334…
```
combined_public_df = pd.concat([st_public_sector,ot_public_sector,pt_public_sector]
# print(combined_public_df)
combined_public_df.head(5)
```

Out[334…

	year	sector	count
0	2014	Public Sector	193
3	2015	Public Sector	212
6	2016	Public Sector	215
9	2017	Public Sector	232
12	2018	Public Sector	246

In [336…
```
sum_public_df = combined_public_df.groupby(['year','sector'],as_index=False)['count
sum_public_df.head(5)
```

Out[336…

	year	sector	count
0	2014	Public Sector	1555
1	2015	Public Sector	1587
2	2016	Public Sector	1635
3	2017	Public Sector	1626
4	2018	Public Sector	1745

3.3.5 Therapists in private sector

In [338…
```
combined_private_df = pd.concat([st_private_sector,ot_private_sector,pt_private_sec
combined_private_df.head(5)
```

Out[338…

	year	sector	count
1	2014	Private Sector	207
4	2015	Private Sector	244
7	2016	Private Sector	269
10	2017	Private Sector	307
13	2018	Private Sector	319

In [340…
```
sum_private_df = combined_private_df.groupby(['year','sector'],as_index=False)['cou
sum_private_df.head(5)
```

Out[340…

	year	sector	count
0	2014	Private Sector	1141
1	2015	Private Sector	1279
2	2016	Private Sector	1428
3	2017	Private Sector	1580
4	2018	Private Sector	1652

3.3.6 Inactive Therapists

In [342…
```
combined_inactive_df = pd.concat([st_inactive,ot_inactive,pt_inactive])
combined_inactive_df.head(5)
```

Out[342...

	year	sector	count
2	2014	Not in Active Practice	0
5	2015	Not in Active Practice	18
8	2016	Not in Active Practice	40
11	2017	Not in Active Practice	56
14	2018	Not in Active Practice	77

In [344...

```
sum_inactive_df = combined_inactive_df.groupby(['year','sector'],as_index=False)['c
sum_inactive_df.head(5)
```

Out[344...

	year	sector	count
0	2014	Not in Active Practice	3
1	2015	Not in Active Practice	124
2	2016	Not in Active Practice	221
3	2017	Not in Active Practice	328
4	2018	Not in Active Practice	413

Plot dataframes for therapists in a Line Plot

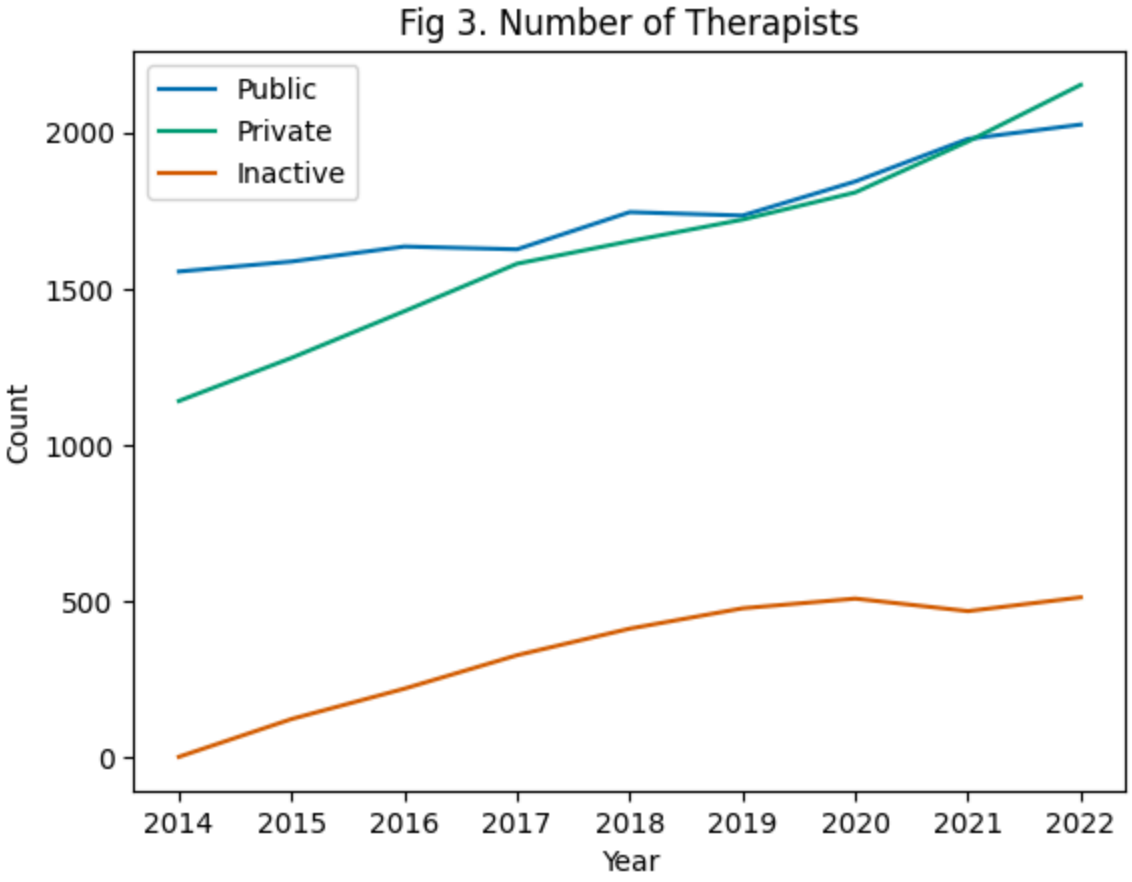
In [89]:

```
fig, therapists_plot = plt.subplots()
therapists_plot.plot(sum_public_df['year'],sum_public_df['count'], label = 'Public'
therapists_plot.plot(sum_private_df['year'],sum_private_df['count'], label = 'Priva
therapists_plot.plot(sum_inactive_df['year'],sum_inactive_df['count'], label = 'Ina
therapists_plot.legend()

therapists_plot.set_xlabel("Year")
therapists_plot.set_ylabel("Count")
therapists_plot.set_title("Fig 3. Number of Therapists")
```

Out[89]:

```
Text(0.5, 1.0, 'Fig 3. Number of Therapists')
```



3.3.7 Analysis of Speech Therapists, Occupational Therapists and Physiotherapists

Analysis of the plotted graph lines reveal that the private sector experienced a greater increase in the number of therapists across the three occupations compared to the public sector. This trend suggests a lack of focus on the public sector for therapists as the therapists prefer working in the private sector.

It is unclear what is the reason for this preference as no concrete evidence could be found to explain this difference. We can speculate multiple reasons like work-life balance or salary difference, but we cannot confirm these reasons. Not enough focus has been placed in these allied health sectors and as such the reasons remain inconclusive.

The fact that not enough focus is placed on these professions could also be proof that this occupation is not as recognised or supported in the healthcare world.

4 Exploring Healthcare Quality in Singapore

This section will explore the trends and implications of healthcare quality in Singapore. Datasets for the healthcare quality in Singapore were obtained from Institute for Health Metrics and Evaluation.

Webscraping will be used to import datasets used for this section and processed based on the requirements of the sub-sections.

The first part of section 4 will initialise the libraries used in order to import the HTML pages, including javascsript loaded elements, and parsing its contents. Following that, analysis of the healthcare quality index will take place.

The steps for analysing the healthcare quality in Singapore are
Step 1: Import the HTML page into text using requests for healthcare quality data for 1990 to 2016 and for 2019
Step 2: Parse the elements using BeautifulSoup
Step 3: Use Selenium and browser drivers to search for and obtain elements in the page loaded by scripts
Step 4: Create a CSV to store the healthcare quality according to the years and extract a dataframe from the new CSV file
Step 5: According to the years, plot a graph of healthcare quality index against the years
Step 6: Observe and understand the graph trends, finding further research on possible causes for the trends

4.1 Extract data for Healthcare Access and Quality Index (HAQ Index)

4.1.1 Importing data from HTML pages

To import data from an HTML page, we require the requests library.
We will import the requests library with the following code:
"import requests"

In [99]:
```
import requests
```

Using the function "requests.get()", we can input a URL to the function to obtain the HTML page.
We then use the function ".text" to convert the HTML page into text form.

In [376…
```
HAQ_1990_to_2016_url = "https://www.thelancet.com/action/showFullTableHTML?isHtml=t
HAQ_1990_to_2016_HTML = requests.get(HAQ_1990_to_2016_url)
HAQ_1990_to_2016_text = HAQ_1990_to_2016_HTML.text
print(HAQ_1990_to_2016_HTML.text[:500]) #print first 500 characters for cleaner out
```

```
<!DOCTYPE html><html lang="en-US"><head><title>Just a moment...</title><meta http-eq
uiv="Content-Type" content="text/html; charset=UTF-8"><meta http-equiv="X-UA-Compati
ble" content="IE=Edge"><meta name="robots" content="noindex,nofollow"><meta name="vi
ewport" content="width=device-width,initial-scale=1"><style>*{box-sizing:border-box;
margin:0;padding:0}html{line-height:1.15;-webkit-text-size-adjust:100%;color:#31313
1;font-family:system-ui,-apple-system,BlinkMacSystemFont,Segoe UI,Roboto,Helvetic
```

4.1.2 Parsing HTML pages

To parse the HTML page, we require BeautifulSoup. BeautifulSoup is Python package to parse HTML and XML documents.
The purpose of using BeautifulSoup is to create a parse tree useful for web scraping, navigating, searching and modifying.
We will import BeautifulSoup from bs4 with the following code:
"from bs4 import BeautifulSoup"

In [104…
```
from bs4 import BeautifulSoup
```

Using BeautifulSoup's HTML parser, we are able to generate a clean parse tree of the HTML elements.

In [378…
```
HAQ_1990_to_2016_soup = BeautifulSoup(HAQ_1990_to_2016_HTML.text,'html.parser')
print(HAQ_1990_to_2016_soup.prettify()[2000:2200]) #print section using script
```

```
to continue
     </span>
    </div>
   </noscript>
  </div>
 </div>
 <script>
  (function(){window._cf_chl_opt={cvId: '3',cZone: "www.thelancet.com",cType: 'mana
ged',cRay: '8f9acb779f12823d',c
```

We note that the HTML elements of the table is loaded by Javascript.
The use of Selenium is needed here to extract the Javascript information using appropriate browser drivers.

4.1.3 Importing data from Javascript loaded elements using Selenium

Selenium is used to automate browsers. We will use Selenium with browser drivers to obtain Javascript loaded elements.
It is important to note that a stable internet connection is required to load the pages.

To install Selenium, we first have to use the following command in a terminal:
"pip install selenium"

Next we need to import the necessary modules from Selenium using the following:

from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

In [356…
```
from selenium import webdriver
from selenium.webdriver.common.by import By
```

```
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
```

The last step would be identifying which drivers are needed to execute Selenium. This will be according to the user's browser. Below is the corresponding drivers needed for the appropriate browsers.
Chrome → ChromeDriver
Firefox → GeckoDriver
Edge → EdgeDriver

For convenience purposes, all drivers have been downloaded and stored together in the folder.

Now that Selenium is installed and set up, we need to set up the drivers according to the browsers.

Firefox

Set up the Selenium WebDriver for Firefox

In [361…
```
from selenium.webdriver.firefox.service import Service

gecko_driver_path = './geckodriver'
ffox_service = Service(gecko_driver_path)
```

Google Chrome (Chromium)

Set up the Selenium WebDriver for Chromium browsers

In [119…
```
# from selenium.webdriver.chrome.service import Service

# chrome_driver_path = './chromedriver'
# chrome_service = Service(chrome_driver_path)
```

Microsoft Edge

Set up the Selenium WebDriver for Microsoft Edge

In [121…
```
# from selenium.webdriver.edge.service import Service

# edge_driver_path = './msedgedriver.exe'
# edge_service = Service(edge_driver_path)
```

If the Jupyter Notebook is open in Firefox, run the cell for Firefox.
If the Jupyter Notebook is open in Google Chrome or Chromium based browsers, run the cell for Chromium browsers.
If the Jupyter Notebook is open in Microsoft Edge, run the cell for Microsoft Edge.

For convenience purposes, the Jupyter Notebook is open in Firefox and will be run in Firefox.

4.1.4 Importing HAQ Index data for 1990 to 2016

Open the HTML page and wait for Javascript elements to load then extract necessary information for HAQ Index from 2000 to 2016

Use exception handling of "try" and "finally" to avoid any complications while the driver is running.

In [380…

```
# If browser is Firefox run:
driver = webdriver.Firefox(service=ffox_service)

## If browser is Chromium run:
# driver = webdriver.Chrome(service=chrome_service)

## If browser is Microsoft Edge run:
# driver = webdriver.Edge(service=edge_service)
try:
    driver.get("https://www.thelancet.com/action/showFullTableHTML?isHtml=true&tabl
    HAQ_1990_to_2016_table = driver.execute_script("return document.querySelector('

    # Wait for the table to load and the element with the classes 'scrollable borde
    WebDriverWait(driver, 60).until(EC.presence_of_element_located((By.CSS_SELECTOR
    HAQ_1990_to_2016_table_soup = BeautifulSoup(HAQ_1990_to_2016_table, 'html.parse

    print(HAQ_1990_to_2016_table_soup.prettify()[:300]) #print first 300 characters
finally:
    driver.quit()
```

```
<table border="0" cellpadding="0" cellspacing="0" class="scrollable bordered" tabind
ex="0" width="100%">
 <thead>
  <tr>
   <th>
   </th>
   <th>
   </th>
   <th>
   </th>
   <th class="rowsep" colspan="3">
    <strong>
     HAQ Index (95% UI)
    </strong>
   </th>
   <th class="rowsep" colspan="3"
```

4.1.5 Importing HAQ Index data for 2019

Open the HTML page and wait for Javascript elements to load then extract necessary information for HAQ Index for 2019.

Use exception handling of "try" and "finally" to avoid any complications while the driver is running.

In [369…
```
# If browser is Firefox run:
driver = webdriver.Firefox(service=ffox_service)

## If browser is Chromium run:
# driver = webdriver.Chrome(service=chrome_service)

## If browser is Microsoft Edge run:
# driver = webdriver.Edge(service=edge_service)
try:
    driver.get("https://www.thelancet.com/action/showFullTableHTML?isHtml=true&tabl
    HAQ_2019_table = driver.execute_script("return document.querySelector('.scrolla

    # Wait for the table to load and the element with the classes 'scrollable borde
    WebDriverWait(driver, 60).until(EC.presence_of_element_located((By.CSS_SELECTOR
    HAQ_2019_table_soup = BeautifulSoup(HAQ_2019_table, 'html.parser') # Use Beauti

    print(HAQ_2019_table_soup.prettify()[:300]) #print first 300 characters for cle
finally:
    driver.quit()
```

```
<table border="0" cellpadding="0" cellspacing="0" class="scrollable bordered" tabind
ex="0" width="100%">
 <thead>
  <tr>
   <th>
   </th>
   <th>
   </th>
   <th class="rowsep" colspan="4">
    <strong>
     2019 HAQ Index (95% UI)
    </strong>
   </th>
   <th class="rowsep" colspan="4">
    <stron
```

4.2 Assign extracted data to new table and csv

4.2.1 Years

The first step would be locating the row for the 'Year' header and extracting only the years '1990', '2000' and '2016'.

In [130…
```
rows_1990_to_2016 = HAQ_1990_to_2016_table_soup.find_all('tr') # Assign all rows wi
year_row = None
for row in rows_1990_to_2016:
    # Check if '1990' is in any of the rows in the HTML page for
    if '1990' in row.text:
        # Print row where '1990' is located and assign to variable 'year_row'
```

```
        print(row)
        year_row = row
        break # Stop iteration once the row is found
```

```
<tr>
<th class="rowsep"></th>
<th class="rowsep"></th>
<th class="rowsep"></th>
<th class="rowsep">1990</th>
<th class="rowsep">2000</th>
<th class="rowsep">2016</th>
<th class="rowsep">1990–2016</th>
<th class="rowsep">1990–2000</th>
<th class="rowsep">2000–16</th>
<th class="rowsep">1990–2016</th>
<th class="rowsep">1990–2000</th>
<th class="rowsep">2000–16</th>
</tr>
```

From the row, we can see that the years '1990', '2000' and '2016' are in the first three non-empty elements of tag 'th'. We iterate through all elements of tag 'th' and assign only '1990', '2000' and '2016' to a list.

In [132…
```
# Assign each individual cell to list 'year_cells'
year_cells = year_row.find_all('th')
four_year_cells = [] # Initialise empty list to store 4 years
# Extract the index from the first 3 non-empty cells
for cell in year_cells[0:6]: # Iterate through first 6 cells to locate first 3 non-
    if cell.text != '': # Check that the cell is not empty
        four_year_cells.append(int(cell.text)) # Add cell to the list
print("Four year cells:", four_year_cells) # Print to check that the append has suc
```

```
Four year cells: [1990, 2000, 2016]
```

The fourth year to be added is '2019'. However, in both HTML pages, the string '2019' does not appear in its own cell. As such, we have to manually add the year '2019' to the list of 'Four year cells' using '.append'

In [134…
```
four_year_cells.append(2019)
print("Four year cells:", four_year_cells) # Print to check that the append has suc
```

```
Four year cells: [1990, 2000, 2016, 2019]
```

4.2.2 HAQ Index

The next step would be locating the row containing 'HAQ Index' values for Singapore only. We start with the first HTML page.

In [137…
```
sg_row_1990_to_2016 = None
for row in rows_1990_to_2016:
    # Check if 'Singapore' is in any of the table cells (td)
    if 'Singapore' in row.text:
        # Print row where 'Singapore' is located and assing to variable 'sg_row'
        print(row)
```

```
        sg_row_1990_to_2016 = row
        break # Stop iteration once the row is found
```

```
<tr>
<td></td>
<td></td>
<td>Singapore</td>
<td>69·2 (66·5 to 72·0)</td>
<td>79·7 (77·2 to 82·0)</td>
<td>90·6 (87·2 to 93·3)</td>
<td>21·4 (17·5 to 25·0)<sup>*</sup></td>
<td>10·5 (7·1 to 13·9)<sup>*</sup></td>
<td>10·9 (7·1 to 14·8)<sup>*</sup></td>
<td>1·04 (0·85 to 1·21)<sup>*</sup></td>
<td>1·41 (0·95 to 1·88)<sup>*</sup></td>
<td>0·80 (0·53 to 1·08)<sup>*</sup></td>
</tr>
```

From the row, we can see that the HAQ Index values from the years '1990', '2000' and '2016' are in the first three elements of tag 'th' after the cell containing 'Singapore'. We iterate through all elements of tag 'th' and assign only the HAQ Index values for the years '1990', '2000' and '2016' to a list.

In [139…
```
sg_cells_1990_to_2016 = sg_row_1990_to_2016.find_all('td') # Assign all cells to a
sg_haq_cells = []
# Extract the text from the first 4 cells
for cell in sg_cells_1990_to_2016[0:6]: # Iterate through first 6 cells to locate f
    if cell.text != '' and cell.text != 'Singapore': # Check if cell is not empty a
        # Replace dot in string with decimal separator '.' and convert to float bef
        sg_haq_cells.append(float(cell.text[0:4].replace('·','.')))
print("Singapore HAQ cells:", sg_haq_cells)
```

```
Singapore HAQ cells: [69.2, 79.7, 90.6]
```

Here we execute 'cell.text[0:4]' to obtain the first 4 characters in the selected cell. This removes the bracketed content in the cell, which is the range for the HAQ Index value.

Next, we execute '.replace('·','.')' to obtain a string that can be converted into a float. From the print of 'sg_row_1990_to_2016', we see that the decimal point is not represented by a numerical dot. This prevents the conversion of the string into a float. To counter this, we convert the decimal point into the actual numerical dot.

Finally we use 'float()' to convert the string into a float. The previous two parts ensure that the string obtained from the cell is able to be converted into a float value. The execution of 'float()' should not face any issues.

We move on to the second HTML page. We intend to extract only 1 HAQ Index value from this page.

In [142…
```
rows_2019 = HAQ_2019_table_soup.find_all('tr') # Assign all rows with tag 'tr' to a
sg_row_2019 = None
```

```
for row in rows_2019:
    # Check if 'Singapore' is in any of the table cells (td)
    if 'Singapore' in row.text:
        # Print row where 'Singapore' is located and assing to variable 'sg_row'
        print(row)
        sg_row_2019 = row
        break # Stop iteration once the row is found
```

```
<tr>
<td></td>
<td>Singapore</td>
<td>86·2 (84·4 to 87·6)</td>
<td>91·6 (89·8 to 93·0)</td>
<td>86·0 (83·5 to 88·0)</td>
<td>80·6 (77·9 to 83·0)</td>
<td>25·2 (23·3 to 27·0)</td>
<td>20·7 (18·4 to 22·9)</td>
<td>25·9 (23·3 to 28·2)</td>
<td>25·5 (22·3 to 28·4)</td>
</tr>
```

From the row, we can see that the HAQ Index value for the year '2019' is in the first element of tag 'th' after the cell containing 'Singapore'. We iterate through all elements of tag 'th' and assign only the HAQ Index value for the year '2019' to the 'sg_haq_cells' list.

In [144…
```
sg_cells_2019 = sg_row_2019.find_all('td') # Assign all cells to a list
# Extract the text from the first 4 cells
for cell in sg_cells_2019[0:3]: # Iterate through first 3 cells to locate the cell
    if cell.text != '' and cell.text != 'Singapore': # Check if cell is not empty a
        # Replace dot in string with decimal separator '.' and convert to float bef
        sg_haq_cells.append(float(cell.text[0:4].replace('·','.')))
print("Singapore HAQ cells:", sg_haq_cells)
```

```
Singapore HAQ cells: [69.2, 79.7, 90.6, 86.2]
```

The same code is executed before adding the HAQ Index value for '2019' into the list of all HAQ Index values for Singapore.

Before uploading the data we have collected to a new csv file, we pair the data to its corresponding year using the 'zip()' function.

In [147…
```
zipped_year_to_haq = zip(four_year_cells,sg_haq_cells)
```

4.2.3 Writing to new CSV file

In the next section, we require the 'csv' module for python which allows file reading and writing.
We can import this module by using the code "import csv".

In [150…
```
import csv
```

We create a new csv file and name it 'HAQSingapore1990to2019.csv' and write the zipped data to the csv file. The csv module provides the functions 'writer()' and 'writerow()' in order to place each zipped pair in correct rows. We ensure that the data has column headers for future plotting purposes.

In [152…
```
with open('HAQSingapore1990to2019.csv', 'w', newline='') as csvfile:
    writer = csv.writer(csvfile)
    writer.writerow(['Year', 'HAQ Index 95% UI']) # Assign column headers as 'Year'
    for four_year_cells, sg_haq_cells in zipped_year_to_haq: # Iterate through the
        # Write each pair in a row with the first column being 'Year' and the secon
        writer.writerow([four_year_cells, sg_haq_cells])
```

The pandas module is used here for dataframe management, reading the newly created csv file 'HAQSingapore1990to2019.csv' and assigning it to a dataframe.

In [227…
```
haq_df = pd.read_csv('HAQSingapore1990to2019.csv',header=0)
haq_df # Print the dataframe to check it is correct
```

Out[227…

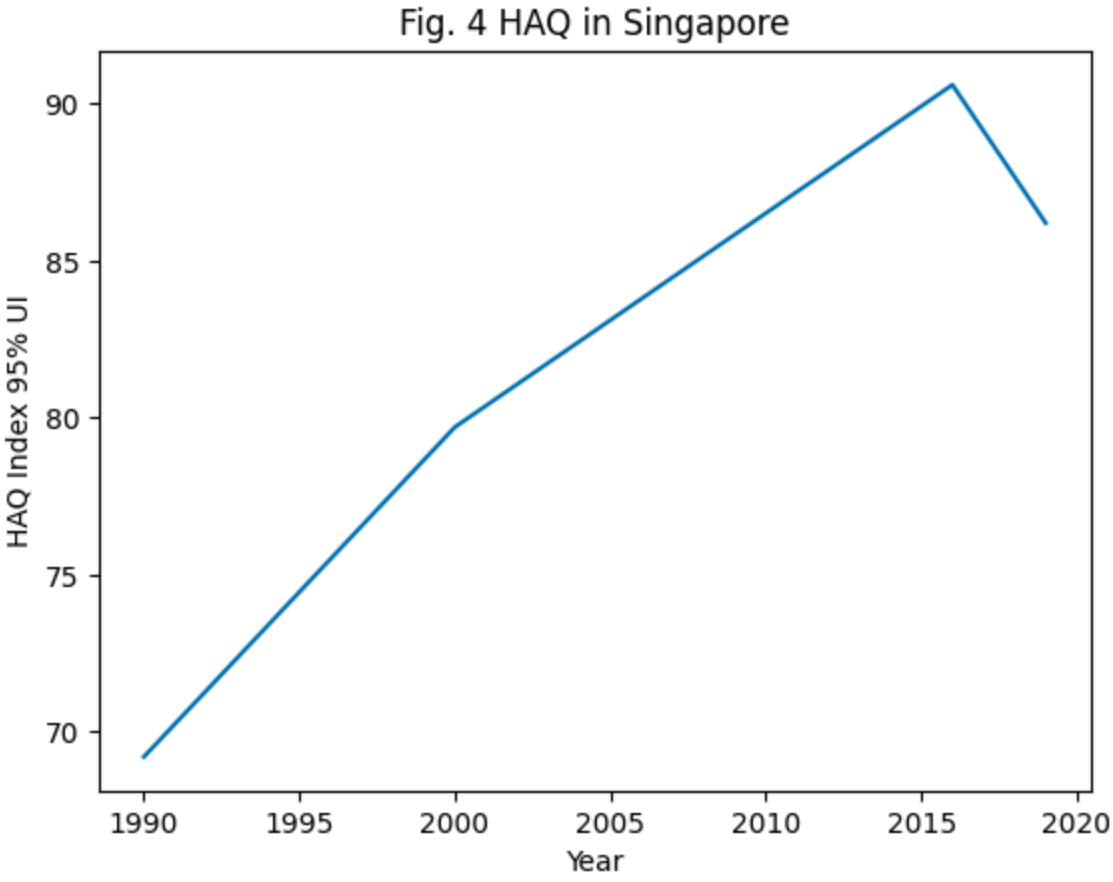
	Year	HAQ Index 95% UI
0	1990	69.2
1	2000	79.7
2	2016	90.6
3	2019	86.2

By using 'header=0', we assign the first row of 'HAQSingapore1990to2019.csv' as the headers which are 'Year' and 'HAQ Index 95% UI'.
We then use matplotlib and seaborn libraries again to plot a line plot for the HAQ Index values for Singapore.

In [156…
```
fig, haq_plot = plt.subplots()
haq_plot.plot(haq_df['Year'],haq_df['HAQ Index 95% UI'])

haq_plot.set_xlabel("Year")
haq_plot.set_ylabel("HAQ Index 95% UI")
haq_plot.set_title("Fig. 4 HAQ in Singapore")
```

Out[156…
```
Text(0.5, 1.0, 'Fig. 4 HAQ in Singapore')
```



4.3 Analysis of HAQ Index values for Singapore

Analysis of the plotted line plot reveals that the HAQ Index for Singapore experiences a steep drop after 2016. This suggests that the quality of accessible healthcare in Singapore experiences a rapid decline despite growing from 1990 to 2015.

We will study the correlation between the allocation of the healthcare workforce and the change in the quality of accessible healthcare quality in Singapore.

5 Linking Workforce Trends to Healthcare Quality

This section will combine the analysis of all data to determine the link between the healthcare workforce with the healthcare quality in Singapore. Link the findings in section 3 and 4 together and analyse how they are connected and correlated to one another. Following that, more research is done to find how the trends come about and how they may impact the future of the healthcare service and the quality in can provide.

5.1 Correlations Between Staffing and HAQ Scores

Observing on a broader level, the number of healthcare workers in Singapore experiences an overall increase between 2016 and 2019. However, from 2016 to 2019, the HAQ Index score

experiences a rapid decline. A deeper level of observation and analysis is required to understand the causes in terms of the healthcare workforce.

From studying the healthcare workforce for different sectors, between 2016 and 2019 we see that the number of Advanced Practice Nurses and, Nurses and Midwives experience a greater increase in the public sector compared to the private sector. This goes against the finding that the HAQ Index scored lower in 2019 than in 2016.

In section 3.2.3, it was found that there was a greater focus on providing educational opportunities to pursue a career in nursing as well as benefits to those in the nursing field. It could be argued that the increase in opportunities to pursue a career in nursing lead to the quality of nurses dropping. Besides that, providing more benefits to those in the nursing field could shift the focus of nurses away from providing quality healthcare towards superficial benefits. Both these factors could lead to the decline in the quality of nursing and eventually the quality of healthcare in Singapore.

As for therapists, between 2016 and 2019, the number of Speech Therapists, Occupational Therapists and Physiotherapists experience a greater increase in the private sector compared to the public sector. This aligns with the finding that the HAQ Index scored lower in 2019 than in 2016.

Therapists play an imperative role for Intermediate and Long Term Care (ILTC).[8] The Singapore government provides an avenue to access these healthcare services through agencies like the Agency for Integrated Care (AIC). However this avenue itself comes with challenges like high costs and long waiting times. It is estimated that the costs ranges from $52 to $102 per session without transport fees.[9] Along with that, waiting times for the application process may take up to four weeks.[10] Both the high costs and long waiting times can be attributed to the less than proportionate increase of the number of therapists in the public sector compared to the private sector. This limits the access of quality healthcare for patients and in turn affects the quality of healthcare in Singapore.

5.2 Singapore Population

Focus on the Singapore population count and distribution is important in understanding the effects of the healthcare workforce on the quality of healthcare in Singapore. From 2016 to 2019, Singapore experienced a population increase of almost 1 million.[11] Beyond the population count, the percentage of elderly in the population must be considered as well. The percentage of residents aged 65 and above in 2016 was 12.4%. This percentage would rise to 14.4% in 2014.[12] While a 2% increase may not seem like much, the percentage increase of elderly along with the absolute population count increase indicates that the percentage increase is far greater than it seems.

6 Implications and Future Research

6.1 Implications

The decline in Singapore's HAQ Index scores from 2016 to 2019, despite an increase in the healthcare workforce, reveals complex challenges in balancing workforce growth with quality outcomes.

6.1.1 Workforce Distribution

There exists an imbalanced focus on different healthcare professions across the public and private healthcare sectors. This suggests that, on a systemic level, policies tend to focus on the solution instead of the root cause. While nursing in the public expanded, the quality may have been impacted as the rapid increase in the avenue for education and superficial incentives could have diluted skill levels in nurses. On the other hand, limited focus on therapists led to the slow growth in the public sector and thus, affecting the ability of public healthcare agencies to operate efficiently and quickly.

6.1.2 Aging Population

There is inadequate scaling of the public healthcare sector, especially for the aging population and particularly for those in need of ILTC. The ever increasing percentage of the elderly in Singapore calls for the need to shift the focus towards this demographic and ensure accessible healthcare of quality.

6.2 Future Research

Further research topics that can be explored include:

1. Investigating the relationship between rapid workforce growth and the quality outcomes in nursing.
2. Analysing the demands of ILTC on the healthcare workforce

7 Conclusions and Reflections

The research highlights the importance of aligning workforce growth with healthcare quality. Increasing the healthcare workforce numbers alone is not sufficient to meet rising demands as this study highlights that absolute growth does not guarantee improved quality. Vulnerabilities in the healthcare system appear due to the disproportionate healthcare workforce distribution in the private and public sectors as well as potential skill dilution during the sudden growth of the healthcare workforce, particularly nursing.

Beyond that, the challenges associated with accessibility to healthcare services, like high costs and long waiting times, reinforce the importance of balancing resources for both the

healthcare workforce and quality. It is important that policies and healthcare providers remain focused on providing sufficient quality healthcare services.

These reflections emphasise the significance of addressing the changing population of Singapore with solutions that provide immediate effects and long-term focus.

References

[1] The Legatum Prosperity Index. Singapore. 2023. (Last accessed on 15 December 2024). Available from: https://www.prosperity.com/globe/singapore

[2] The Straits Times. S'pore's population ageing rapidly: Nearly 1 in 5 citizens is 65 years and older. 2022. (Last accessed on 15 December 2024). Available from: https://www.straitstimes.com/singapore/singapores-population-ageing-rapidly-184-of-citizens-are-65-years-and-older

[5] Ministry of Health Singapore. New 2-Year Bachelor of Science (Nursing) Programme to Attract More Mid-Career Singaporeans to Join Nursing. 2018. (Last accessed on 15 December 2024). Available from: https://www.moh.gov.sg/newsroom/new-2-year-bachelor-of-science-(nursing)-programme-to-attract-more-mid-career-singaporeans-to-join-nursing

[6] National University of Singapore. New graduate diploma for community health nursingc. 2018. (Last accessed on 15 December 2024). https://news.nus.edu.sg/new-graduate-diploma-for-community-health-nursing/

[7] Ministry of Health Singapore, Cheng Li Hui, MP of Tampines GRC. Retention and career opportunities of nurses. 2016. (Last accessed on 15 December 2024). Available from: https://www.moh.gov.sg/newsroom/retention-and-career-opportunities-of-nurses

[8] Ministry of Health Singapore. Intermediate and Long Term Care (ILTC) Services. (Last accessed on 15 Decemeber 2024). Available from: https://www.moh.gov.sg/seeking-healthcare/find-a-medical-facility/mental-health-services/intermediate-and-long-term-care-services

[9] Agency for Integrated Care. Community Rehabilitation Centre. (Last accessed on 15 December 2024). Available from: https://www.aic.sg/care-services/community-rehabilitation-centre/

[10] Agency for Integrated Care. Interim Disability Assistance Programme for the Elderly (IDAPE). (Last accessed on 15 December 2024). Available from: https://www.aic.sg/financial-assistance/interim-disability-assistance-programme-for-the-elderly-idape/

[11] Strategy Group, Prime Minister's Office, Singapore Department of Statistics, Ministry of Home Affairs, Immigration & Checkpoints Authority, Ministry of Manpower. Population in Brief 2019. 2019. (Last accessed on 15 December 2024). Available from: https://www.strategygroup.gov.sg/files/media-centre/publications/population-in-brief-2019.pdf

Data and codes

[3] data.gov.sg. Open data license. (Last accessed on 15 December 2024). Available from: https://data.gov.sg/open-data-licence

[4] Institute for Health Metrics and Evaluation. Data use agreement. (Last accessed on 15 December 2024). Available from: https://www.healthdata.org/Data-tools-practices/data-practices/ihme-free-charge-non-commercial-user-agreement

[12] statista. Residents aged 65 years and older as share of the resident population in Singapore from 1970 to 2023. 2024. (Last accessed from 15 December 2024). Available from: https://www.statista.com/statistics/1112943/singapore-elderly-share-of-resident-population/