In [1]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
footBall_data = pd.read_csv('HW1_data.csv')
```

1. The dataframe has 684 records.

In [2]:
```
footBall_data.shape[0]
```

Out[2]:
```
684
```

1. The numerical fields are: matches, wins, draws, loses, scored, conceded, year, position and pts. And the categorical fields are: league and team. We can infer from the code below that 'league' and 'team' are categorical values because they were typed as "object", and the others fields are 'numerical' because they were typed as "int\float".

In [3]:
```
print(footBall_data.dtypes)
```

```
league       object
year          int64
position      int64
team         object
matches       int64
wins          int64
draws         int64
loses         int64
scored        int64
conceded    float64
pts           int64
dtype: object
```

1. The Number of unique values in categorical fields are: in league: 6 and in team: 168.

In [6]:
```
print("Number of unique values in league:", footBall_data['league'].nunique())
print("Number of unique values in team:", footBall_data['team'].nunique())
```

```
Number of unique values in league: 6
Number of unique values in team: 168
```

1. footBall_data has one field with null value, which is 'conceded' that has 3. we can see from the code below that only 'conceded' has null value, and it has 3 of them.

In [ ]:
```
footBall_data.isna().sum()
```

5.footBall_data has 6 different leagues; 'Bundesliga', 'EPL', 'La_liga', 'Ligue_1', 'RFPL' and 'Serie_A'. In 'Bundesliga' played 24 teams - which makes her the league with the least played teams. In 'EPL' played 30 teams. In 'La_liga' played 30 teams. In 'Ligue_1' played 29 teams. In 'RFPL' played 25 teams. In 'Serie_A' played 30 teams.

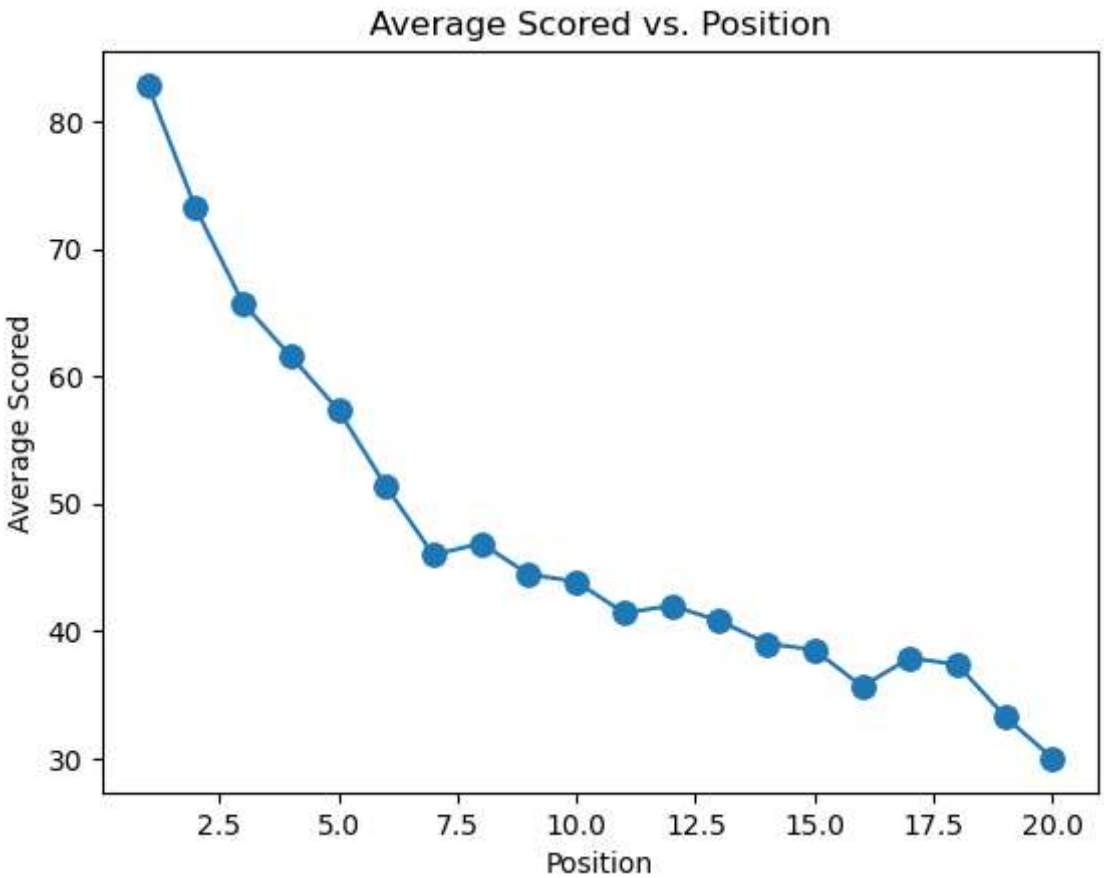
In [7]:
```
gb_league = footBall_data.groupby('league')
gb_league.team.nunique()
```

Out[7]:
```
league
Bundesliga    24
EPL           30
La_liga       30
Ligue_1       29
RFPL          25
Serie_A       30
Name: team, dtype: int64
```

6.The code below indicates that as the average score increases, the position of the group increases. The Correlation between Scored ans Position is: -0.7002088043380128

In [8]:
```
df_team_year_mean = footBall_data.groupby(['team', 'year']).mean()['scored'].reset_inc
df_team_year_mean['position'] = footBall_data.groupby(['team', 'year']).mean()['positi
print(df_team_year_mean)

df_team_year_mean = footBall_data.groupby(['team', 'year']).mean()[['scored', 'positic
df_mean_position = df_team_year_mean.groupby('position').mean()

sns.lineplot(x='position', y='scored', data=df_mean_position)
sns.scatterplot(x='position', y='scored', data=df_mean_position, s=100)

plt.xlabel('Position')
plt.ylabel('Average Scored')
plt.title('Average Scored vs. Position')
plt.show()

print("The Correlation between Scored ans Position is:",  df_team_year_mean['position'
```

```
                     team  year  scored  position
0                AC Milan  2014    56.0      10.0
1                AC Milan  2015    49.0       7.0
2                AC Milan  2016    57.0       6.0
3                AC Milan  2017    56.0       6.0
4                AC Milan  2018    55.0       5.0
..                    ...   ...     ...       ...
679  Zenit St. Petersburg  2015    61.0       3.0
680  Zenit St. Petersburg  2016    50.0       3.0
681  Zenit St. Petersburg  2017    46.0       5.0
682  Zenit St. Petersburg  2018    57.0       1.0
683  Zenit St. Petersburg  2019    65.0       1.0

[684 rows x 4 columns]
```




```
the Correlation between Scored ans Position is: -0.7002088043380128
```

1. The league with the largest difference between means is: Ligue_1. The year with the lowest scored goals mean is: 2019. The year with the highest scored mean goals is: 2015.

In [19]:
```
median_scored_df = footBall_data.groupby(['league', 'year'])['scored'].median()

median_diff = median_scored_df.groupby('league').apply(lambda x: x.max() - x.min())

largest_diff_league = median_diff.idxmax()

largest_diff_league_df = median_scored_df.loc[median_scored_df.index.get_level_values(

lowest_median_year = largest_diff_league_df.idxmin()[0]
highest_median_year = largest_diff_league_df.idxmax()[0]

print (median_scored_df)
print ("The year with the lowest scored goals median is:", lowest_median_year)
print ("The year with the highest scored goals median is:",highest_median_year)
```

```
league      year
Bundesliga  2014    44.0
            2015    44.0
            2016    44.5
            2017    43.5
            2018    53.0
            2019    48.0
EPL         2014    46.5
            2015    48.5
            2016    47.5
            2017    44.5
            2018    51.5
            2019    46.5
La_liga     2014    43.0
            2015    45.5
            2016    53.0
            2017    46.5
            2018    47.0
            2019    46.0
Ligue_1     2014    44.0
            2015    46.0
            2016    42.5
            2017    45.5
            2018    46.0
            2019    33.0
RFPL        2014    31.0
            2015    34.5
            2016    30.5
            2017    31.5
            2018    30.5
            2019    36.5
Serie_A     2014    49.0
            2015    46.5
            2016    56.0
            2017    51.0
            2018    51.5
            2019    51.5
Name: scored, dtype: float64
The year with the lowest scored goals median is: Ligue_1
The year with the highest scored goals median is: Ligue_1
```

1. We have found 1 team that her points value is incorrect. the team is Crotone from Seria_A league in 2017. We assume that the source of the mistake is in the process of entering the data to the point column.

In [ ]:
```
footBall_data['my_cal_points'] = (footBall_data['wins'] * 3) + footBall_data['draws']
points_col = footBall_data['my_cal_points']

points_df = footBall_data[['league', 'team', 'my_cal_points']]

print(points_df.sort_values('my_cal_points', ascending=False))

not_equal_rows = footBall_data[footBall_data['my_cal_points'] != footBall_data['pts']]

print(not_equal_rows)
```

1. The code below indicates that the The league with the most failed team over the years is 'La_liga'.

In [ ]:
```
scoredVsConceded = footBall_data.loc[:, ['league', 'team', 'scored', 'conceded']].copy
scoredVsConceded = scoredVsConceded.dropna()
scoredVsConceded['failed'] = scoredVsConceded['conceded'] > scoredVsConceded['scored']
scoredVsConceded = scoredVsConceded.groupby('league')['failed'].sum().idxmax()
print("The league with the most failed teams:", scoredVsConceded)
```

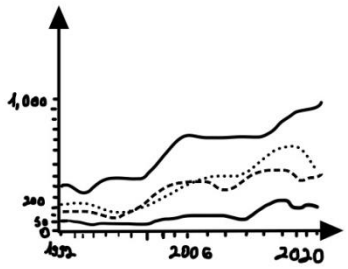
In [ ]:

נעשה שימוש ב GPT CHAT

חלק שני -

1. A. ניתן להסיק כי לכל המדינות המוצגות בגרף ישנה עלייה בהוצאות הצבאיות לאורך השנים 1992-2021.
לארצות הברית יש פיק בהוצאות בשנת 2010. לסין יש עליה מגמטית לאורך השנים.
לשאר המדינות המוצגות בגרף ישנה עליה מתונה יחסית לארצות הברית ולסין. (לערב הסעודית נראה שיש גם כן פיק בסביבות שנת 2015.)

B. הבעיה שקיימת בגרף הינה שקימיים שני צירי Y עם ערכים שונים, דבר אשר עלול לגרום לצופה להסיק מסקנות שגויות משום שזה יקשה עליו להעריך את ההוצאות ביחס לאיזה ציר.
C. ניתן להציג את הנתונים בצורה האפקטיבית ביותר באמצעות גרף עם שני צירים בלבד – X וY.
ציר Y שייצג דולר אמריקאי בטווח של 0 – 1000 (במיליארדים).
וציר X שייצג את השנים בסדר עולה משנת 1992 עד לשנת 2021.
את המדינות נציג בגרף זה ע"י ציור גרף קווי.



2. A. ניתן להסיק מהגרף שככל שיש עליה באחוזי הגידולי תירס וסויה המהונדסים גנטית ועליה בשימוש בחומר קוטל עשבים Glyphosate כך גם מקרי המוות עולים. נראה שיש קשר ישיר בין אותם גורמים לבין העלייה במקרי התמותה.

B. בעיצוב הגרף, יוצריו קיבלו מספר החלטות לא סטנדרטיות על מנת להעביר מסר מסויים.

החלטה אחת שעשו היא הצגת הנתון של אחוזי הגידול של תירס וסויה מהונדסים גנטית וגם כמות חומר קוטל העשבים שהוסיפו לגידולים יחד באותו ציר. פעולה זו יוצרת עומס של נתונים וסיווגם כנתון אחד הפועל באותה הדרך ויוצר את אותה המגמה ללא הפרדה ביניהם.
בנוסף, מאותו המסר, בחרו לסרטט באותו הגרף את המגמות של שניהם ולא להפרידם לשני גרפים נפרדים.
החלטה נוספת שעשו היא להוסיף את קו המגמה של אחוזי המוות כפי שהייתה עד 1993, כדי להראות שאחוזי המוות עלו באופן משמעותי יחסית לצפי ולמגמה שממשיכה מ1993.

אנחנו לא מסכימים עם מסר זה, מכיוון שלדעתנו הגרף מציג שני גורמים יחד, בלי יכולת להבדיל בין השפעת כל אחד מהם על שיעור התמותה. זאת אומרת, יוצר הגרף יכל להציגם זה לצד זה כדי להראות ששניהם מסוכנים לאדם באותה המידה, אך בפועל איננו יודעים מי מהם הוא הגורם העיקרי לעליה זו.

C.בעיה אחת בולטת בגרף היא העומס של נתונים המוצגים בה.
הגרף מכיל יותר מ2 משתנים והרבה מאוד מידע. לקורא הגרף קשה להבין מה כל ייצוג ויזואלי מייצג ומכך גם

קשה להבין את המסר המוצג בגרף. בנוסף, בגרף שני צירים וי שונים שגם הם מוסיפים לעומס ולבלבול.
בעיה שנייה היא אלמנטים ויזואלים זרים שמסיחים את הדעת מהמסר- הפסים האנכיים המוצגים מאחורי הגרף.

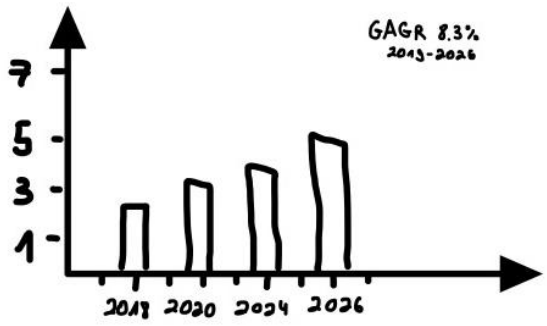
3. A. המידע שאמור להיות מועבר לצופה של הגרף באמצעות וויזואליזציה זו הוא שלאורך השנים 2019 – 2026 קיימת עליה בשוק הגבינות הטבעוניות.

B. בעיה אחת בויזואלזציה של הגרף היא השימוש בתרשים פאי שיוצר יחס לא פרופרציונלי בין הנתונים. בשנת 2026 שוק הגבינות הטבעוניות אמור להיות גדול כימעט פי 2 מזה של 2018, אך בתרשים נראה שגודלה היחסי של החתיכה המייצגת את שנת 2026 הוא גדול יותר מפי 2. זאת אומרת מבחינה ויזואלית קשה לצופה לעמוד את הפרשי הגדלים בין חתיכה לחתיכה.

בעיה נוספת הינה חוסר הצגת כל המידע הרלוונטי בתרשים ,חסרה שקיפות לגבי שאר השנים שנמדדו (חתיכות פאי ריקות).

בנוסף, בגרף אלמנטים ויזואלים רבים שמסיחים את הדעת מהמסר (רקע של גבינות, צבעים רבים בתרשים הפאי שאינם מייצגים לצופה דבר).

C. ויזואלזציה חלופית אחרת שניתן להשתמש בה כדי להציג את נתוני הגרף היא בגרף עמודות.
ציר X ייצג את השנים אותם רוצים להציג (כל שנה מ2018 עד 2026).
ציר Y ייצג את שווי השוק של כל שנה בשוק הגבינות הטבעונית.
לכל שנה תהיה עמודה אחת בגובה שווי השוק של שנה זו זאת.
בצד ירשם קצב צמיחה שנתי ממוצע (2019-2026).



4. הבעיה בכותרת הינה שכאשר רשום שכר ממוצע, הממוצע מושפע מערכי קיצון ולכן עלול שלא לייצג באופן אמין את משכורות המורים בשוק העבודה במדינה. זאת אומרת, המידע אכן נכון, אך אינו מייצג בצורה אמיתית את השכר שרוב המורים במדינה ירוויחו. לכן ישנה סבירות שאותם מורים צודקים בטענתם. כדי להציג את השכר שרוב המורים מקבלים במשק בצורה אמינה ניתן לחשב שכר חציוני, אשר נותן משקל רב יותר לערכים שמייצגים את הרוב.