harden

February 12, 2024

1 James Harden

1.1 Jaimez Hurden

[ ]:
```
import warnings
warnings.filterwarnings("ignore", category=DeprecationWarning)

import matplotlib.pyplot as plt
from gql import gql, Client
from gql.transport.requests import RequestsHTTPTransport
import pandas as pd
import seaborn as sns
```

[ ]:
```
# Setup the transport and client
_transport = RequestsHTTPTransport(
    url='http://127.0.0.1:8000/graphql/',
    use_json=True,
)

client = Client(
    transport = _transport,
    fetch_schema_from_transport=True,
)

# Define Query
query = gql('''
query playerAdvancedTotals {
  playerAdvancedAll {
    playerName
    season
    team
    position
    age
    winShares
    offensiveWs
    defensiveWs
    usagePercent
    minutesPlayed
```

```
    tsPercent
    assistPercent
    blockPercent
    offensiveRbPercent
    defensiveRbPercent
    ftr
    games
    per
    winSharesPer
    vorp
    turnoverPercent
    totalRbPercent
    threePAr
    stealPercent
    playerId
    id
  }
  playerTotalsAll {
      playerName
      season
      team
      position
      age
      effectFgPercent
      minutesPg
      twoPercent
      threePercent
      points
      assists
      blocks
      defensiveRb
      fieldAttempts
      fieldGoals
      ft
      fieldPercent
      ftAttempts
      ftPercent
      games
      gamesStarted
      offensiveRb
      personalFouls
      playerId
      steals
      threeAttempts
      threeFg
      totalRb
      turnovers
```

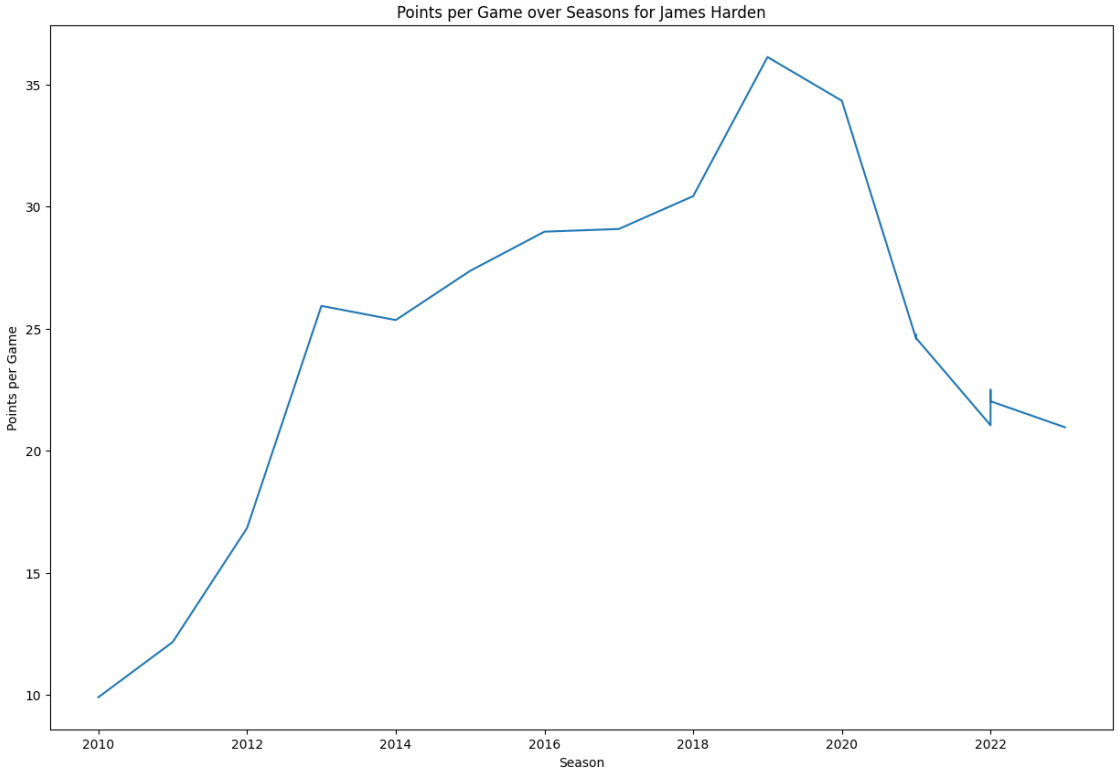
```
      twoFg
      playerId
      id
  }
}
''')

# Execute and store response

response = client.execute(query)
```

[ ]:
```
df_advanced = pd.DataFrame(response['playerAdvancedAll'])

df_total = pd.DataFrame(response['playerTotalsAll'])

final_df = pd.merge(df_advanced, df_total, on=['playerName', 'season', 'team',␣
 ↪'position', 'age', 'id'])

final_df['winShares']= final_df['winShares'].astype(float)
final_df['usagePercent']= final_df['usagePercent'].astype(float)

final_df.columns
```

[ ]:
```
Index(['playerName', 'season', 'team', 'position', 'age', 'winShares',
       'offensiveWs', 'defensiveWs', 'usagePercent', 'minutesPlayed',
       'tsPercent', 'assistPercent', 'blockPercent', 'offensiveRbPercent',
       'defensiveRbPercent', 'ftr', 'games_x', 'per', 'winSharesPer', 'vorp',
       'turnoverPercent', 'totalRbPercent', 'threePAr', 'stealPercent',
       'playerId_x', 'id', 'effectFgPercent', 'minutesPg', 'twoPercent',
       'threePercent', 'points', 'assists', 'blocks', 'defensiveRb',
       'fieldAttempts', 'fieldGoals', 'ft', 'fieldPercent', 'ftAttempts',
       'ftPercent', 'games_y', 'gamesStarted', 'offensiveRb', 'personalFouls',
       'playerId_y', 'steals', 'threeAttempts', 'threeFg', 'totalRb',
       'turnovers', 'twoFg'],
      dtype='object')
```

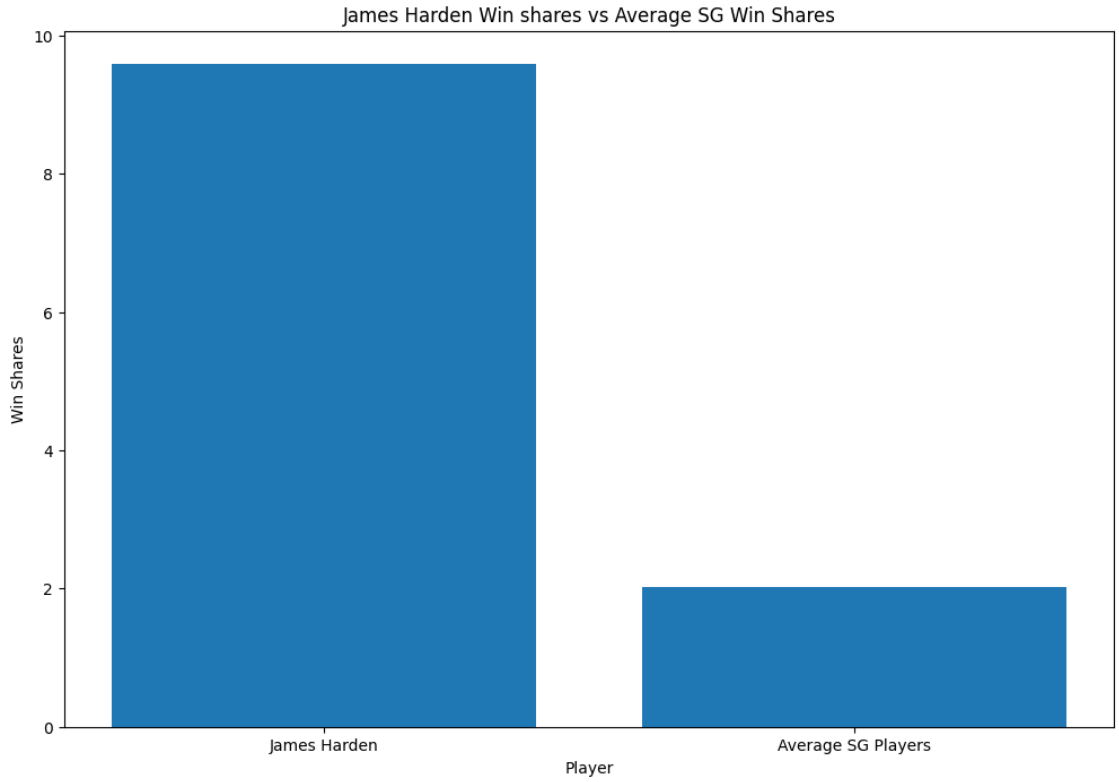
[ ]:
```
# Assuming pandas dataframe 'df'

# Time-Series Analysis for Points per game over seasons
plt.figure(figsize=(15,10))
plt.plot(final_df[final_df['playerName']=='James Harden']['season'],␣
 ↪final_df[final_df['playerName']=='James Harden']['points']/
 ↪final_df[final_df['playerName']=='James Harden']['games_x'])
plt.title('Points per Game over Seasons for James Harden')
plt.xlabel('Season')
plt.ylabel('Points per Game')
```

[ ]: Text(0, 0.5, 'Points per Game')



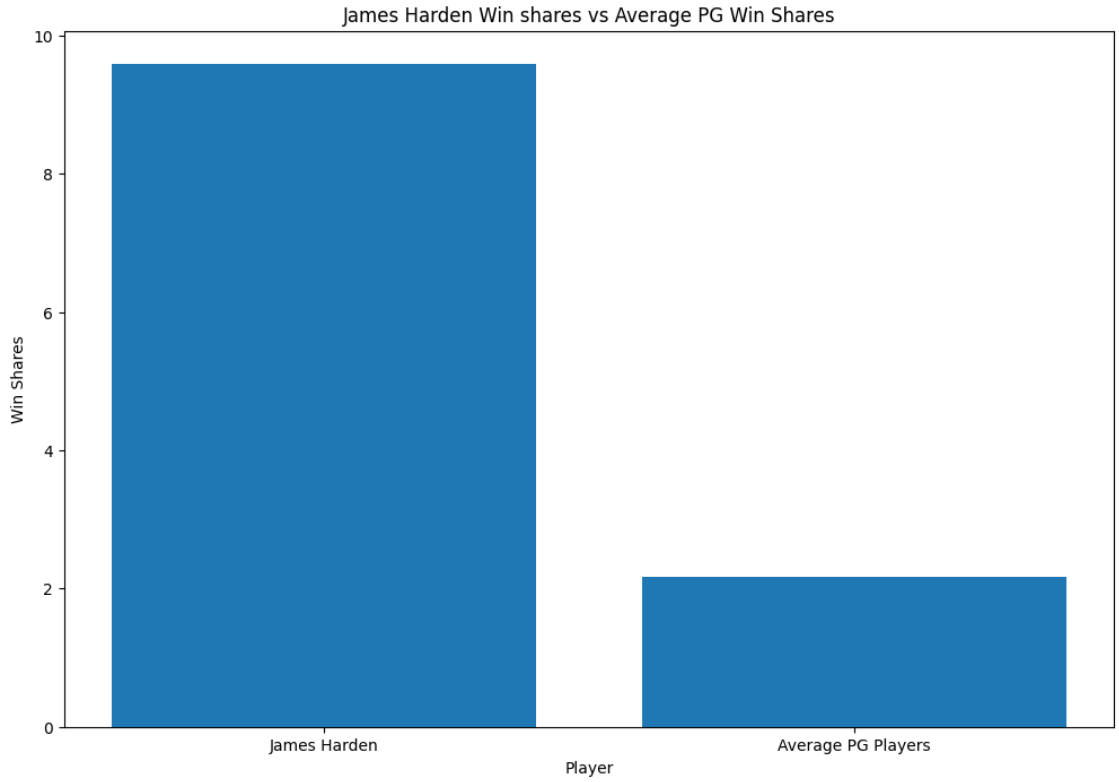
[ ]:
```
# Comparative Analysis for Win Shares
plt.figure(figsize=(12,8))
plt.bar(['James Harden', 'Average SG Players'],␣
  ↪[final_df[final_df['playerName']=='James Harden']['winShares'].mean(),␣
  ↪final_df[final_df['position']=='SG']['winShares'].mean()])
plt.title('James Harden Win shares vs Average SG Win Shares')
plt.xlabel('Player')
plt.ylabel('Win Shares')
```

[ ]: Text(0, 0.5, 'Win Shares')



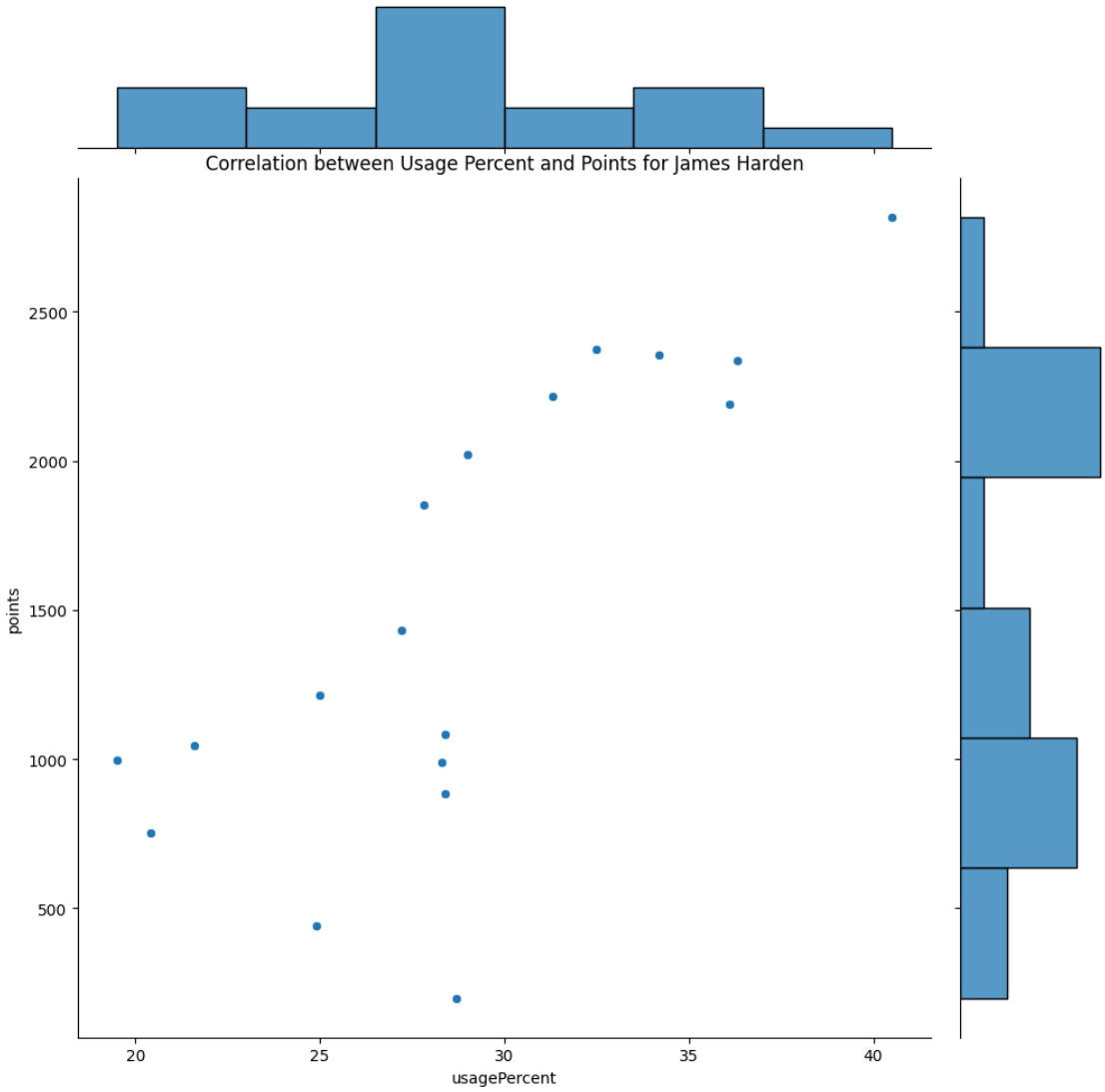
[ ]:
```
# Comparative Analysis for Win Shares
plt.figure(figsize=(12,8))
plt.bar(['James Harden', 'Average PG Players'],␣
  ↪[final_df[final_df['playerName']=='James Harden']['winShares'].mean(),␣
  ↪final_df[final_df['position']=='PG']['winShares'].mean()])
plt.title('James Harden Win shares vs Average PG Win Shares')
plt.xlabel('Player')
plt.ylabel('Win Shares')
```

[ ]:
```
Text(0, 0.5, 'Win Shares')
```



[ ]:
```
# Correlation Analysis for Points and UsagePercent
sns.jointplot(x='usagePercent', y='points',␣
↪data=final_df[final_df['playerName']=='James Harden'], height=10)
plt.title('Correlation between Usage Percent and Points for James Harden')
```

[ ]:
```
Text(0.5, 1.0, 'Correlation between Usage Percent and Points for James Harden')
```


Correlation between Usage Percent and Points for James Harden
points
usagePercent
2500
2000
1500
1000
500
20
25
30
35
40