NFL_Salary_Analysis

August 9, 2024

[1]:
```
# Import necessary libraries
import pandas as pd
from NFL_Functions import *
```

[2]:
```
# Create DataFrame with team cap data
df = pd.read_excel('nfl_team_cap_data.xlsx')

df.head()
```

[2]:

	Player	Pos	Age	Cap Hit	Cap Hit Pct	Dead Cap	Base	Signing Bonus	Per Game	Roster	Option	Workout	Restructure	Incentives	Team	Year
0	Larry Fitzgerald	WR	28.0	$16,250,000	13.50%	-	$2,000,000	$5,000,000	-	$8,000,000	$1,250,000	-	-	-	arizona-cardinals	2011
1	Levi Brown	LT	27.0	$9,420,000	7.83%	-	$6,777,500	-	-	-	$2,392,500	$250,000	-	-	arizona-cardinals	2011
2	Adrian Wilson	S	32.0	$7,550,000	6.27%	-	$3,500,000	$200,000	-	$3,000,000	-	$250,000	$600,000	-	arizona-cardinals	2011
3	Darnell Dockett	DE	30.0	$5,600,000	4.65%	($15,000,000)	$2,350,000	-	-	-	$3,000,000	$250,000	-	-	arizona-cardinals	2011
4	Derek Anderson	QB	28.0	$5,387,500	4.48%	-	$4,087,500	$1,250,000	-	-	-	$50,000	-	-	arizona-cardinals	2011

[3]:
```
# Begin programmatically inspecting the data
print(df.dtypes)
```

```
Player            object
Pos               object
Age              float64
Cap Hit           object
Cap Hit Pct       object
```

```
Dead Cap          object
Base              object
Signing Bonus     object
Per Game          object
Roster            object
Option            object
Workout           object
Restructure       object
Incentives        object
Team              object
Year               int64
dtype: object
```

[4]:
```
# Inspect for NaN values
df[df.isnull().any(axis=1)]
```

[4]:

	Player	Pos	Age	Cap Hit	Cap Hit Pct	Dead Cap	Base	Signing Bonus	Per Game	Roster	Option	Workout	Restructure	Incentives	Team	Year
915	Billy Volek	QB	NaN	$2,500,000	2.08%	-	$2,500,000	-	-	-	-	-	-	-	los-angeles-chargers	2011
1404	Matt Kroul	C	NaN	$450,000	0.37%	-	$450,000	-	-	-	-	-	-	-	new-york-jets	2011
1556	Ty Boyle	DE	NaN	$375,333	0.31%	-	$375,000	$333	-	-	-	-	-	-	pittsburgh-steelers	2011
3170	Nick Hixon	CB	NaN	$390,000	0.32%	-	$390,000	-	-	-	-	-	-	-	new-orleans-saints	2012
3399	Grant Ressel	K	NaN	$390,833	0.32%	-	$390,000	$833	-	-	-	-	-	-	pittsburgh-steelers	2012
6943	Cameron Fuller	CB	NaN	$24,705	0.02%	-	$420,000	-	-	-	-	-	-	-	san-francisco-49ers	2014

[5]:
```
# List of row indexes to update in 'Age' column
age_rows = [915, 1404, 1556, 3170, 3399, 6943]

# List of ages to update
ages = [35, 25, 23, 23, 24, 23]

# Update the 'Age' column for the specified row indexes
for row, age in zip(age_rows, ages):
```

```
    df.loc[row, 'Age'] = age

# Display updated rows to verify changes
df.loc[age_rows]
```

[5]:

	Player	Pos	Age	Cap Hit	Cap Hit Pct	Dead Cap	Base	Signing Bonus	Per Game	Roster	Option	Workout	Restructure	Incentives	Team	Year
915	Billy Volek	QB	35.0	$2,500,000	2.08%	-	$2,500,000	-	-	-	-	-	-	-	los-angeles-chargers	2011
1404	Matt Kroul	C	25.0	$450,000	0.37%	-	$450,000	-	-	-	-	-	-	-	new-york-jets	2011
1556	Ty Boyle	DE	23.0	$375,333	0.31%	-	$375,000	$333	-	-	-	-	-	-	pittsburgh-steelers	2011
3170	Nick Hixon	CB	23.0	$390,000	0.32%	-	$390,000	-	-	-	-	-	-	-	new-orleans-saints	2012
3399	Grant Ressel	K	24.0	$390,833	0.32%	-	$390,000	$833	-	-	-	-	-	-	pittsburgh-steelers	2012
6943	Cameron Fuller	CB	23.0	$24,705	0.02%	-	$420,000	-	-	-	-	-	-	-	san-francisco-49ers	2014

[6]:
```
# List of columns to exclude from data type conversion
exclude_cols = ['Player', 'Pos', 'Age', 'Team', 'Year']

# Loop through columns and remove characters to leave only numeric values
for column in df.columns:
    if column not in exclude_cols:
        if column == 'Cap Hit Pct':
            df[column] = df[column].str.replace('%', '').replace('-', '0')
        df[column] = df[column].str.replace('[\$,]', '', regex=True).
 ↪replace('-', '0')
        if column == 'Dead Cap':
            df[column] = df[column].apply(lambda x: -int(x.replace('(', '').
 ↪replace(')', '')) if '(' in x else (int(x) if x != '0' else 0))
        else:
            df[column] = pd.to_numeric(df[column], errors='coerce',␣
 ↪downcast='integer')

df.head()
```

[6]:

	Player	Pos	Age	Cap Hit	Cap Hit Pct	Dead Cap	Base	Signing Bonus	Per Game	Roster	Option	Workout	Restructure	Incentives	Team	Year
0	Larry Fitzgerald	WR	28.0	16250000	13.50	0	2000000	5000000	0	8000000	1250000	0	0	0	arizona-cardinals	2011
1	Levi Brown	LT	27.0	9420000	7.83	0	6777500	0	0	0	2392500	250000	0	0	arizona-cardinals	2011
2	Adrian Wilson	S	32.0	7550000	6.27	0	3500000	200000	0	3000000	0	250000	600000	0	arizona-cardinals	2011
3	Darnell Dockett	DE	30.0	5600000	4.65	-15000000	2350000	0	0	0	3000000	250000	0	0	arizona-cardinals	2011
4	Derek Anderson	QB	28.0	5387500	4.48	0	4087500	1250000	0	0	0	50000	0	0	arizona-cardinals	2011

[7]:
```
# Convert the 'Age' column to int64 data type
df['Age'] = df['Age'].astype('int64')

df.head()
```

[7]:

	Player	Pos	Age	Cap Hit	Cap Hit Pct	Dead Cap	Base	Signing Bonus	Per Game	Roster	Option	Workout	Restructure	Incentives	Team	Year
0	Larry Fitzgerald	WR	28	16250000	13.50	0	2000000	5000000	0	8000000	1250000	0	0	0	arizona-cardinals	2011
1	Levi Brown	LT	27	9420000	7.83	0	6777500	0	0	0	2392500	250000	0	0	arizona-cardinals	2011
2	Adrian Wilson	S	32	7550000	6.27	0	3500000	200000	0	3000000	0	250000	600000	0	arizona-cardinals	2011
3	Darnell Dockett	DE	30	5600000	4.65	-15000000	2350000	0	0	0	3000000	250000	0	0	arizona-cardinals	2011
4	Derek Anderson	QB	28	5387500	4.48	0	4087500	1250000	0	0	0	50000	0	0	arizona-cardinals	2011

[8]:
```
# Applys convert_team_name function to 'Team' column
df['Team'] = df['Team'].apply(convert_team_name)

df.head()
```

[8]:

	Player	Pos	Age	Cap Hit	Cap Hit Pct	Dead Cap	Base	Signing Bonus	Per Game	Roster	Option	Workout	Restructure	Incentives	Team	Year
0	Larry Fitzgerald	WR	28	16250000	13.50	0	2000000	5000000	0	8000000	1250000	0	0	0	Cardinals	2011
1	Levi Brown	LT	27	9420000	7.83	0	6777500	0	0	0	2392500	250000	0	0	Cardinals	2011
2	Adrian Wilson	S	32	7550000	6.27	0	3500000	200000	0	3000000	0	250000	600000	0	Cardinals	2011
3	Darnell Dockett	DE	30	5600000	4.65	-15000000	2350000	0	0	0	3000000	250000	0	0	Cardinals	2011
4	Derek Anderson	QB	28	5387500	4.48	0	4087500	1250000	0	0	0	50000	0	0	Cardinals	2011

[9]:
```
# Check to make sure team names are correct
df['Team'].unique()
```

[9]:
```
array(['Cardinals', 'Falcons', 'Ravens', 'Bills', 'Panthers', 'Bears',
       'Bengals', 'Browns', 'Cowboys', 'Broncos', 'Lions', 'Packers',
       'Texans', 'Colts', 'Jaguars', 'Chiefs', 'Chargers', 'Rams',
       'Raiders', 'Dolphins', 'Vikings', 'Patriots', 'Saints', 'Giants',
       'Jets', 'Eagles', 'Steelers', 'Seahawks', '49Ers', 'Buccaneers',
       'Titans', 'Commanders'], dtype=object)
```

[10]:
```
df.describe()
```

[10]:

	Age	Cap Hit	Cap Hit Pct	Dead Cap	Base
count	22412.000000	2.241200e+04	22412.000000	2.241200e+04	2.241200e+04
mean	26.246297	2.222560e+06	1.363261	-3.145746e+06	1.377600e+06
std	3.260187	3.311812e+06	1.979749	8.062680e+06	2.072396e+06
min	20.000000	0.000000e+00	0.000000	-2.293675e+08	0.000000e+00
25%	24.000000	5.256510e+05	0.340000	-2.200000e+06	4.950000e+05
50%	26.000000	8.716155e+05	0.500000	-1.350000e+05	7.500000e+05
75%	28.000000	2.500000e+06	1.520000	0.000000e+00	1.165000e+06
max	46.000000	3.713382e+07	17.530000	0.000000e+00	2.750000e+07

```
       Signing Bonus      Per Game        Roster        Option       Workout  \
count   2.241200e+04  2.241200e+04  2.241200e+04  2.241200e+04  2.241200e+04
mean    5.501230e+05  4.931955e+03  1.612745e+05  3.914382e+04  1.820571e+04
std     1.165480e+06  5.404391e+04  8.015357e+05  3.090350e+05  7.146214e+04
min    -1.080880e+05  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00
25%     0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00
50%     6.459100e+04  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00
75%     5.459368e+05  0.000000e+00  0.000000e+00  0.000000e+00  0.000000e+00
max     2.448157e+07  1.750000e+06  2.740000e+07  1.096000e+07  1.745000e+06

        Restructure    Incentives          Year
count  2.241200e+04  2.241200e+04  22412.000000
mean   8.622632e+04  1.622952e+04   2016.937043
std    5.850063e+05  1.658205e+05      3.776072
min    0.000000e+00 -1.852500e+06   2011.000000
25%    0.000000e+00  0.000000e+00   2014.000000
50%    0.000000e+00  0.000000e+00   2017.000000
75%    0.000000e+00  0.000000e+00   2020.000000
max    1.526750e+07  7.535000e+06   2023.000000
```

[11]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 22412 entries, 0 to 22411
Data columns (total 16 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Player         22412 non-null  object
 1   Pos            22412 non-null  object
 2   Age            22412 non-null  int64
 3   Cap Hit        22412 non-null  int32
 4   Cap Hit Pct    22412 non-null  float64
 5   Dead Cap       22412 non-null  int64
 6   Base           22412 non-null  int32
 7   Signing Bonus  22412 non-null  int32
 8   Per Game       22412 non-null  int32
 9   Roster         22412 non-null  int32
 10  Option         22412 non-null  int32
 11  Workout        22412 non-null  int32
 12  Restructure    22412 non-null  int32
 13  Incentives     22412 non-null  int32
 14  Team           22412 non-null  object
 15  Year           22412 non-null  int64
dtypes: float64(1), int32(9), int64(3), object(3)
memory usage: 2.0+ MB
```

[12]:
```
print(df.isnull().sum())
```

```
Player           0
Pos              0
Age              0
Cap Hit          0
Cap Hit Pct      0
Dead Cap         0
Base             0
Signing Bonus    0
Per Game         0
Roster           0
Option           0
Workout          0
Restructure      0
Incentives       0
Team             0
Year             0
dtype: int64
```

[13]:
```
# Find the total cap hit by position, grouped by team and year
sal_by_pos = df.groupby(['Year', 'Team', 'Pos']).agg({'Cap Hit': 'sum', 'Cap␣
 ↪Hit Pct': 'sum'}).reset_index()

print(sal_by_pos)
```

```
      Year     Team Pos   Cap Hit  Cap Hit Pct
0     2011    49Ers   C   2678642         2.22
1     2011    49Ers  CB  10518506         8.74
2     2011    49Ers  DE  11179166         9.29
3     2011    49Ers  DT   7811375         6.49
4     2011    49Ers  FB    795613         0.66
…     …       …   ..       …           …
8055  2023  Vikings  RB   4472910         1.99
8056  2023  Vikings  RT  10749690         4.78
8057  2023  Vikings   S  16449943         7.31
8058  2023  Vikings  TE   4654000         2.08
8059  2023  Vikings  WR  10616227         4.73

[8060 rows x 5 columns]
```

[14]:
```
sal_by_pos.head()
```

[14]:

	Year	Team	Pos	Cap Hit	Cap Hit Pct
0	2011	49Ers	C	2678642	2.22
1	2011	49Ers	CB	10518506	8.74
2	2011	49Ers	DE	11179166	9.29
3	2011	49Ers	DT	7811375	6.49
4	2011	49Ers	FB	795613	0.66

[15]:
```
sal_by_pos.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8060 entries, 0 to 8059
Data columns (total 5 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   Year         8060 non-null   int64
 1   Team         8060 non-null   object
 2   Pos          8060 non-null   object
 3   Cap Hit      8060 non-null   int32
 4   Cap Hit Pct  8060 non-null   float64
dtypes: float64(1), int32(1), int64(1), object(2)
memory usage: 283.5+ KB
```

[16]:
```
sal_by_pos.describe()
```

[16]:

	Year	Cap Hit	Cap Hit Pct
count	8060.000000	8.060000e+03	8060.000000
mean	2016.919603	6.180151e+06	3.790744
std	3.723605	6.365228e+06	3.776843
min	2011.000000	0.000000e+00	0.000000
25%	2014.000000	1.342709e+06	0.850000
50%	2017.000000	3.980798e+06	2.515000
75%	2020.000000	8.928885e+06	5.570000
max	2023.000000	4.716807e+07	23.790000

[17]:
```
# Export dataset to excel
sal_by_pos.to_excel('salary_by_position.xlsx', index=False)
```