PUBG Data Description

In a PUBG game, up to 100 players start in each match (matchId). Players can be on teams (groupId) which get ranked at the end of the game (winPlacePerc) based on how many other teams are still alive when they are eliminated. In game, players can pick up different munitions, revive downed-but-not-out (knocked) teammates, drive vehicles, swim, run, shoot, and experience all of the consequences -- such as falling too far or running themselves over and eliminating themselves. You are provided with a large number of anonymized PUBG game stats, formatted so that each row contains one player's post-game stats. The data comes from matches of all types: solos, duos, squads, and custom; there is no guarantee of there being 100 players per match, nor at most 4 players per group.

In [17]:

```
Data fields
DBNOs - Number of enemy players knocked.
assists - Number of enemy players this player damaged that were killed by teammates.
boosts - Number of boost items used.
damageDealt - Total damage dealt. Note: Self inflicted damage is subtracted.
headshotKills - Number of enemy players killed with headshots.
heals - Number of healing items used.
Id - Player’s Id
killPlace - Ranking in match of number of enemy players killed.
killPoints - Kills-based external ranking of players. (Think of this as an Elo ranking where only kills matter.) If there is a value other than -1 in rankPoints, then any 0 in killPoints should be treated as a “None”.
killStreaks - Max number of enemy players killed in a short amount of time.
kills - Number of enemy players killed.
longestKill - Longest distance between player and player killed at time of death. This may be misleading, as downing a player and driving away may lead to a large longestKill stat.
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

```
matchDuration - Duration of match in seconds.
matchId - ID to identify matches. There are no matches that are in both
the training and testing set.
matchType - String identifying the game mode that the data comes from.
The standard modes are “solo”, “duo”, “squad”, “solo-fpp”, “duo-fpp”
, and “squad-fpp”; other modes are from events or custom matches.
```

```
  File "<ipython-input-17-6cdcf5ece6ee>", line 1
    Data fields
              ^
SyntaxError: invalid syntax
```

In [ ]:

```
# Import all required library
import numpy as np
import pandas as pd
import scipy.stats as stats
import seaborn as sns
import matplotlib.pyplot as plt
from scipy import stats
from sklearn.feature_selection import chi2
```

In [2]:

```
data = pd.read_csv("D:\\data dipali\\Python\\Analaytic\\data.csv",sep=
',',engine='python')
```

In [3]:

```
data
```

Out[3]:

	Unnamed: 0	Id	groupId	matchId	assists	boosts
0	2093518	a3e3cea9f2e314	c9f6eaa81245b5	a3d8693390584c	0	0
1	3701672	43e4be7ad72cc7	521836de101ee8	b65de9055327e0	0	0
2	2059784	9a68690e31fdff	4a72ffa2cebd90	d6aad3f9830e60	0	1
3	1302036	b147e1bd448fc4	9a8991656b3fea	d931c0932d8aca	0	0
4	297180	d818b4edd59612	eece87c8b846b3	ec2b5ed94baae3	0	2

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

	Unnamed: 0	Id	groupId	matchId	assists	boosts
5	2862125	e705cb7a4446ea	9d7180a567d020	b146588581df18	0	0
6	365937	e253262031e08f	e5df2dcdaf3f4b	c1385ae84655e6	0	0
7	3651102	c264a5ce0d4ca1	0efbbf391d7f62	b280791eb7a3fd	0	8
8	2363197	342437cd2b5680	7acbacbed1b711	99cbf12f555f61	0	2
9	271498	3cf3cc81325a37	537c4bd393dcff	38a8b94623d77d	2	1
10	2329569	cf3fc190ef0636	9ab58287ac919e	2b13e071d8c637	0	0
11	2748178	e623835ca5d21d	6316b093b9ecbe	895de0a11d38c9	0	0
12	2282301	d7c5c1a1bd918a	e2029475901d30	1a2af48f98525c	0	0
13	950009	0bcc4efeade33f	ae841837835a89	4ae66868f61da3	0	0
14	414480	f842ff43c8f9ad	864a3a2466261f	4d7116c9b9a2b2	0	0
15	2620592	4b5dcd20c7810c	7c9f91decbf708	f8e9b98338c0da	0	0
16	1020408	681756a82a8567	60bcf075484413	cfe91f6f964f80	0	0
17	421860	864943a1ad0e67	7dbd019271daf8	39f290e85e9cd1	0	0
18	893884	cd66f4def42850	b4ec6ac08989f8	daa7b148b9e29d	0	0
19	3326034	129a839a443099	0d5f1c9f9a7cce	197cb0812e83b8	0	0
20	3767124	56d6949b3d643d	27f4292af663d1	afb05e37d884fc	0	1
21	1391598	a78f18848550a2	195a30af54412f	9ac74b44f9da61	0	0
22	1861760	8ce910a270c624	01f3f60d62612c	69d619361cfc04	0	0
23	2281893	fb4820bfb8dcd2	c0ab52f7a902f7	a04d064c6573f1	2	0
24	1670186	a5b20e86171dd4	40ae649de9221b	0d9888f4645e60	0	0
25	3980392	1fdf906d152644	f00195b2d9d4ae	96979cb196a8eb	0	2

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

	Unnamed: 0	Id	groupId	matchId	assists	boosts
26	2239736	ffe0c4090cb8b4	729e23962383e3	4ad654434314df	0	2
27	1524762	e4c4a682901463	a260bad6273f10	6e5c1a0a0d7178	0	4
28	1560682	519c15946d2955	c89dda20dae66f	c152b30f489721	0	2
29	4402296	653ff0a2e9379a	038734e9095980	665dcf41d7c9dc	0	0
...	...	...	...	...	...	...
1111712	2156969	7a00c8222be933	63a0af0a80415f	f7417a09f65d58	0	0
1111713	2760844	79c80a1d10c430	74b2bce81c3bc5	088a24aefbaec6	1	4
1111714	3946818	ed64920197e5e5	6a8c92d369cb02	36810c05773c92	0	0
1111715	4298326	729efc24aab8f4	2d64d3e37a7d14	ad77c274378efb	0	1
1111716	3664219	1f276a7066cc01	4b6ba917e238d2	77b35f4635d4e6	0	0
1111717	1676427	f5c0468c4866d5	7cccd055ffaefe	072c16df049d48	1	3
1111718	2983592	5946a6d2b84747	94654723ace7ea	b9ea634476340c	1	6
1111719	3661404	09f36a7a2b1292	1d3946fdbbc3dd	20889f7ebaa234	0	0
1111720	2711931	97487c0bd531f0	a4e7730b773693	5b050493441ebf	0	0
1111721	2131535	529f421121a5ca	dbf930b13cc53a	925aa4cbbf441a	0	0
1111722	619954	90ce43c21ca401	fe3297fddbfa32	91aa551b4b76a6	0	3
1111723	2925622	992ffcec71aaa0	e96faa66efd3aa	ec2e688f651285	0	0
1111724	3054936	5fc1856ba940ff	ce018aa97ed235	e2f979e4b63bf5	0	1
1111725	3584733	e1132e9542113c	0bf134774af94e	77f66ab2b695d5	0	0
1111726	49496	1dc29b628c606c	61bc986ca70eb8	0457d313c80464	0	0
1111727	3478282	f908a2dafdc7ac	65946fec17034e	8ff7eb22d29207	0	0

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

	Unnamed: 0	Id	groupId	matchId	assists	boosts
1111728	4420005	0ca2712cb43a9c	248b2d28ef1b09	e574505645dcb2	0	2
1111729	78370	b5fa7d80480418	69675b37746877	d7014101ef2efd	1	4
1111730	627847	a92ddae489f6af	d7bdb668e0ff5a	6e9eebf99700d6	0	0
1111731	4378842	fafc64c04afc15	f2a39d4c19f529	22867907188e08	0	3
1111732	3865567	11986081581a4b	c6721e09b70b7e	432fe708b75af9	0	0
1111733	34444	e78e5ff96a9780	3f3e6497f02f15	dccb3ec00e243d	0	0
1111734	820478	6843f7438a5b1e	c00f94a2f82d74	d0b24a95f98534	1	1
1111735	820792	49397c8362ea49	5396945f3be420	cf3a8317cd5e9d	0	0
1111736	4412080	a87134976177e4	b22d88e51a064b	175b0cabe633c0	0	0
1111737	4209703	43739241d6a8e8	afb7c87045825e	ad5719d2422207	0	0
1111738	4311549	d57c8a22a9e38b	85e0624ddd2cb0	c3f50eef6dbb01	0	3
1111739	3521011	c7ad131758ba55	b3759687b385b3	990a8d68ec2bc4	0	0
1111740	2138826	8237d06ea2c508	13a05023ab7735	f93b1f0defaee5	1	4
1111741	609836	9beb0064c56445	5cfa1834dc5b11	0406ce4ab7a1bd	5	3

1111742 rows × 30 columns

In [4]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1111742 entries, 0 to 1111741
Data columns (total 30 columns):
Unnamed: 0          1111742 non-null int64
Id                  1111742 non-null object
groupId             1111742 non-null object
matchId             1111742 non-null object
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

```
assists            1111742 non-null int64
boosts             1111742 non-null int64
damageDealt        1111742 non-null float64
DBNOs              1111742 non-null int64
headshotKills      1111742 non-null int64
heals              1111742 non-null int64
killPlace          1111742 non-null int64
killPoints         1111742 non-null int64
kills              1111742 non-null int64
killStreaks        1111742 non-null int64
longestKill        1111742 non-null float64
matchDuration      1111742 non-null int64
matchType          1111742 non-null object
maxPlace           1111742 non-null int64
numGroups          1111742 non-null int64
rankPoints         1111742 non-null int64
revives            1111742 non-null int64
rideDistance       1111742 non-null float64
roadKills          1111742 non-null int64
swimDistance       1111742 non-null float64
teamKills          1111742 non-null int64
vehicleDestroys    1111742 non-null int64
walkDistance       1111742 non-null float64
weaponsAcquired    1111742 non-null int64
winPoints          1111742 non-null int64
winPlacePerc       1111741 non-null float64
dtypes: float64(6), int64(20), object(4)
memory usage: 254.5+ MB
```

```
In [5]: data.isnull().sum()
```

```
Out[5]: Unnamed: 0         0
        Id                 0
        groupId            0
        matchId            0
        assists            0
        boosts             0
        damageDealt        0
        DBNOs              0
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

```
headshotKills      0
heals              0
killPlace          0
killPoints         0
kills              0
killStreaks        0
longestKill        0
matchDuration      0
matchType          0
maxPlace           0
numGroups          0
rankPoints         0
revives            0
rideDistance       0
roadKills          0
swimDistance       0
teamKills          0
vehicleDestroys    0
walkDistance       0
weaponsAcquired    0
winPoints          0
winPlacePerc       1
dtype: int64
```

In [6]:

```
data.tail()
```

Out[6]:

	Unnamed: 0	Id	groupId	matchId	assists	boosts
1111737	4209703	43739241d6a8e8	afb7c87045825e	ad5719d2422207	0	0
1111738	4311549	d57c8a22a9e38b	85e0624ddd2cb0	c3f50eef6dbb01	0	3
1111739	3521011	c7ad131758ba55	b3759687b385b3	990a8d68ec2bc4	0	0
1111740	2138826	8237d06ea2c508	13a05023ab7735	f93b1f0defaee5	1	4
1111741	609836	9beb0064c56445	5cfa1834dc5b11	0406ce4ab7a1bd	5	3

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

5 rows × 30 columns

```
In [7]: data.columns
```

```
Out[7]: Index(['Unnamed: 0', 'Id', 'groupId', 'matchId', 'assists', 'boosts',
               'damageDealt', 'DBNOs', 'headshotKills', 'heals', 'killPlace',
               'killPoints', 'kills', 'killStreaks', 'longestKill', 'matchDurat
        ion',
               'matchType', 'maxPlace', 'numGroups', 'rankPoints', 'revives',
               'rideDistance', 'roadKills', 'swimDistance', 'teamKills',
               'vehicleDestroys', 'walkDistance', 'weaponsAcquired', 'winPoint
        s',
               'winPlacePerc'],
              dtype='object')
```

Exploratery data analysis in python

```
In [8]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import matplotlib as matplot
        import seaborn as sns
        %matplotlib inline
```

```
In [9]: #Correlation Matrix
        corr = data.corr()
        corr = (corr)
        sns.heatmap(corr,
                    xticklabels=corr.columns.values,
                    yticklabels=corr.columns.values)

        corr
```

Out[9]:

	Unnamed: 0	assists	boosts	damageDealt	DBNOs	headshotKil

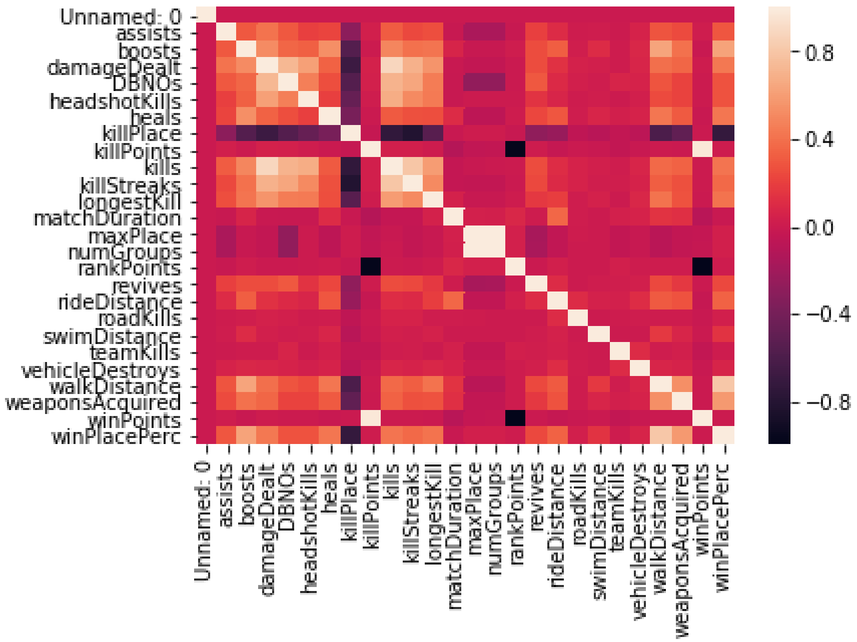
Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

	Unnamed: 0	assists	boosts	damageDealt	DBNOs	headshotKil
Unnamed: 0	1.000000	0.001488	0.000571	-0.000012	-0.000261	0.001353
assists	0.001488	1.000000	0.306582	0.409369	0.302200	0.199487
boosts	0.000571	0.306582	1.000000	0.519600	0.357903	0.331536
damageDealt	-0.000012	0.409369	0.519600	1.000000	0.735084	0.613306
DBNOs	-0.000261	0.302200	0.357903	0.735084	1.000000	0.467393
headshotKills	0.001353	0.199487	0.331536	0.613306	0.467393	1.000000
heals	-0.000650	0.228456	0.535388	0.341620	0.265348	0.197936
killPlace	0.000104	-0.289984	-0.554046	-0.676134	-0.554858	-0.468079
killPoints	0.000051	0.038331	0.008992	0.050217	0.042139	0.024203
kills	0.000249	0.322068	0.499478	0.889038	0.706211	0.673739
killStreaks	-0.000256	0.243386	0.403484	0.703222	0.646344	0.513207
longestKill	0.001358	0.260407	0.421444	0.561656	0.449521	0.444803
matchDuration	-0.000414	-0.019399	0.071471	-0.007605	-0.014080	-0.018887
maxPlace	-0.001337	-0.147629	-0.013297	-0.041203	-0.267302	0.008062
numGroups	-0.001359	-0.146552	-0.012529	-0.040570	-0.265376	0.008236
rankPoints	-0.000243	-0.015586	0.022659	-0.001615	-0.002738	0.005047
revives	-0.000484	0.197938	0.254125	0.256318	0.301098	0.149131
rideDistance	0.000627	0.110496	0.329281	0.139840	0.101882	0.076275
roadKills	0.000200	0.012366	0.033775	0.050660	0.035705	0.011974
swimDistance	0.000795	0.022871	0.107880	0.037509	0.018726	0.029988
teamKills	0.000071	0.007775	0.014645	0.016209	0.071295	0.009198

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

	Unnamed: 0	assists	boosts	damageDealt	DBNOs	headshotKil
vehicleDestroys	-0.000483	0.058241	0.085997	0.079901	0.059857	0.039857
walkDistance	-0.000273	0.288579	0.639361	0.396621	0.284051	0.249591
weaponsAcquired	-0.000137	0.244595	0.405301	0.354515	0.217667	0.215613
winPoints	0.000043	0.023341	-0.006973	0.017990	0.010483	0.005029
winPlacePerc	-0.000144	0.297738	0.633482	0.438973	0.279322	0.276170

26 rows × 26 columns



In [10]:
```
numeric_col=data.select_dtypes(exclude=['object'])
```

In [11]:
```
charcter_col =data.select_dtypes(include=['object'])
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

In [12]:
```
# Plot normal probability plot
```

In [13]:
```
numeric_col.head()
charcter_col.head()
```

Out[13]:

	Id	groupId	matchId	matchType
0	a3e3cea9f2e314	c9f6eaa81245b5	a3d8693390584c	squad-fpp
1	43e4be7ad72cc7	521836de101ee8	b65de9055327e0	squad-fpp
2	9a68690e31fdff	4a72ffa2cebd90	d6aad3f9830e60	solo-fpp
3	b147e1bd448fc4	9a8991656b3fea	d931c0932d8aca	squad-fpp
4	d818b4edd59612	eece87c8b846b3	ec2b5ed94baae3	solo

In [14]:
```
data.winPlacePerc.value_counts()
```

Out[14]:
```
0.0000    55369
1.0000    31853
0.5000    13790
0.3333    10586
0.6667     9608
0.2500     8262
0.1111     7378
0.7500     7288
0.0370     7273
0.2222     6882
0.0741     6792
0.1481     6465
0.1852     6255
0.1429     6225
0.2593     5903
0.0385     5892
0.4444     5852
0.8889     5841
0.0769     5772
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

```
0.2963     5768
0.2857     5726
0.0714     5718
0.0357     5704
0.7778     5612
0.5556     5534
0.1538     5394
0.1071     5393
0.1154     5388
0.3704     5343
0.8571     5309
          ...
0.9494        1
0.2679        1
0.4219        1
0.4754        1
0.7639        1
0.9324        1
0.7571        1
0.3016        1
0.8113        1
0.0290        1
0.7424        1
0.5690        1
0.5932        1
0.5811        1
0.7403        1
0.0886        1
0.2923        1
0.0508        1
0.8784        1
0.1077        1
0.6406        1
0.2429        1
0.9836        1
0.7091        1
0.8228        1
0.2203        1
0.7564        1
0.2787        1
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

```
0.9492        1
0.2568        1
Name: winPlacePerc, Length: 2557, dtype: int64
```

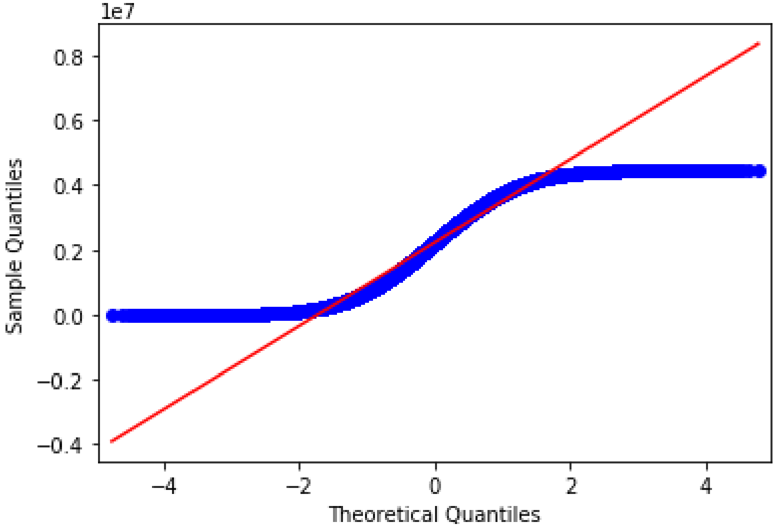
In [15]:

```
from statsmodels.graphics.gofplots import qqplot
from matplotlib import pyplot
```

In [16]:

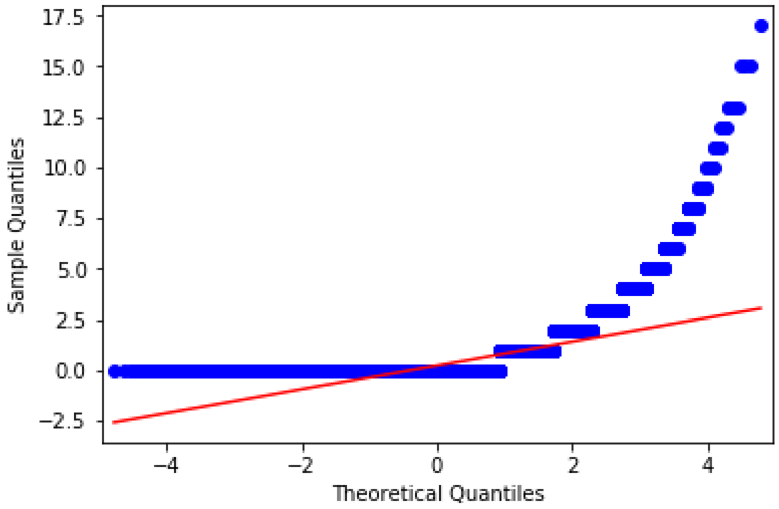
```
# plot one by one column in dataframe
for i in numeric_col:
 print("--------------------")
 print(i)
 # To check the normality by graph
 qqplot(data[i], line='s')
 pyplot.show()
```

```
--------------------
Unnamed: 0
```

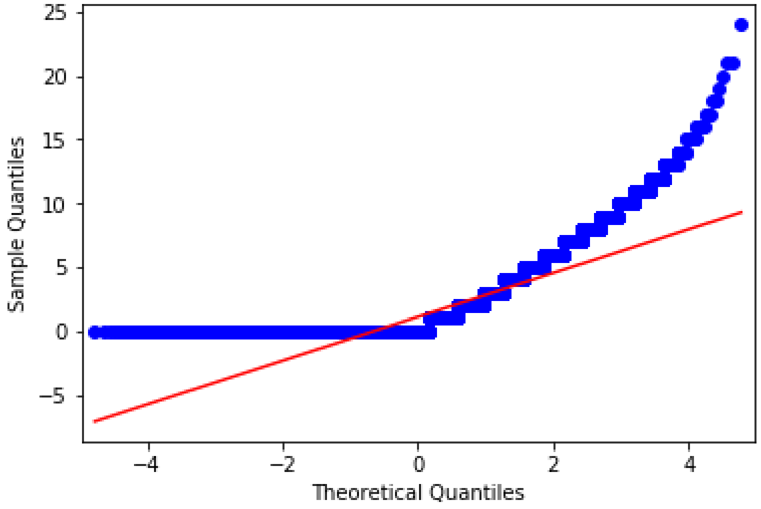


```
--------------------
assists
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

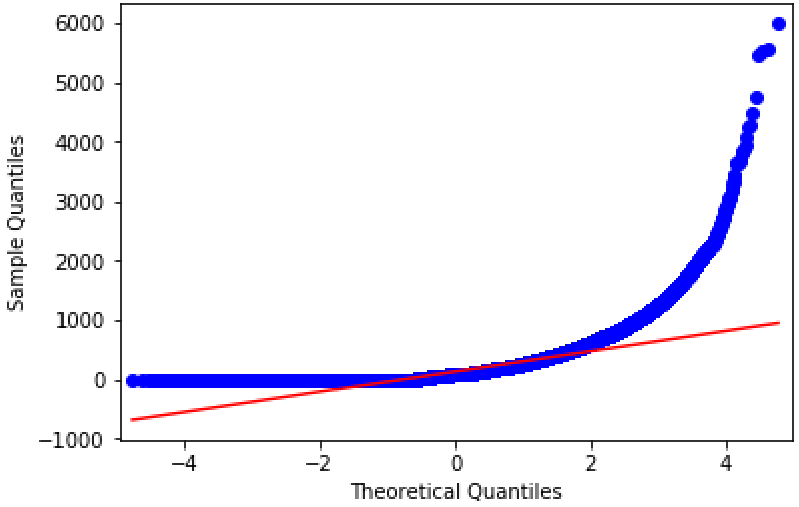


```
--------------------
boosts
```

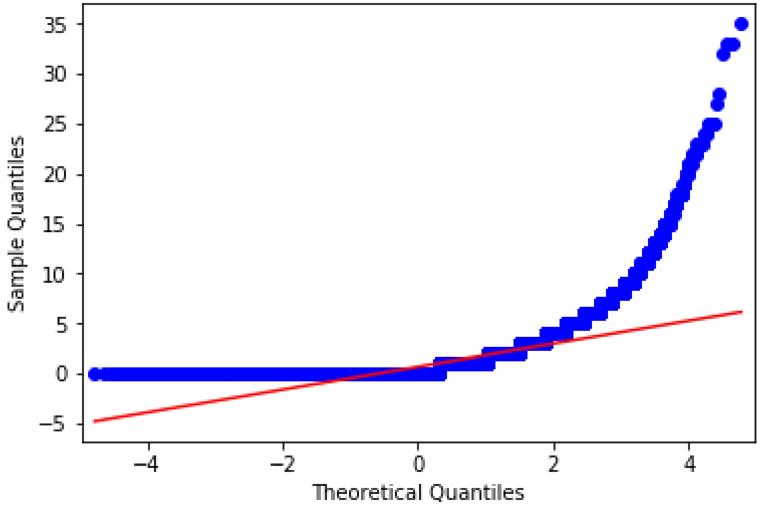


```
--------------------
damageDealt
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

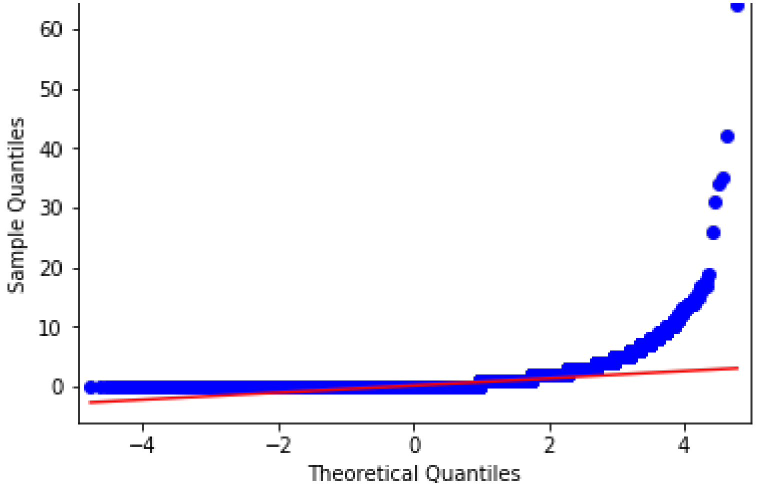


```
--------------------
DBNOs
```

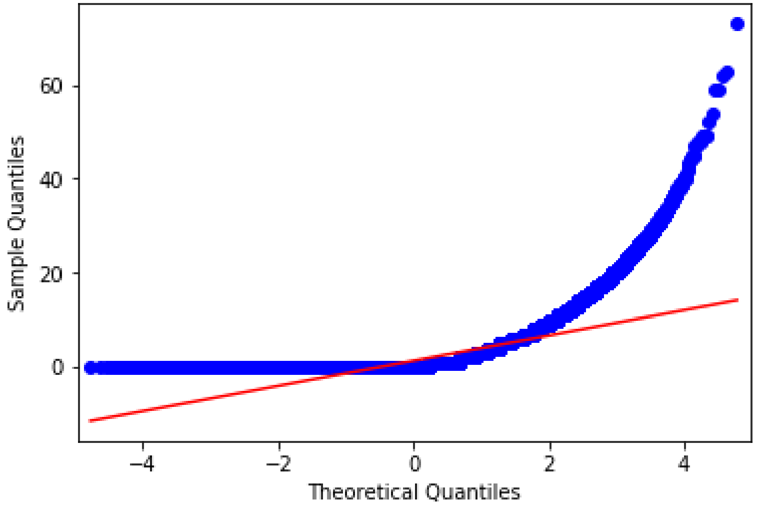


```
--------------------
headshotKills
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

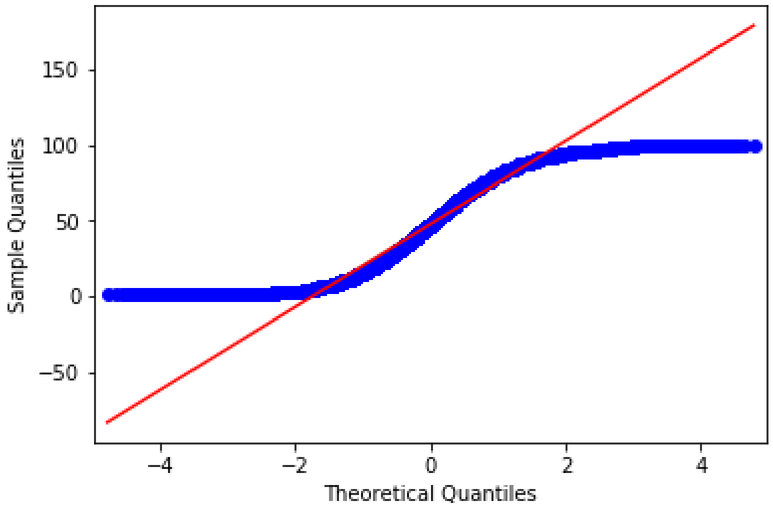


```
--------------------
heals
```

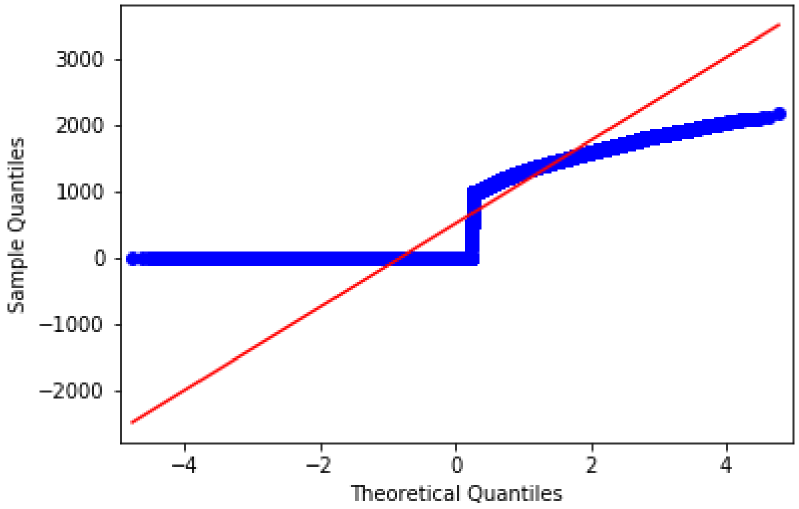


```
--------------------
killPlace
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

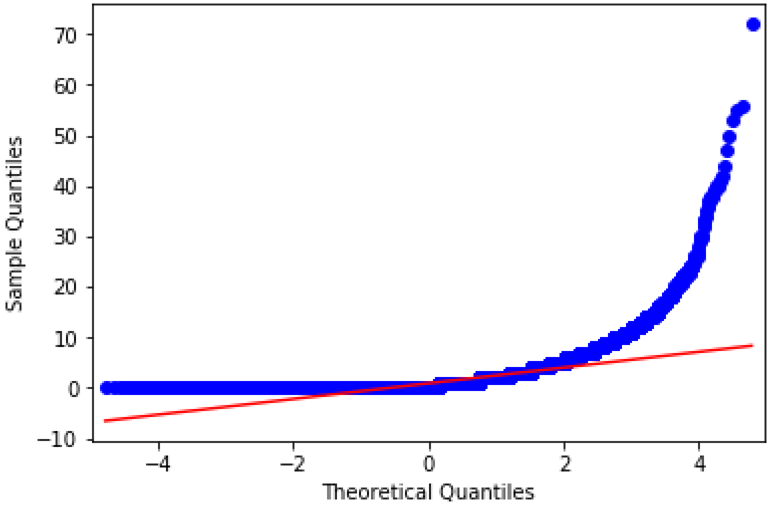


```
--------------------
killPoints
```

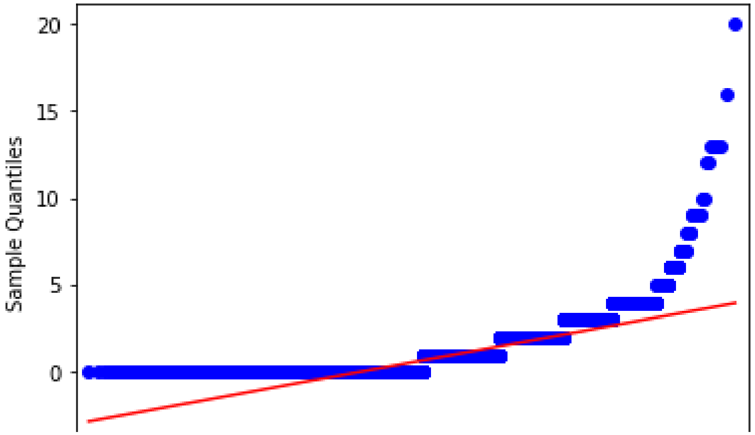


Create PDF in your applications with the Pdfcrowd HTML to PDF API
PDFCROWD

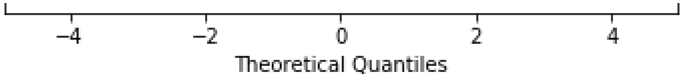
```
--------------------
kills
```



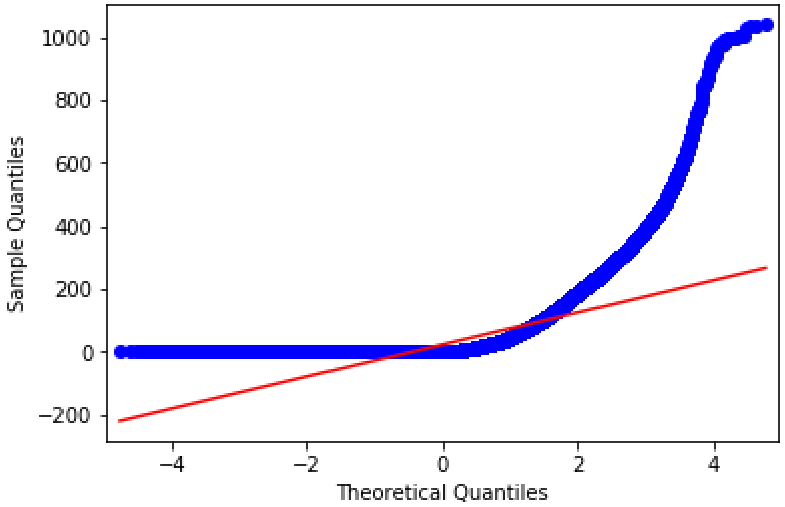
```
--------------------
killStreaks
```



Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

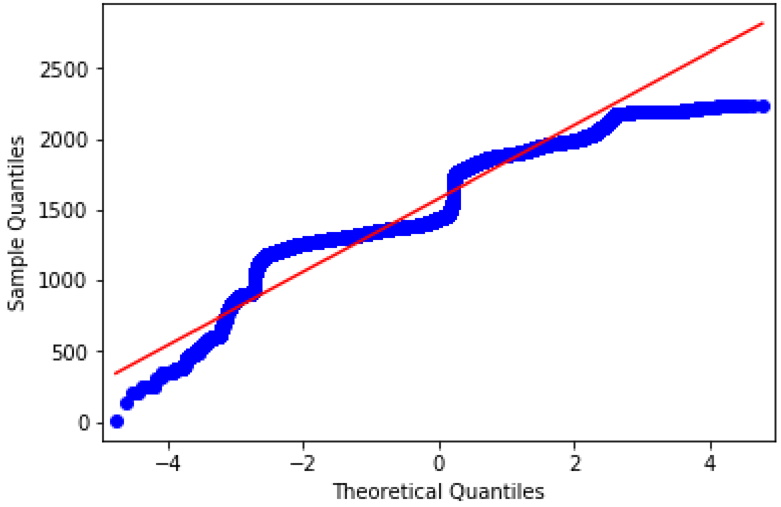


```
--------------------
longestKill
```

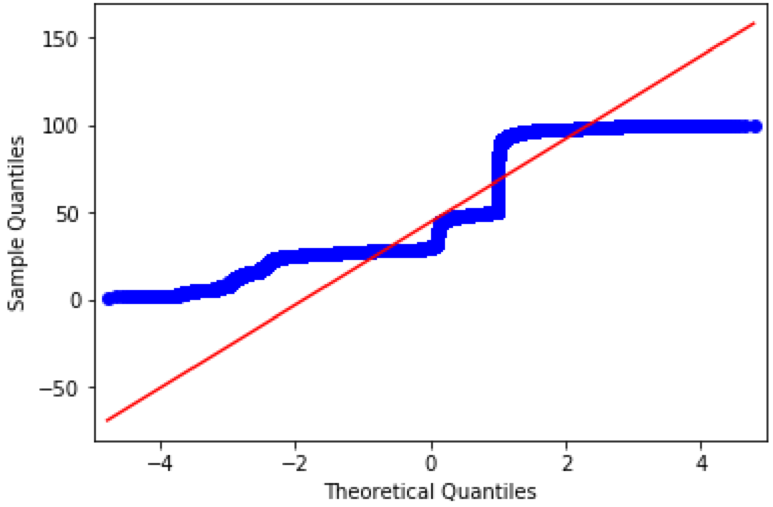


```
--------------------
matchDuration
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

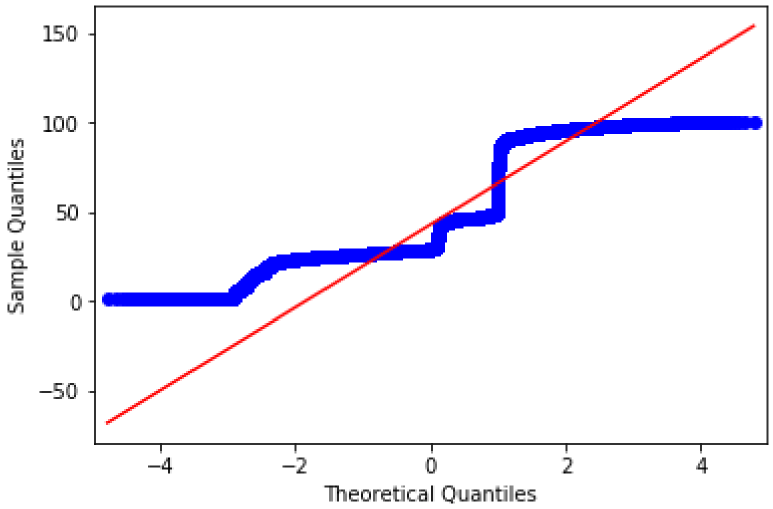


```
--------------------
maxPlace
```



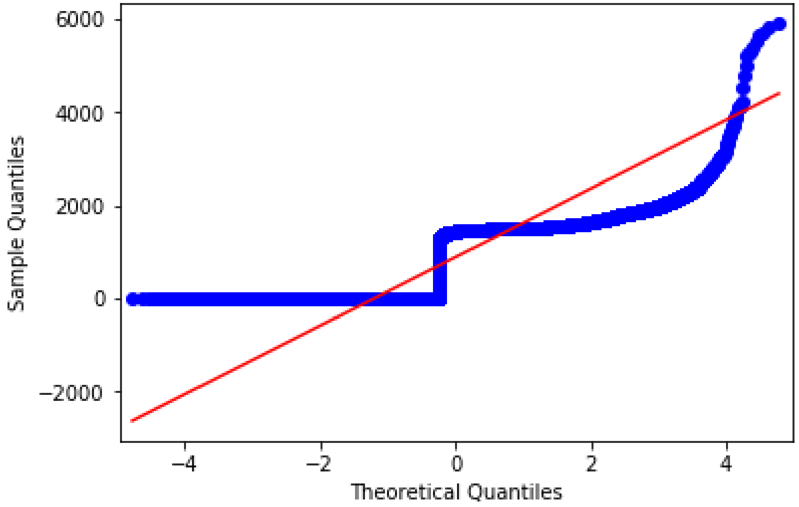
```
--------------------
numGroups
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

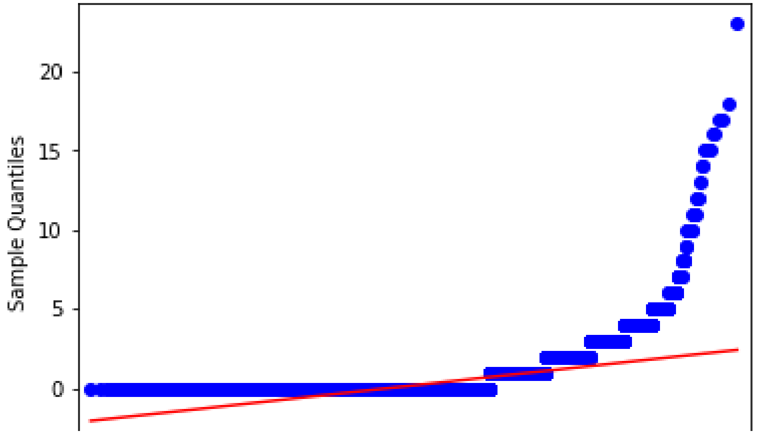


```
--------------------
rankPoints
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



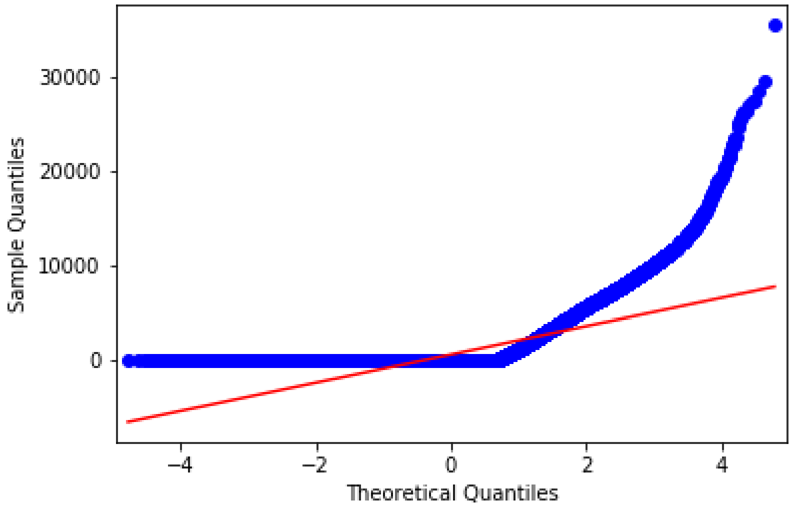
```
--------------------
revives
```



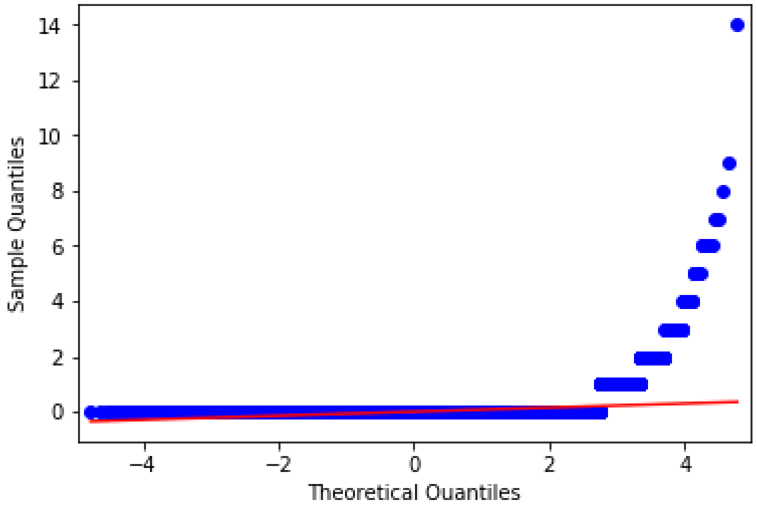
```
--------------------
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

```
rideDistance
```



```
--------------------
roadKills
```

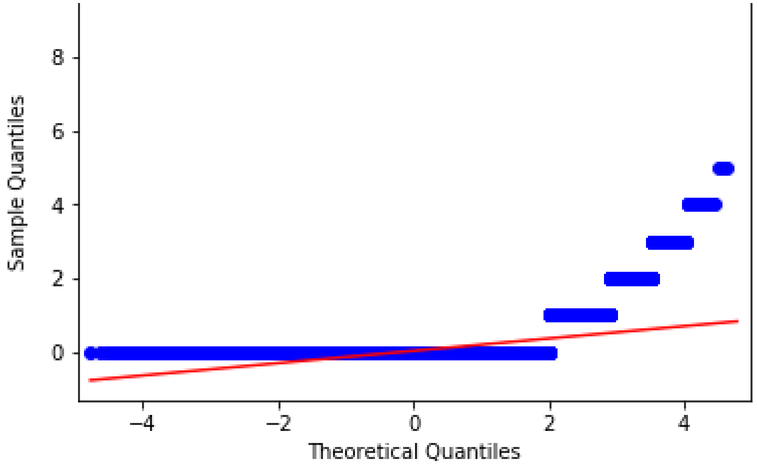


Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

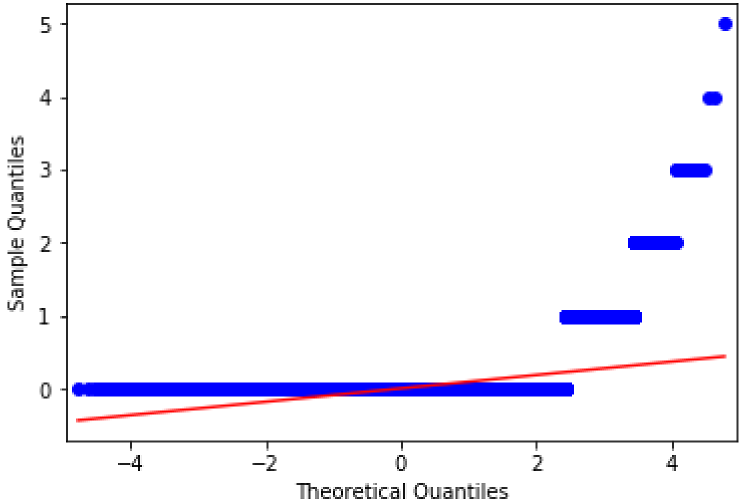
```
--------------------
swimDistance
```

```
--------------------
teamKills
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

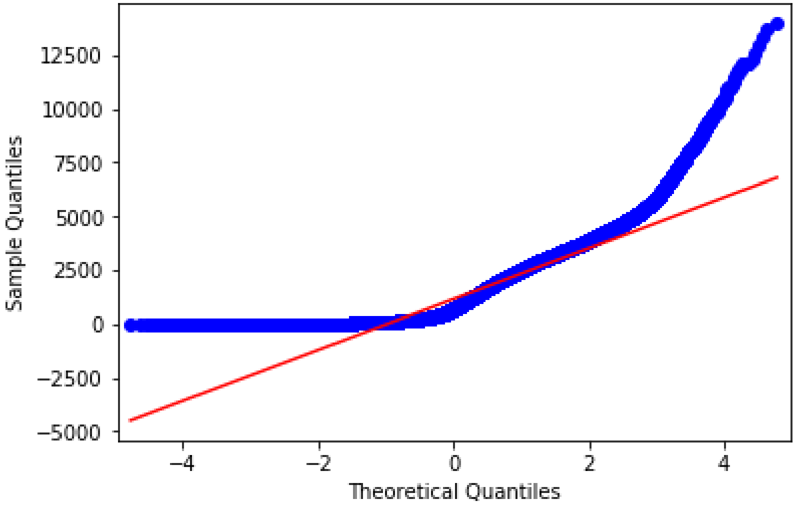


```
--------------------
vehicleDestroys
```



Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

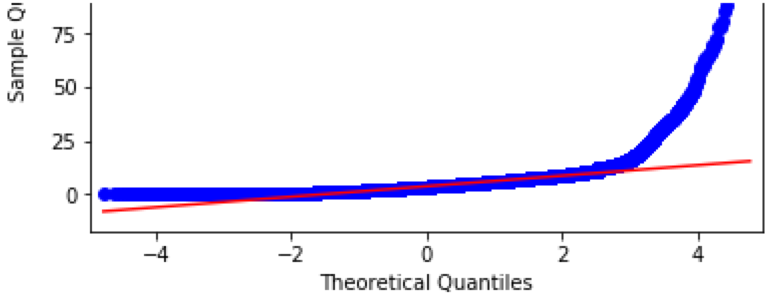
```
--------------------
walkDistance
```



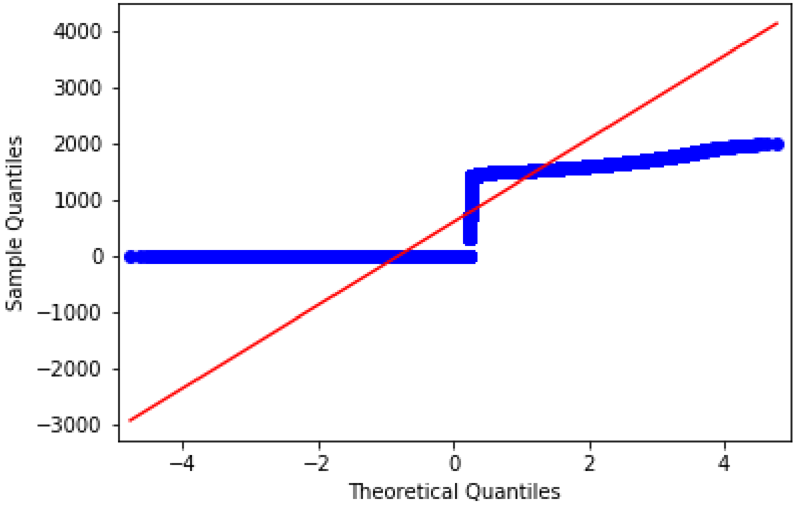
```
--------------------
weaponsAcquired
```

175
150
125
100
uantiles

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

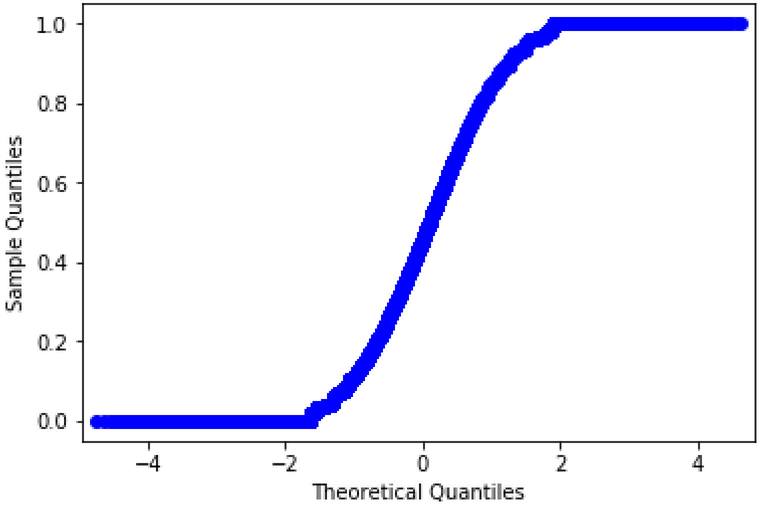


```
--------------------
winPoints
```



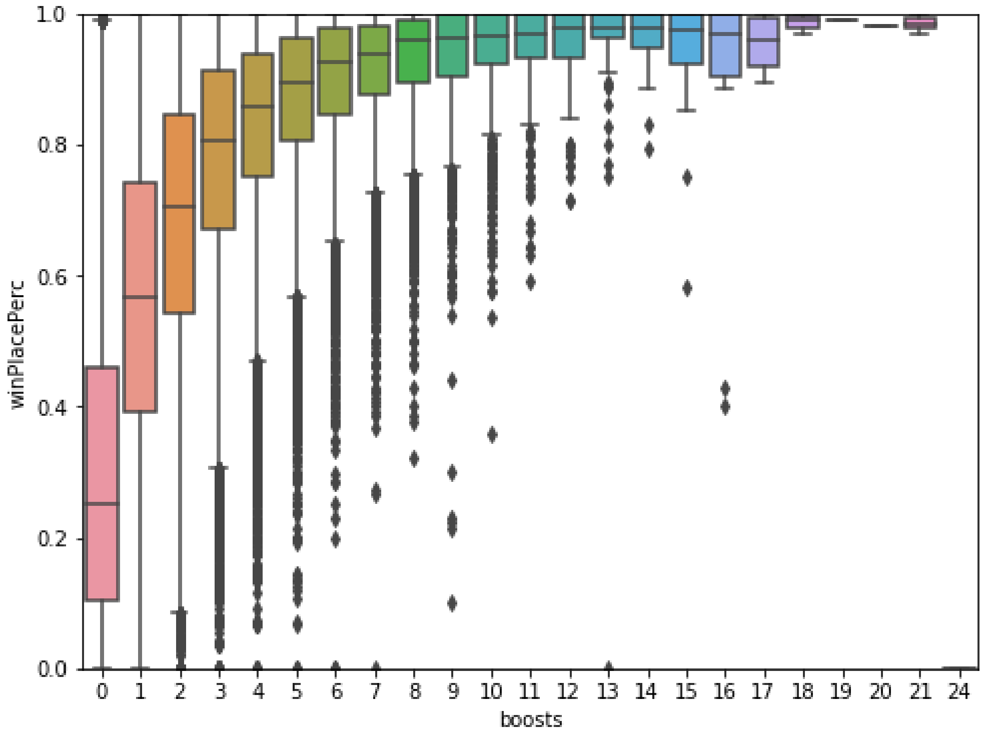
Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

```
--------------------
winPlacePerc
```



In [18]:
```
f, ax = plt.subplots(figsize=(8, 6))
fig = sns.boxplot(x='boosts', y="winPlacePerc", data=data)
fig.axis(ymin=0, ymax=1);
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



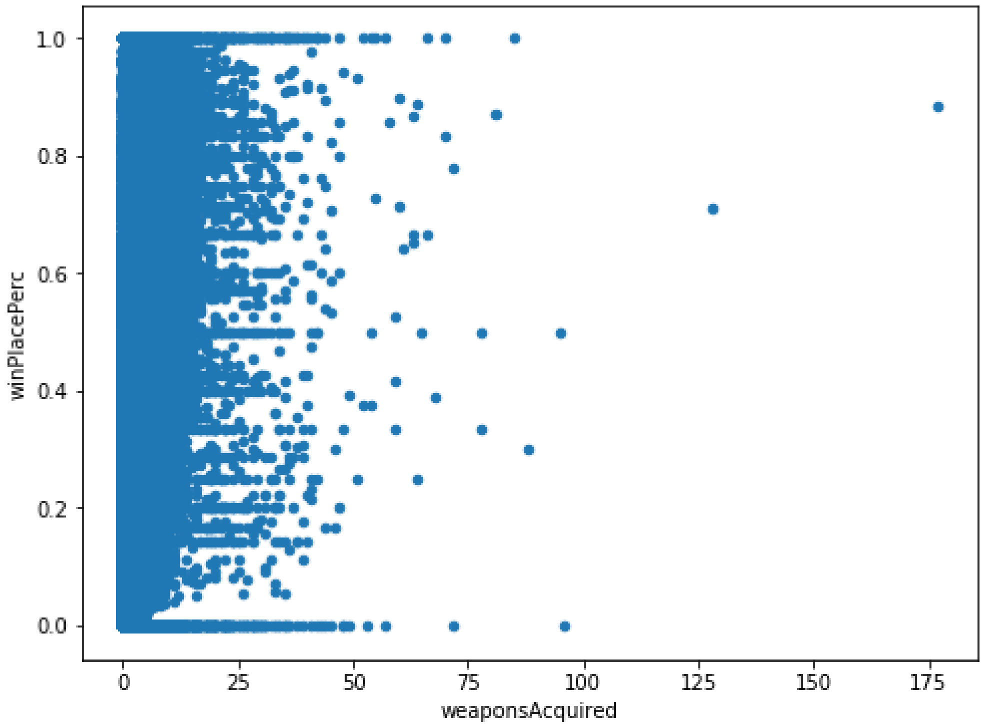
In [19]:

```
data.plot(x="weaponsAcquired",y="winPlacePerc", kind="scatter", figsize
= (8,6))
```

Out[19]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x52107390>
```

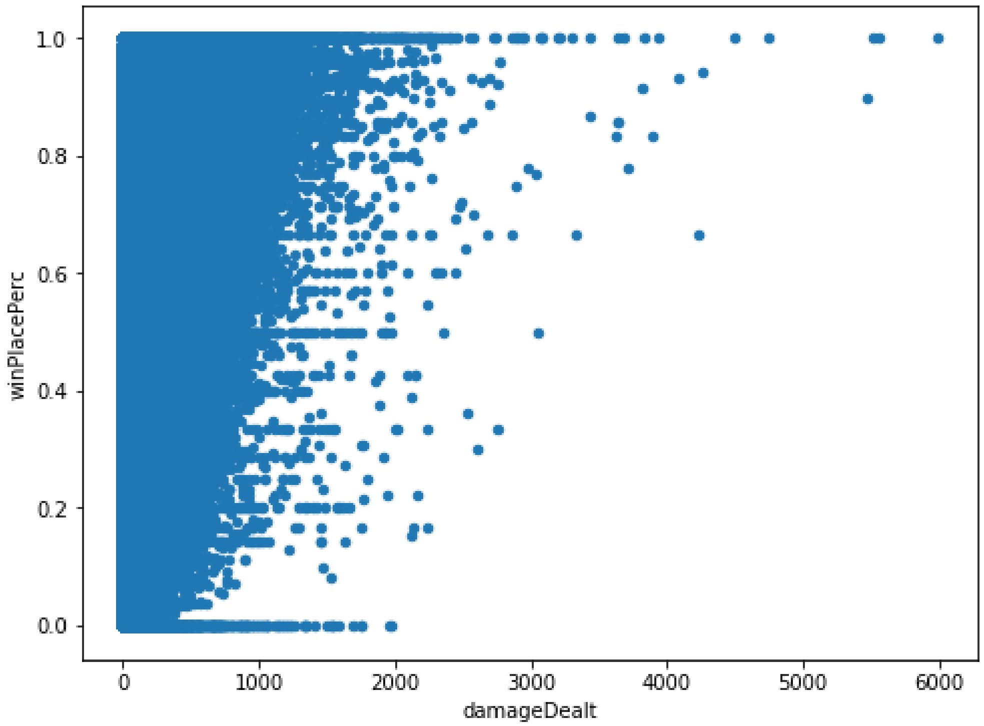
Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



```
In [20]: data.plot(x="damageDealt",y="winPlacePerc", kind="scatter", figsize = (
         8,6))
```

```
Out[20]: <matplotlib.axes._subplots.AxesSubplot at 0x5b691208>
```

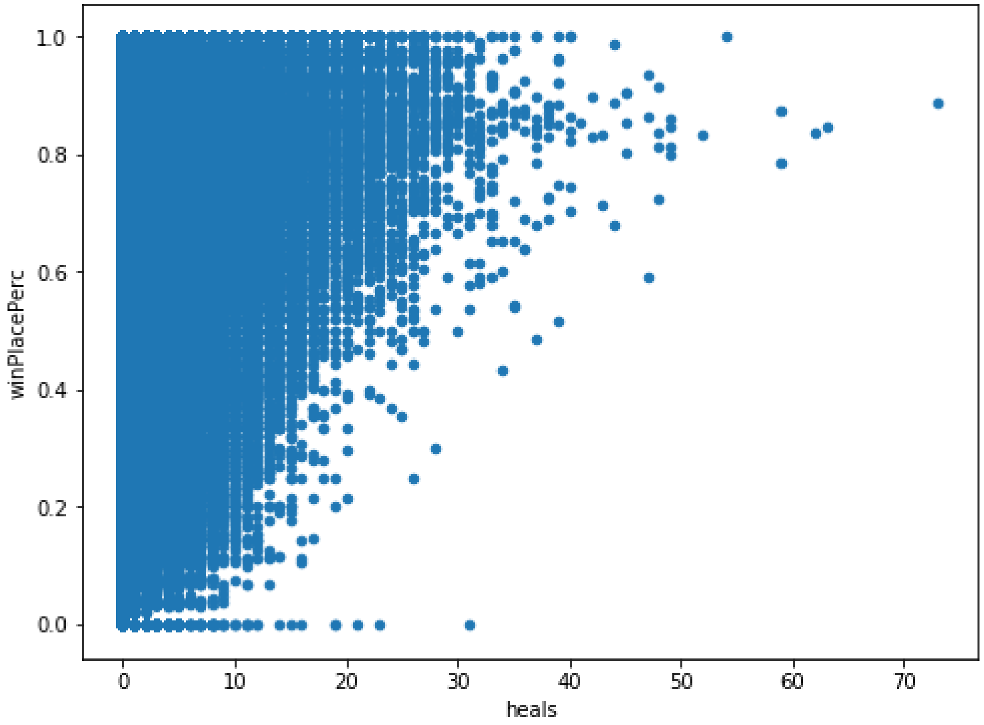
Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



```
In [21]: data.plot(x="heals",y="winPlacePerc", kind="scatter", figsize = (8,6))
```

```
Out[21]: <matplotlib.axes._subplots.AxesSubplot at 0x510b8c18>
```

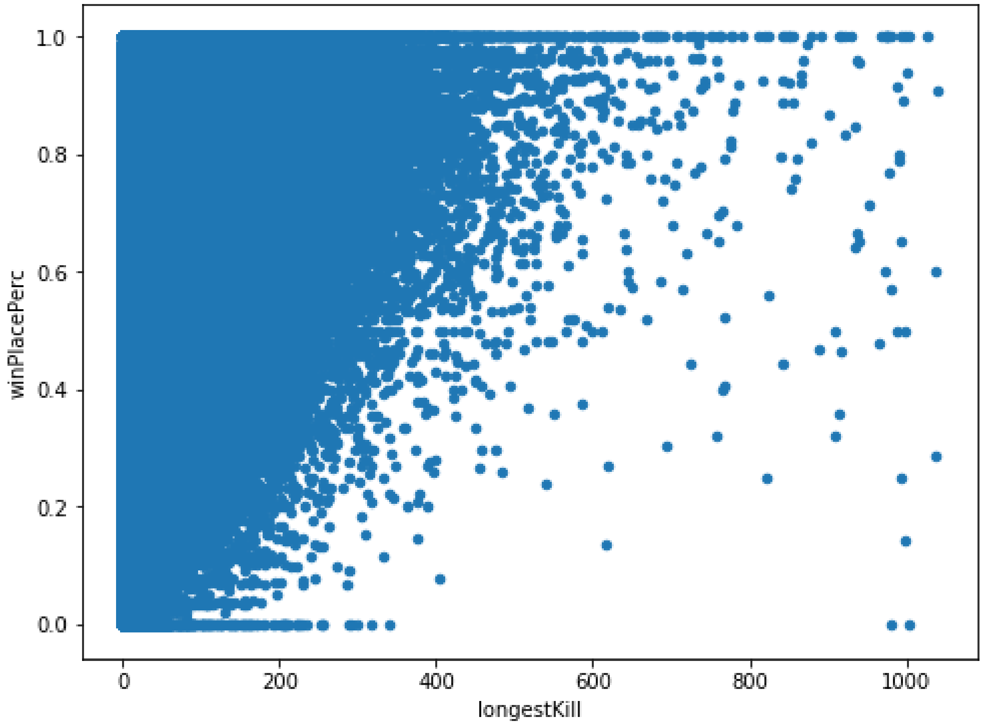
Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



```
In [22]: data.plot(x="longestKill",y="winPlacePerc", kind="scatter", figsize = (
         8,6))

Out[22]: <matplotlib.axes._subplots.AxesSubplot at 0x51a1eac8>
```

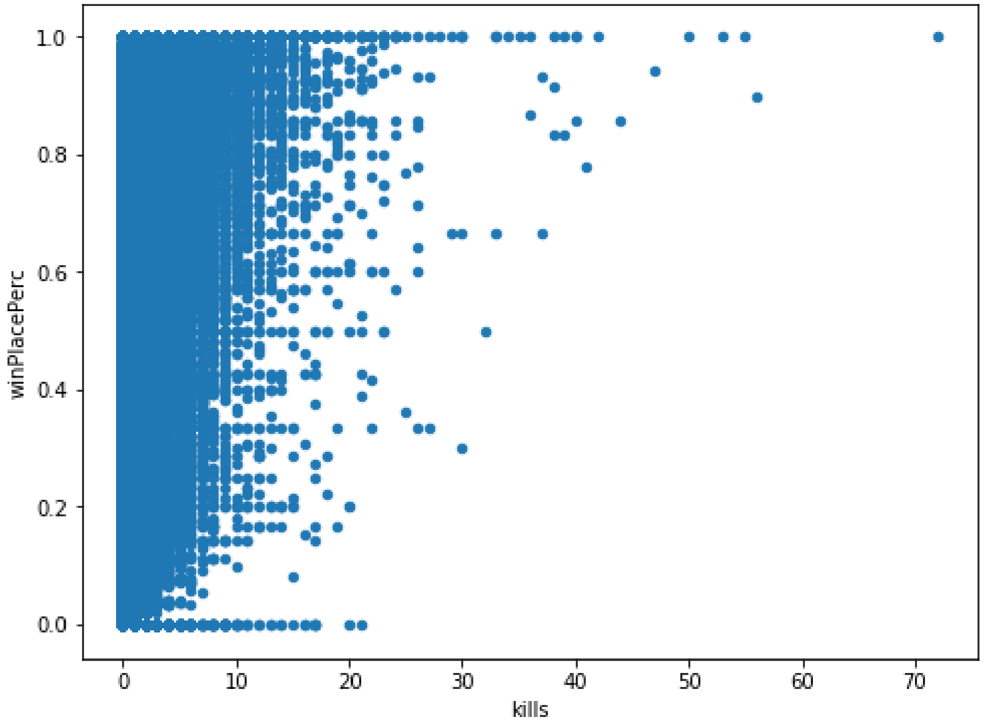
Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



```
In [23]: data.plot(x="kills",y="winPlacePerc", kind="scatter", figsize = (8,6))

Out[23]: <matplotlib.axes._subplots.AxesSubplot at 0x58b0b908>
```

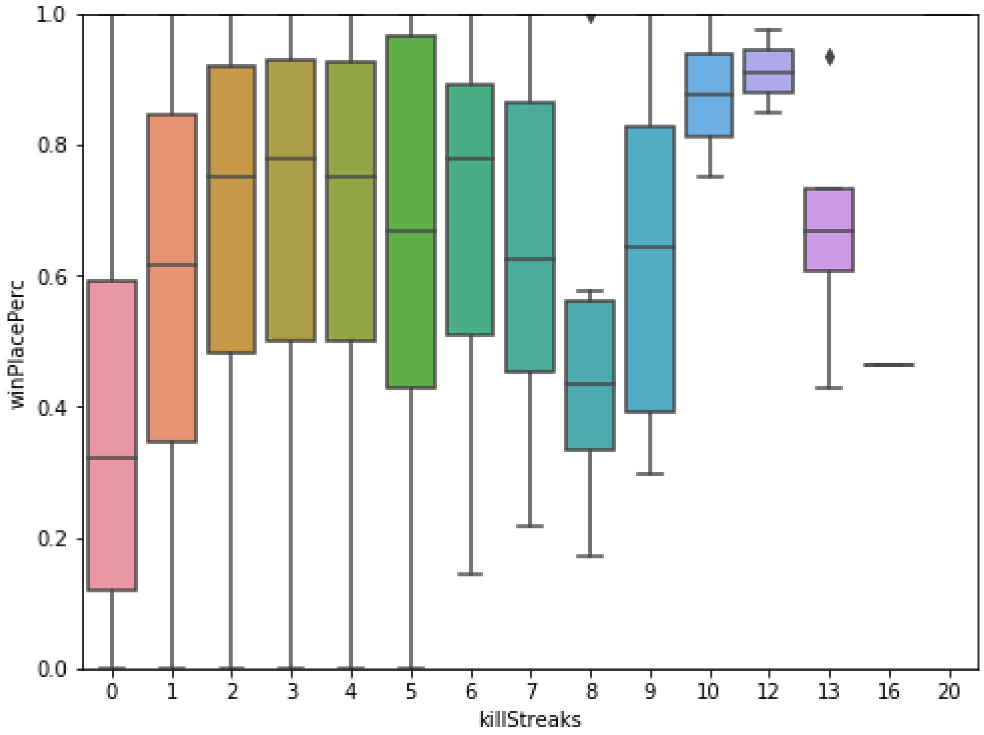
Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



In [24]:
```
f, ax = plt.subplots(figsize=(8, 6))
fig = sns.boxplot(x='killStreaks', y="winPlacePerc", data=data)
fig.axis(ymin=0, ymax=1)
```

Out[24]:
```
(-0.5, 14.5, 0, 1)
```

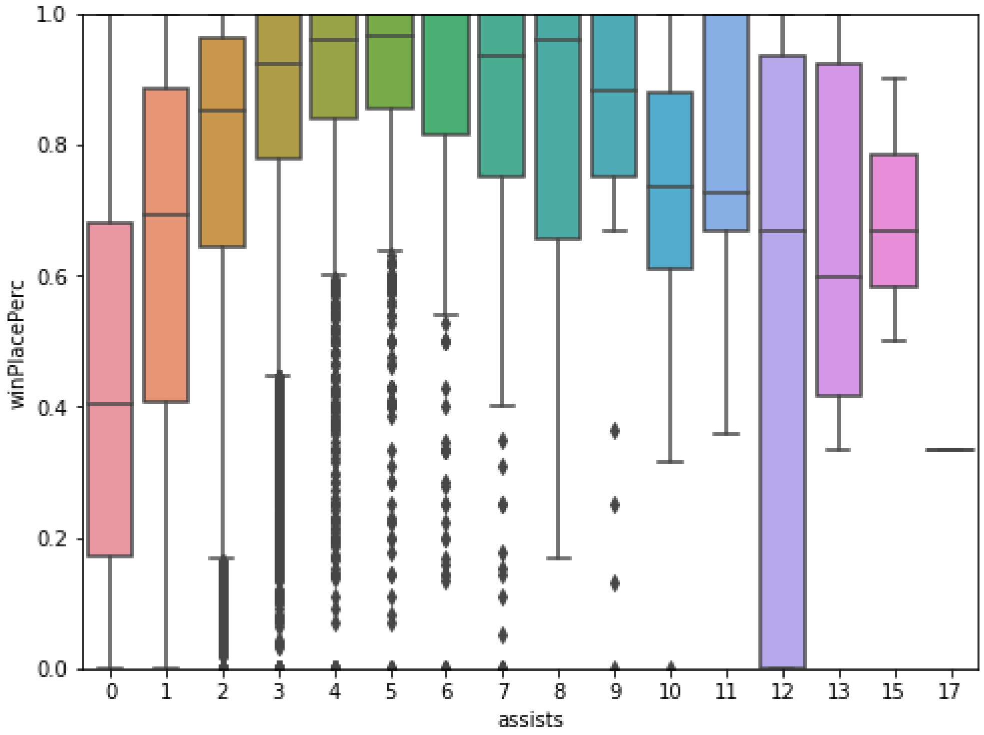
Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



In [25]:

```
f, ax = plt.subplots(figsize=(8, 6))
fig = sns.boxplot(x='assists', y="winPlacePerc", data=data)
fig.axis(ymin=0, ymax=1);
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

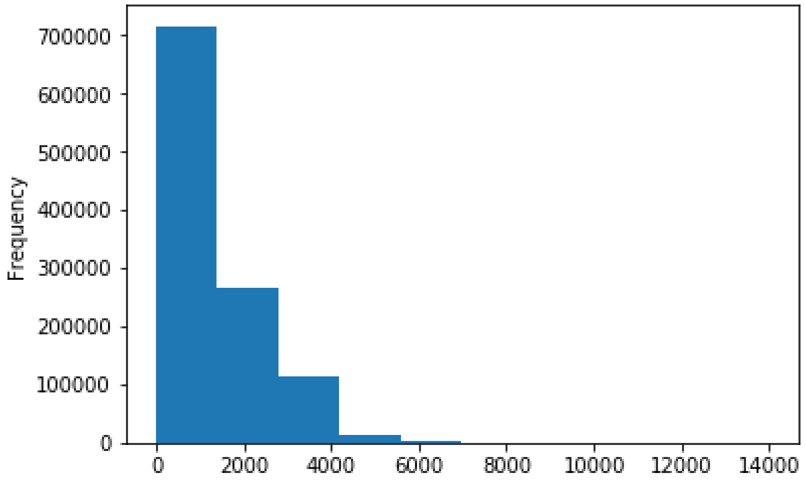


In [4]:
```
data.columns
```

Out[4]:
```
Index(['Unnamed: 0', 'Id', 'groupId', 'matchId', 'assists', 'boosts',
       'damageDealt', 'DBNOs', 'headshotKills', 'heals', 'killPlace',
       'killPoints', 'kills', 'killStreaks', 'longestKill', 'matchDurat
ion',
       'matchType', 'maxPlace', 'numGroups', 'rankPoints', 'revives',
       'rideDistance', 'roadKills', 'swimDistance', 'teamKills',
       'vehicleDestroys', 'walkDistance', 'weaponsAcquired', 'winPoint
s',
       'winPlacePerc'],
      dtype='object')
```

In [6]:
```
data['c'].plot.hist()
plt.show()
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



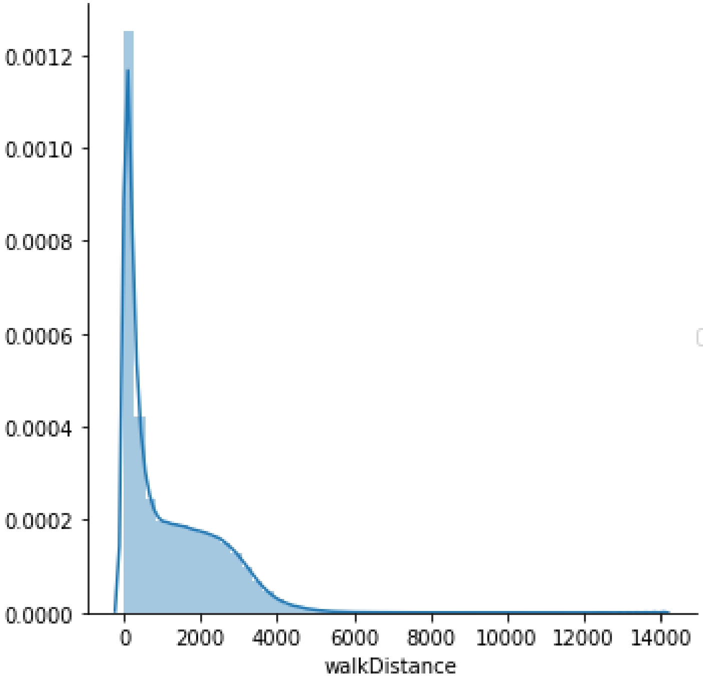
In [7]:

```
sns.FacetGrid(data, size=5)\
   .map(sns.distplot, "walkDistance")\
   .add_legend()

plt.show()
```

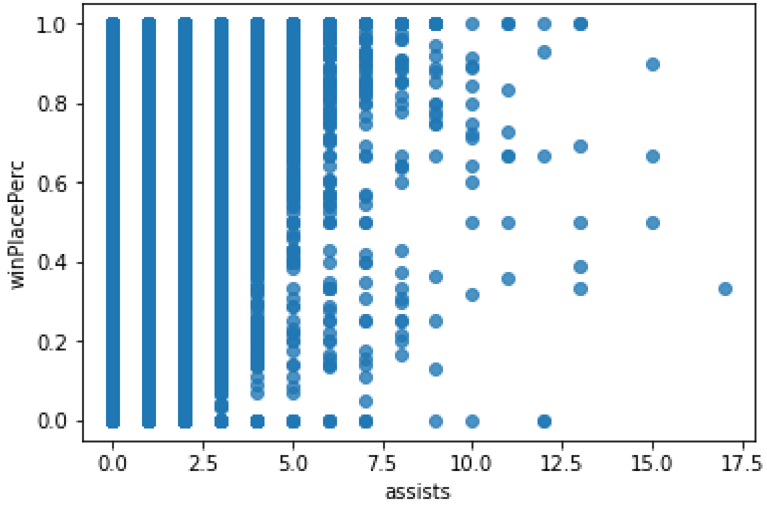
```
C:\Users\hello\Anaconda3\lib\site-packages\matplotlib\axes\_axes.py:646
2: UserWarning: The 'normed' kwarg is deprecated, and has been replaced
by the 'density' kwarg.
  warnings.warn("The 'normed' kwarg is deprecated, and has been "
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



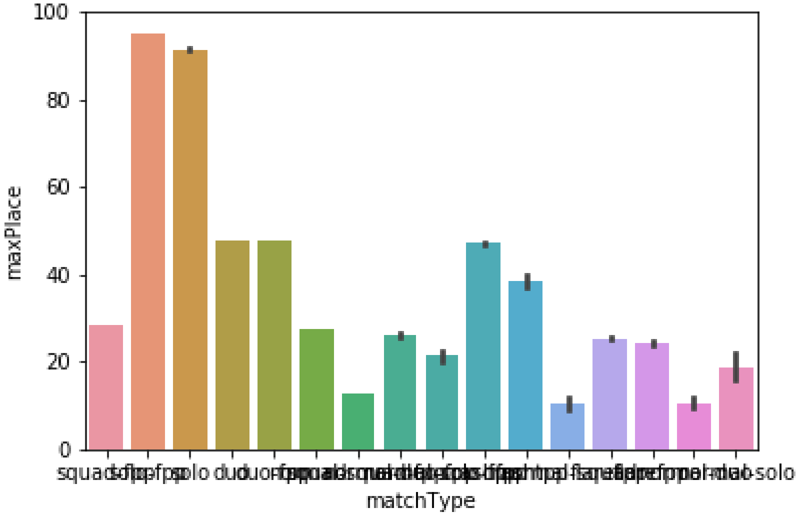
In [8]:
```
sns.regplot(x='assists', y='winPlacePerc', data=data, fit_reg=False)
plt.show()
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD



In [4]:
```
sns.barplot(x = 'matchType',
            y = 'maxPlace',
            data = data)

# Show the plot
plt.show()
```



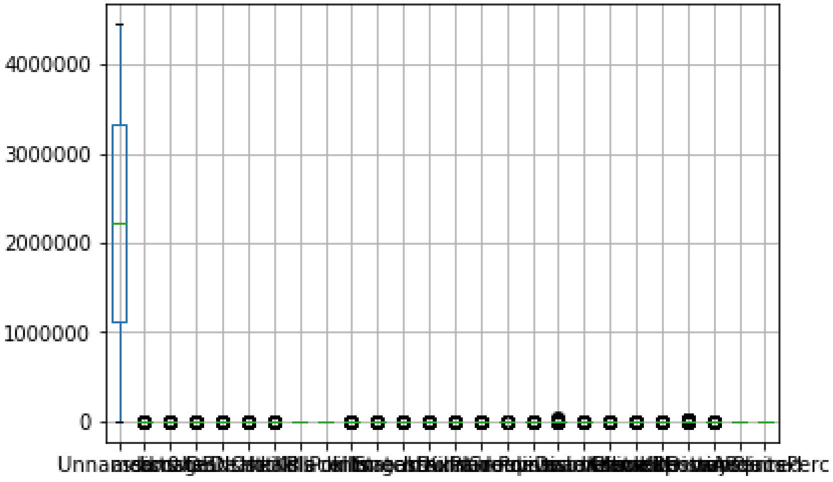
Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

In [6]:

```
data.boxplot()
```

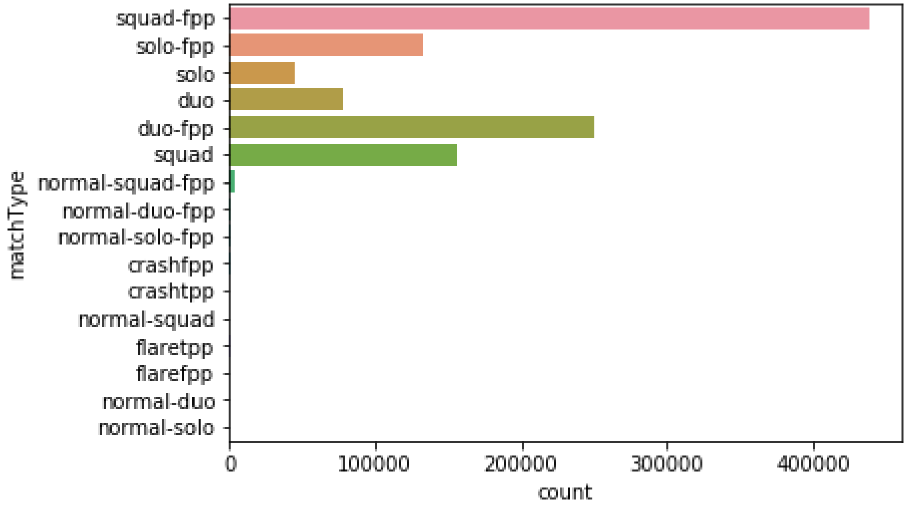
Out[6]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x74ffb630>
```



In [8]:

```
sns.countplot(y="matchType", data=data);
```



Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD

In [ ]:
```
# Chi square test hypothesis:
# Ho :- Varibles are independent.
# H1 :- Varibles are dependent.
```

In [18]:
```
def check_categorical_dependency(independent,dependent,confidence_interval):
 cross_table = pd.crosstab(data[independent], data[dependent],margins=True)
 stat, p, dof, expected = stats.chi2_contingency(cross_table)
 print ("Chi-Square Statistic value = {}".format(stat))
# print ("P - Value = {}".format(p))
 alpha = 1.0 - confidence_interval
# if p <= alpha:
# print('Variables are depndent')
# else:
# print('Variables are Independent')
 return p,alpha
```

In [20]:
```
 p,a=check_categorical_dependency('matchType',"winPlacePerc",0.95)
```

```
Chi-Square Statistic value = 1632541.9458471988
```

In [ ]:
```
for i in X:
 p,confi_level=check_categorical_dependency(i,"winPlacePerc",0.95)
 if p <= confi_level:
 print(i+" and winPlacePerc are dependet on each other.")
 else:
 print(i+" and winPlacePerc are independet on each other.")
```

Create PDF in your applications with the Pdfcrowd HTML to PDF API PDFCROWD