Settembre 2017

Esercizio 0

0.1 Il valore atteso di A - B è uguale a u - u = 0

0.2

$$Var(A-B) = E[(A-B)^2] - E[A-B]^2$$

$$Var(A-B) = E[A^2 - 2AB + B^2]$$

$$Var(A-B) = E[A^2] - 2E[AB] + E[B^2]$$

Perché A e B variabili indipendenti

$$Var(A-B) = E[A^2] - 2E[A]E[B] + E[B^2]$$

$$Var(A-B) = (E[A^2] - u^2) + (E[B^2] - u^2)$$

$$Var(A-B) = Var(A) + Var(B)$$

$$Var(A-B) = 2\sigma^2$$

0.3 La distribuzione di A - B sarebbe a sua volta una variabile aleatoria normale perché somma di variabili aleatorie normali. I parametri sarebbero: valore atteso 0 e varianza 2o^2

0.4

$$T_x = \frac{\sum_{i=1}^{n} (X_i - u)^2}{n}$$

0.5 Lo stimatore è non distorto perché il valore atteso di ogni (Xi -u) è uguale alla varianza di X per definizione:

$$E[(X_i - u)^2] = Var(X)$$

$$E[T_x] = \frac{n * Var(X)}{n} = Var(X)$$

0.6

$$Y = A - B$$

$$T_y = 2T_x$$

0.7

$$E[T_y] = E[2T_x]$$

$$E[T_y] = 2E[T_x]$$

$$E[T_y] = 2\sigma^2$$

Esercizio 1

In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
import scipy.stats as st

df = pd.read_csv('SF_Park_Scores.csv', sep=',', decimal='.')
df[:5]
```

Out[1]:

	ParkID	PSA	Park	FQ	Score	Facility Type	Facility Name	Address	State	Z
0	86	PSA4	Carl Larsen Park	FY05Q3	0.795	Basketball Court	Ocean View Basketball Courts	Capitol & Montana St	CA	9
1	13	PSA4	Junipero Serra Playground	FY05Q3	0.957	Ball Field	Glen ball fields	Diamond & Farnum Street	CA	9
2	9	PSA4	Rolph Nicol Playground	FY05Q3	0.864	Dog Play Area	Douglass dog play area	26th & Douglass Street	CA	9
3	117	PSA2	Alamo Square	FY05Q4	0.857	Restroom	Gilman Bathrooms	Gilman Ave & Griffith	CA	9
4	60	PSA6	Jose Coronado Playground	FY05Q4	0.859	Basketball Court	GGP1 Panhandle Basketball Courts	Stanyan & Great Hwy	CA	9

In [7]:

```
# 1.2
df.describe()
```

Out[7]:

	ParkID	Score	Zipcode	Floor Count	Square Feet	Per
count	5494.000000	5494.000000	4719.000000	1324.000000	4719.000000	4719.(
mean	32991.238260	0.897962	94117.015469	1.205438	26631.239099	548.7
std	150843.356703	0.117428	7.789351	0.555411	63124.930195	793.5
min	1.000000	0.000000	94102.000000	1.000000	213.120658	60.72
25%	55.000000	0.859000	94112.000000	1.000000	1379.550956	169.1
50%	106.000000	0.931000	94116.000000	1.000000	4241.343735	285.1
75%	154.000000	0.976000	94122.000000	1.000000	10192.913376	513.5
max	957226.000000	1.000000	94134.000000	4.000000	515443.479217	5506.

In [8]:

```
# 1.3
len(df)
```

Out[8]:

```
5494
```

In [9]:

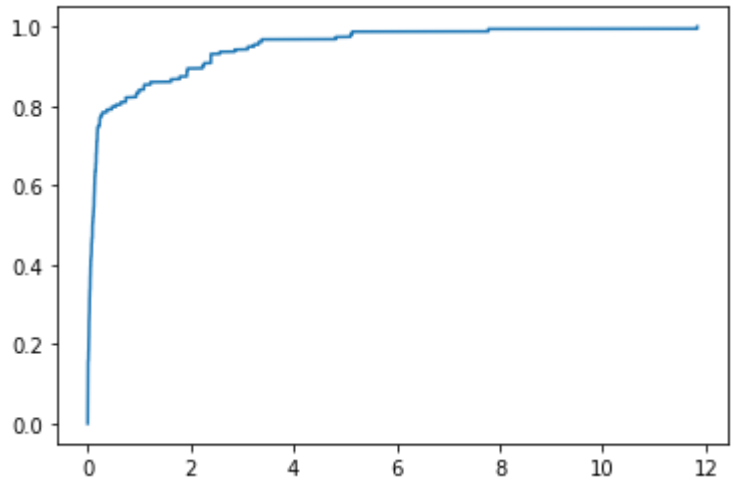
```
# 1.4
(df['PSA'].unique(), len(df['PSA'].unique()))
```

Out[9]:

```
(array(['PSA4', 'PSA2', 'PSA6', 'PSA3', 'GGP', 'PSA1', 'PSA5'],
       dtype=object), 7)
```

In [10]:

```
# 1.5
from statsmodels.distributions.empirical_distribution import ECDF
dist = ECDF(df.Acres.dropna())
plt.plot(dist.x, dist.y)
plt.show()
```



In [11]:

```
# 1.6
    # La metà dei parchi di San Frascisco ha una estensione maggiore di 0.097368 acri
```

In [12]:

```
# 1.7
    # estensione media: 0.611372 acri
```

In [13]:

```
# 1.8
    # media e mediana non sono simili, quindi probabilmente la distribuzione non seguir
à quindi una legge normale.
    # il grafico sarà quindi asimmetrico con una coda a destra
```

In [14]:

```
# 1.9
len(df[df.Acres < 50].dropna())
```

Out[14]:

1324

Esercizio 2

In [15]:

```
# 2.1
# -122.442014
# 37.755449
# 0.032165^2
# 0.025242^2
```

In [16]:

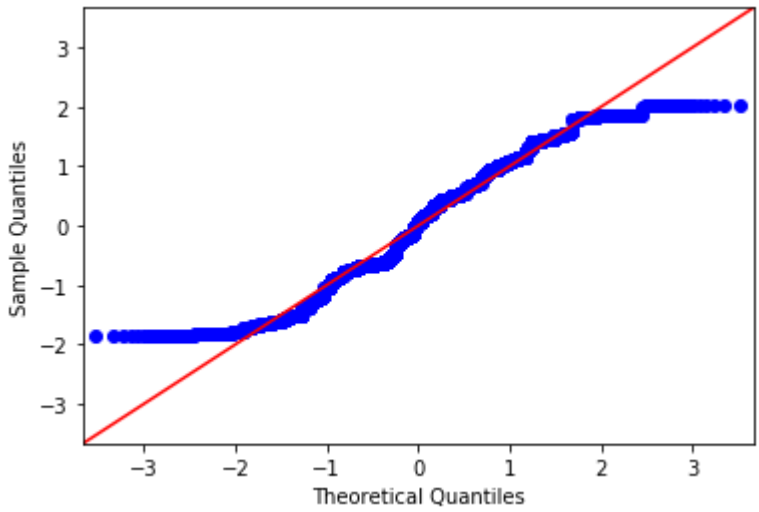
```
# 2.2
#!?
```

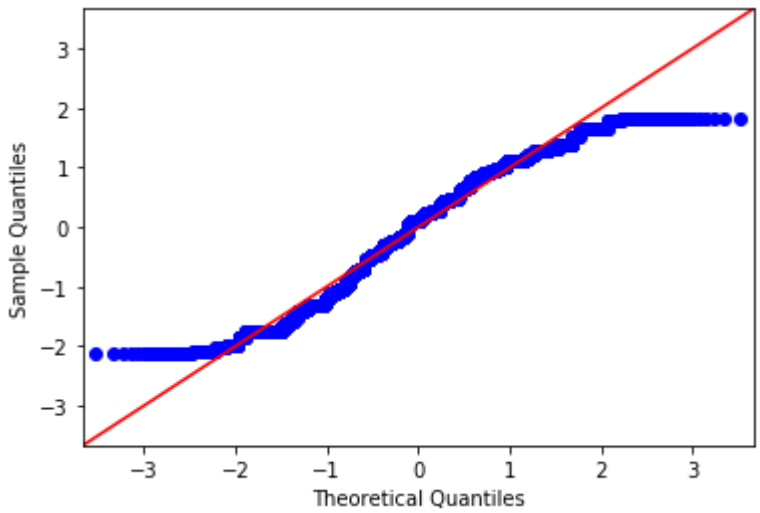
In [17]:

```
# 2.3
import statsmodels.api as sm
sm.qqplot(df.Latitude.dropna(), fit=True, line='45')
plt.show()

sm.qqplot(df.Longitude.dropna(), fit=True, line='45')
plt.show()

# I grafici confermano una distribuzione normale dei due caratteri.
# L'ipotesi è oltretutto confermata dai valori di media e mediana
```



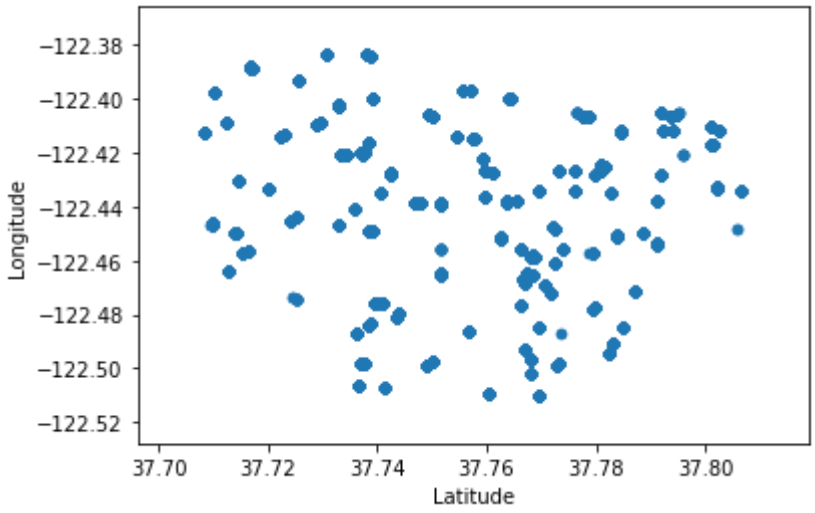


In [18]:

```
# 2.4
# Segue una legge normale perché summa di due variabili aleatorie normali.
# Con valore atteso 0 e varianza 2o^2 (non convintissimo)
```

In [19]:

```
# 2.5
df.plot.scatter('Latitude', 'Longitude')
plt.show()
print(df.Latitude.corr(df.Longitude))
```



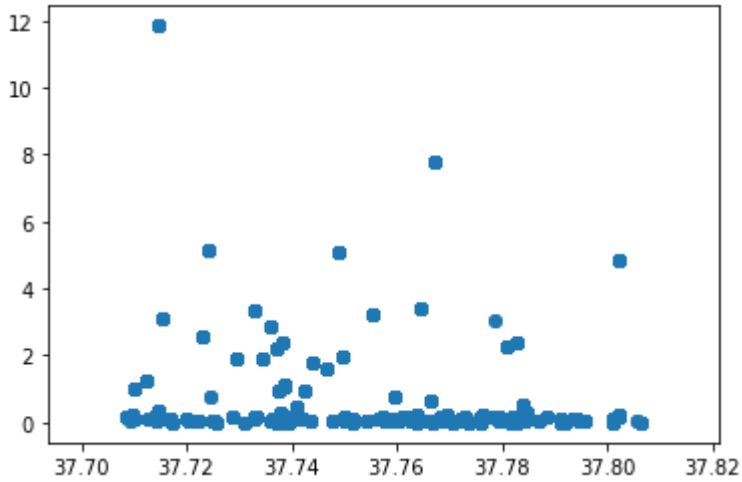

```
-0.07568466249043722
```

In [20]:

```
# 2.6
    # Sia dal grafico molto sparso che dall'indice di correlazione molto vicino a 0 è c
hiaro vedere come non ci sia alcuna
    # dipendenza tra i due caratteri.
```

In [21]:

```
# 2.7
plt.scatter(df.Latitude, df.Acres)
plt.show()
```



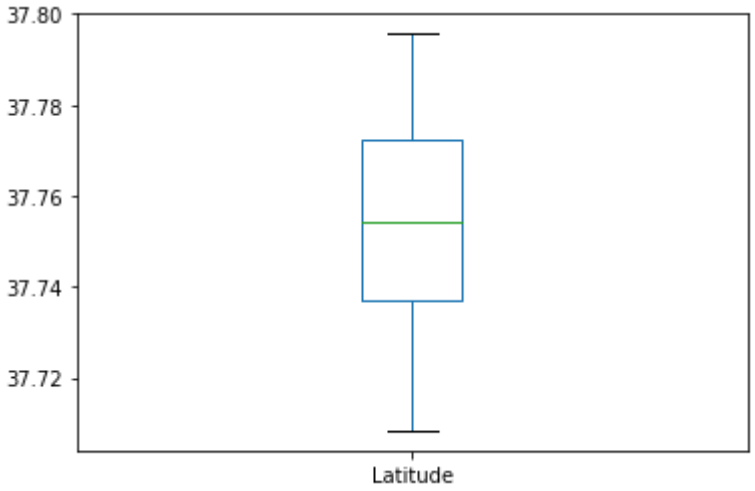
In [22]:

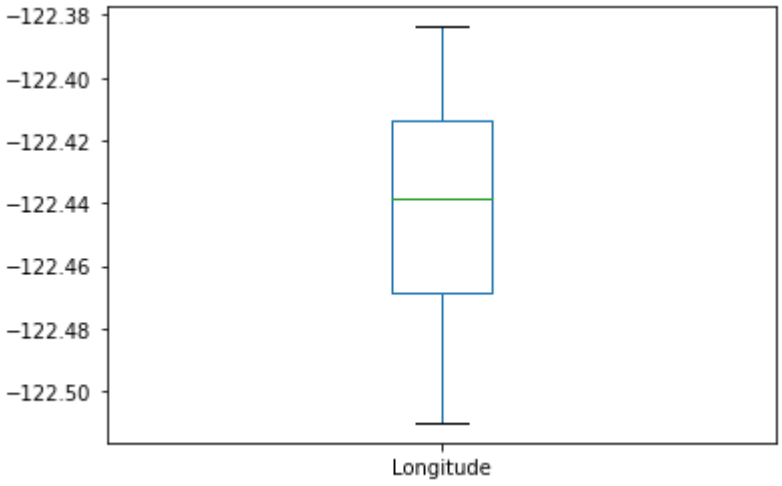
```
# 2.8
parchi = df[(df.Acres < 5) & (df.Latitude < 37.8)]
```

In [23]:

```
# 2.9
parchi['Latitude'].plot.box()
plt.show()

parchi['Longitude'].plot.box()
plt.show()
```





Esercizio 3

3.1

$$P(A - B < 0.1)$$

Visto che la Longitudine ha una distribuzione normale, standardizzo

$$P(\frac{A - B - u}{std(A - B)} < \frac{0.1 - u}{std(A - B)})$$

$$P(Z < \frac{0.1 - u}{std(A - B)})$$

In [27]:

```
import math
## per esercizio 0 la var(A - B) = 2var(X)
devstd = math.sqrt(df.Longitude.std()**2 * 2)
x = (0.1)/devstd
norm = st.norm()
norm.cdf(x)
# Potrebbe essere sbagliato
```

Out[27]:

```
0.9860392741005208
```

3.2

$$P(|A - B| < 0.1)$$

$$P(-0.1 < A - B < 0.1)$$

Visto che la Longitudine ha una distribuzione normale, standardizzo

$$P(-\frac{0.1 - u}{std(A - B)} < \frac{A - B - u}{std(A - B)} < \frac{0.1 - u}{std(A - B)})$$

$$P(Z < \frac{0.1 - u}{std(A - B)}) - P(Z < -\frac{0.1 - u}{std(A - B)})$$

In [28]:

```
devstd = math.sqrt(df.Longitude.std()**2 * 2)
x1 = (0.1)/devstd
x2 = - (0.1)/devstd
norm = st.norm()
norm.cdf(x1) - norm.cdf(x2)
```

Out[28]:

```
0.9720785482010417
```