Two way ANOVA

In [ ]:
```
# import libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [ ]:
```
df = pd.DataFrame({'water': np.repeat(['daily', 'weekly'], 15),
                   'sun': np.tile(np.repeat(['low', 'medium', 'high'], 5),2),
                   'height': [6,6,6,5,6,5,5,6,4,5,
                              6,6,7,8,7,3,4,4,4,5,
                              4,4,4,4,4,5,6,6,7,8]})
df.sample(10)
```

Out[ ]:

	water	sun	height
17	weekly	low	4
25	weekly	high	5
19	weekly	low	5
7	daily	medium	6
27	weekly	high	6
9	daily	medium	5
20	weekly	medium	4
26	weekly	high	6
3	daily	low	5
8	daily	medium	4

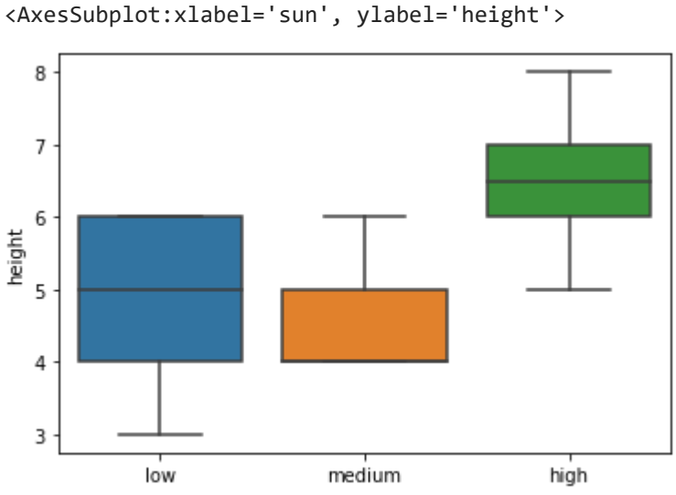
In [ ]:
```
df.dtypes
```

Out[ ]:
```
water     object
sun       object
height     int64
dtype: object
```

In [ ]:
```
sns.boxplot(df['sun'], df['height'])
```

```
c:\Users\kalee\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

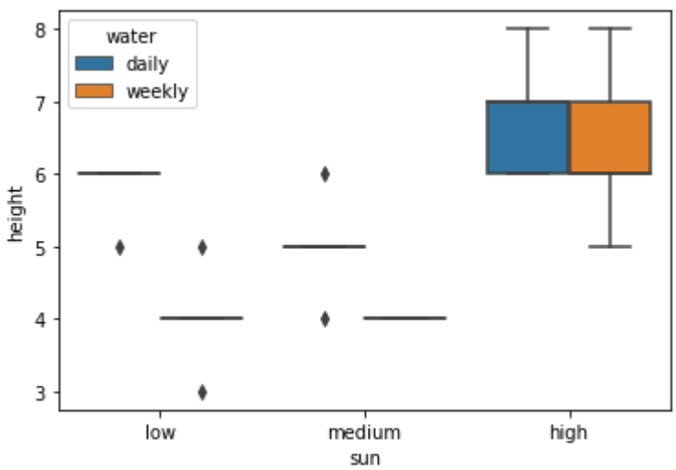
Out[ ]:
```
<AxesSubplot:xlabel='sun', ylabel='height'>
```



In [ ]:
```
sns.boxplot(df['sun'], df['height'], hue=df['water'])
```

```
c:\Users\kalee\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[ ]: <AxesSubplot:xlabel='sun', ylabel='height'>



In [ ]:
```
import statsmodels.api as sm
from statsmodels.formula.api import ols

# Oneway ANOVA
model = ols('height ~ sun', data=df).fit()
sm.stats.anova_lm(model, type=2)
```

Out[ ]:

	df	sum_sq	mean_sq	F	PR(>F)
sun	2.0	24.866667	12.433333	14.105042	0.000064
Residual	27.0	23.800000	0.881481	NaN	NaN

In [ ]:
```
# Twoway-ANOVA
model = ols('height ~ C(sun) + C(water) + C(sun) : C(water)', data=df).fit()
# C(sun) + C(water) means individual effect of sun and water on height
# C(sun) : C(water) means interactive effect of sun and water on height
sm.stats.anova_lm(model, type=2)
```

Out[ ]:

	df	sum_sq	mean_sq	F	PR(>F)
C(sun)	2.0	24.866667	12.433333	23.3125	0.000002
C(water)	1.0	8.533333	8.533333	16.0000	0.000527
C(sun):C(water)	2.0	2.466667	1.233333	2.3125	0.120667
Residual	24.0	12.800000	0.533333	NaN	NaN

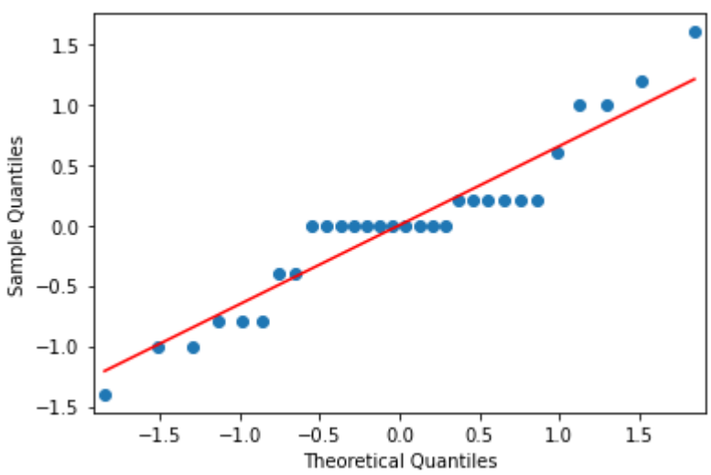
In [ ]:
```
# Another way of Twoway-ANOVA
import pingouin as pg
aov = pg.anova(data = df, dv = 'height', between = ['sun', 'water'], detailed=True)
print(aov)
```

```
        Source         SS  DF         MS        F     p-unc       np2
0          sun  24.866667   2  12.433333  23.3125  0.000002  0.660177
1        water   8.533333   1   8.533333  16.0000  0.000527  0.400000
2  sun * water   2.466667   2   1.233333   2.3125  0.120667  0.161572
3     Residual  12.800000  24   0.533333      NaN       NaN       NaN
c:\Users\kalee\anaconda3\lib\site-packages\pingouin\parametric.py:992: FutureWarning:
Not prepending group keys to the result index of transform-like apply. In the future,
the group keys will be included in the index, regardless of whether the applied funct
ion returns a like-indexed object.
To preserve the previous behavior, use

        >>> .groupby(..., group_keys=False)
```

```
To adopt the future behavior and silence this warning, use

        >>> .groupby(..., group_keys=True)
  sserror = grp.apply(lambda x: (x - x.mean()) ** 2).sum()
c:\Users\kalee\anaconda3\lib\site-packages\pingouin\parametric.py:992: FutureWarning:
Not prepending group keys to the result index of transform-like apply. In the future,
the group keys will be included in the index, regardless of whether the applied funct
ion returns a like-indexed object.
To preserve the previous behavior, use

        >>> .groupby(..., group_keys=False)

To adopt the future behavior and silence this warning, use

        >>> .groupby(..., group_keys=True)
  sserror = grp.apply(lambda x: (x - x.mean()) ** 2).sum()
c:\Users\kalee\anaconda3\lib\site-packages\pingouin\parametric.py:1071: FutureWarnin
g: Not prepending group keys to the result index of transform-like apply. In the futu
re, the group keys will be included in the index, regardless of whether the applied f
unction returns a like-indexed object.
To preserve the previous behavior, use

        >>> .groupby(..., group_keys=False)

To adopt the future behavior and silence this warning, use

        >>> .groupby(..., group_keys=True)
```

In [ ]:

```
# qq norm plot
res = model.resid    # res and resid stands for residual error
fig = sm.qqplot(res, line = 's')
plt.show()
```



Assignments

1. Twoway-ANOVA py Tukey test kesy lgana hy?
2. Lettering aur gouping kesy krni hy?
3. Bos plot m Annotations kesy lgani hen?

1. Twoway-ANOVA py Tukey test kesy lgana hy?

In [ ]:

```
# Tukey hsd with height vs water
from statsmodels.stats.multicomp import pairwise_tukeyhsd
from statsmodels.stats.multicomp import MultiComparison
mc = MultiComparison(df['height'], df['water'])
mcresult = mc.tukeyhsd()
mcresult.summary()
```

Out[ ]:

Multiple Comparison of Means - Tukey HSD, FWER=0.05

group1	group2	meandiff	p-adj	lower	upper	reject
daily	weekly	-1.0667	0.0213	-1.9622	-0.1712	True

> Upper results indicates that significant diff in daily and weekly

> watering as rejection is true

In [ ]:
```
# Tukey hsd with height vs sun
from statsmodels.stats.multicomp import pairwise_tukeyhsd
from statsmodels.stats.multicomp import MultiComparison
mc = MultiComparison(df['height'], df['sun'])
mcresult = mc.tukeyhsd()
mcresult.summary()
```

Out[ ]:

Multiple Comparison of Means - Tukey HSD, FWER=0.05

group1	group2	meandiff	p-adj	lower	upper	reject
high	low	-1.7	0.0011	-2.741	-0.659	True
high	medium	-2.1	0.0001	-3.141	-1.059	True
low	medium	-0.4	0.6124	-1.441	0.641	False

> Upper results indicates that there are significant diff in high vs low
>
> and high vs medium as rejection is true while no diff b/w low vs medium

In [ ]:
```

```