In [ ]:

```
import pandas as pd
import numpy as np
import scipy.stats as sp
import matplotlib.pyplot as plt

rs = np.random.RandomState(seed=10)
```

Problem 1

Consider a Random Walk given by:

$$X_t = \delta + X_{t-1} + Z_t,\ Z_t \overset{iid}{\sim} WN(0,1)$$
$$X_0 = 0$$

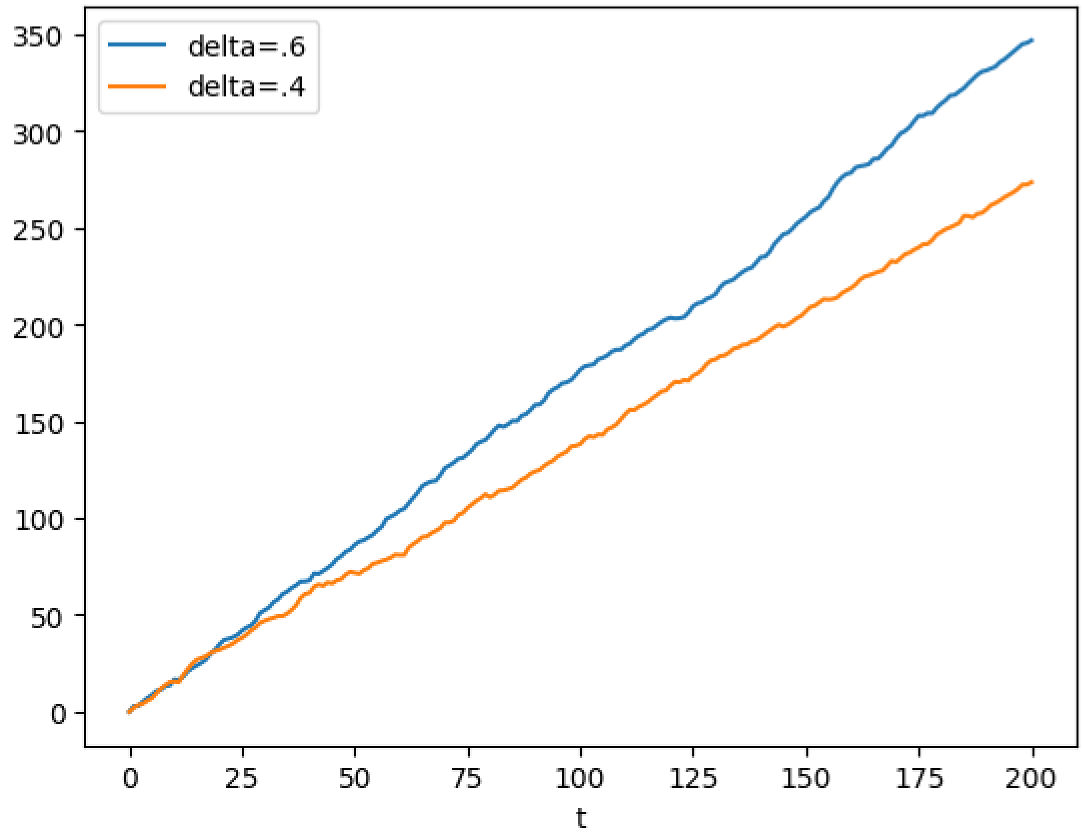
a. Show that this process can be rewritten as the cumulative sum of white noise terms, $X_t = \delta t + \sum_{j=1}^{t} Z_j$

See appendix 1.a

b. Simulate 200 observations **each** of two random walks; the first with $\delta = 0.6$ and the second with $\delta = 0.4$. Plot both realizations on the same plot using different colors.

In [ ]:

```
delta = [.6, .4]
X = pd.DataFrame(columns = ['delta=.6', 'delta=.4'],index = range(201))
X.loc[0] = [0,0]

for t in range(1,201):
    for j in range(2):
        X.loc[t][j] = delta[j] + X.loc[t-1][j] + sp.norm.rvs(loc=1,scale=1,random_state=

plt.plot(range(201), X['delta=.6'], label='delta=.6')
plt.plot(range(201), X['delta=.4'], label='delta=.4')
plt.legend()
plt.xlabel('t')
plt.show()
```



c. Describe the plot. Does this Random Walk appear stationary?

Clearly it is not stationary since it does not have a constant mean over time, which is due to the delta term.

d. Prove that a Random Walk is not Weakly Stationary, even if $\delta = 0$.

See appendix 1.d

Question 2

In [ ]:
```
from statsmodels.tsa.arima_process import ArmaProcess
from statsmodels.tsa.stattools import acf, pacf
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
```

Let's simulate and compare some AR and MA processes. Lets consider both AR(2) and MA(2) processes given by:

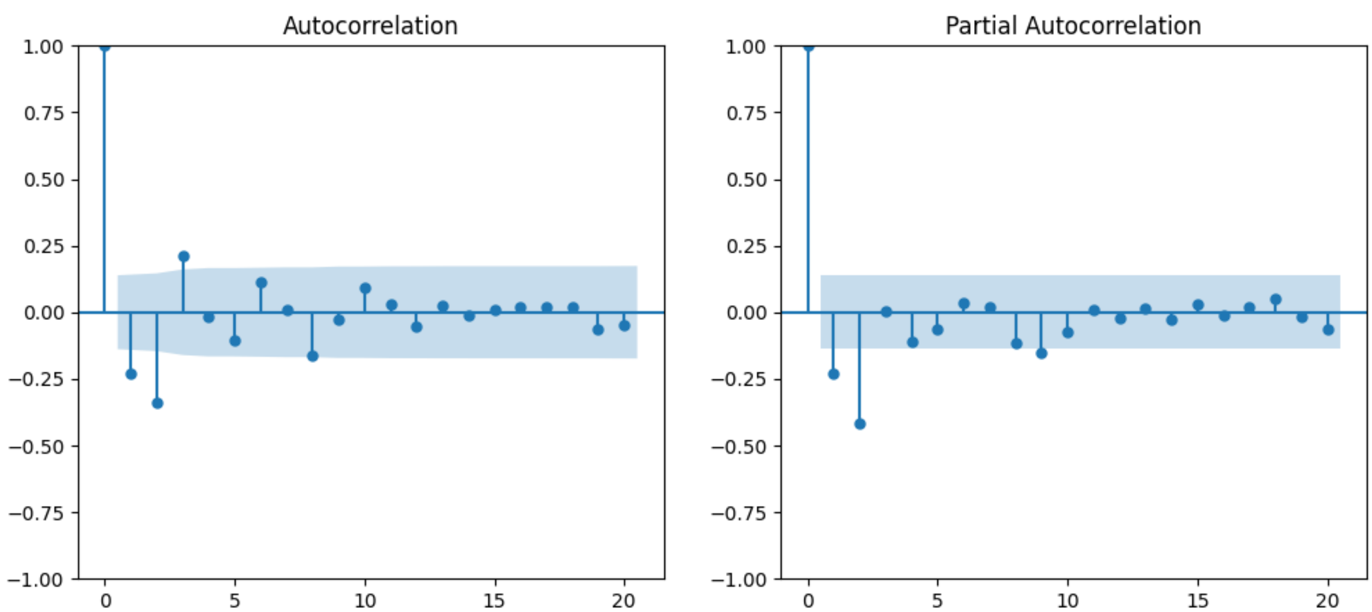
$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \varepsilon_t \quad \varepsilon_t \overset{iid}{\sim} \mathcal{N}(0,1)$$
$$Y_t = Z_t + \theta_1 Z_{t-1} + \theta_2 Z_{t-2} \quad Z_t \overset{iid}{\sim} \mathcal{N}(0,1)$$

a. Recall that for an AR(1) process, we merely required that $|\phi| < 1$ for causality. What is the similar condition for an AR(2) process?

See appendix 2.a

b. Let $\phi_1 = 0.35$ and $\phi_2 = 0.45$. Simulate the AR(2) process for 200 samples and take the sample ACF and PACF. Use 20 lags.

In [ ]:
```
np.random.seed(10)

ar_params = [1, .35, .45] #Form of AR(2)
ma_prams = [1] #Represents MA(0)

ar_2 = ArmaProcess(ar=ar_params, ma=ma_prams)

simulated_ar_2 = ar_2.generate_sample(nsample=200)

sample_acf = acf(simulated_ar_2, nlags=20)
sample_PACF = pacf(simulated_ar_2, nlags=20)


fig, (ax1,ax2) = plt.subplots(1,2, figsize=(12,5))
plot_acf(simulated_ar_2, ax=ax1, lags=20);
plot_pacf(simulated_ar_2, ax=ax2, lags=20);
```

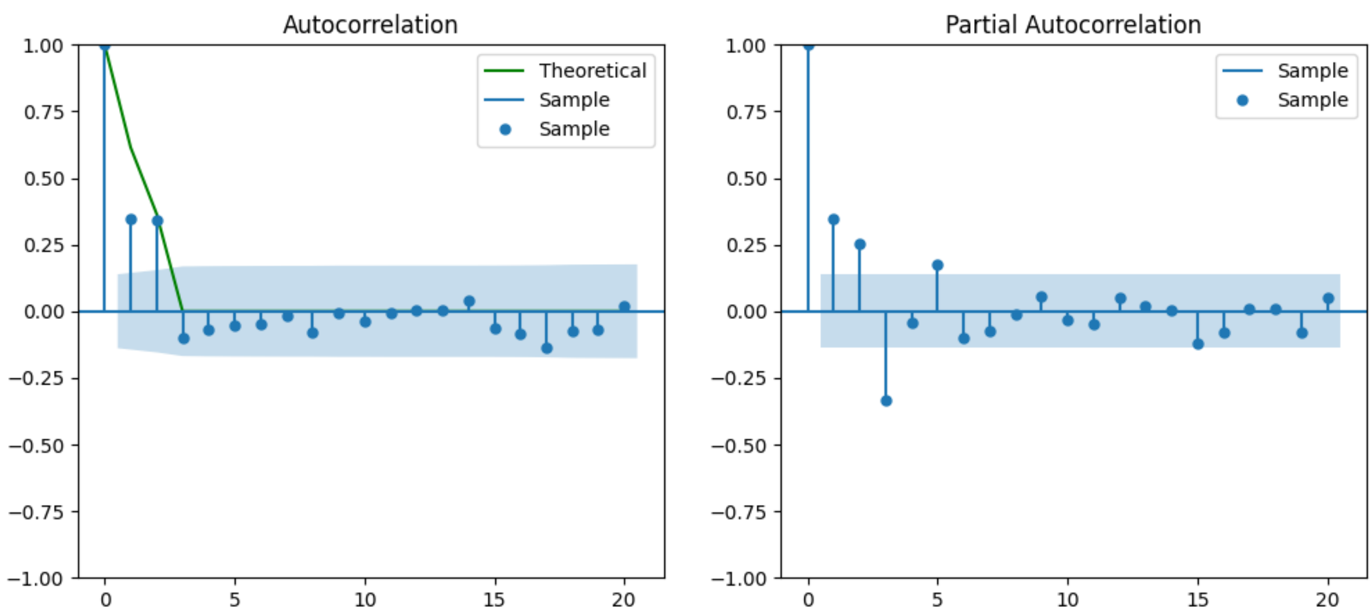


c. For the Moving Average process: let $\theta_1 = 0.45$ and $\theta_2 = 0.55$. For the moving average model only, the theoretical ACF is given by:

$$\rho_Y(h) = \begin{cases} 1 & \text{if } h = 0 \\ \frac{\theta_1 + \theta_1\theta_2}{1+\theta_1^2+\theta_2^2} & \text{if } |h| = 1 \\ \frac{\theta_2}{1+\theta_1^2+\theta_2^2} & \text{if } |h| = 2 \\ 0 & \text{otherwise} \end{cases}$$

In [ ]:
```
ar_params = [1] #Form of AR(0)
ma_params = [1, .45, .55] #Form of MA(2)

def theoretical_ma_acf_function(h, parameters):
    if h == 0:
        return 1
    elif abs(h) == 1:
        return ma_params[1] + ma_params[1] * ma_params[2] / (1 + ma_params[1]**2 + ma_pa
    elif abs(h) == 2:
```

```
        return ma_params[2] / (1 + ma_params[1]**2 + ma_params[2]**2)
    else:
        return 0

theoretical_ma_acf = []
for h in range(21):
    theoretical_ma_acf.append(theoretical_ma_acf_function(h, ma_params))

ma = ArmaProcess(ar_params, ma_params)
simulated_ma_2 = ma.generate_sample(200)

fig, (ax1, ax2) = plt.subplots(1,2,figsize=(12,5))
ax1.plot(range(21), theoretical_ma_acf, color='green', label='Theoretical')
plot_acf(simulated_ma_2, lags=20, ax=ax1, label='Sample');
plot_pacf(simulated_ma_2, lags=20, ax=ax2, label='Sample');
ax1.legend()
ax2.legend()

plt.show()
```



d. Compare the Sample ACF of the AR(2) and MA(2) processes. Do the same for the PACF of both processes

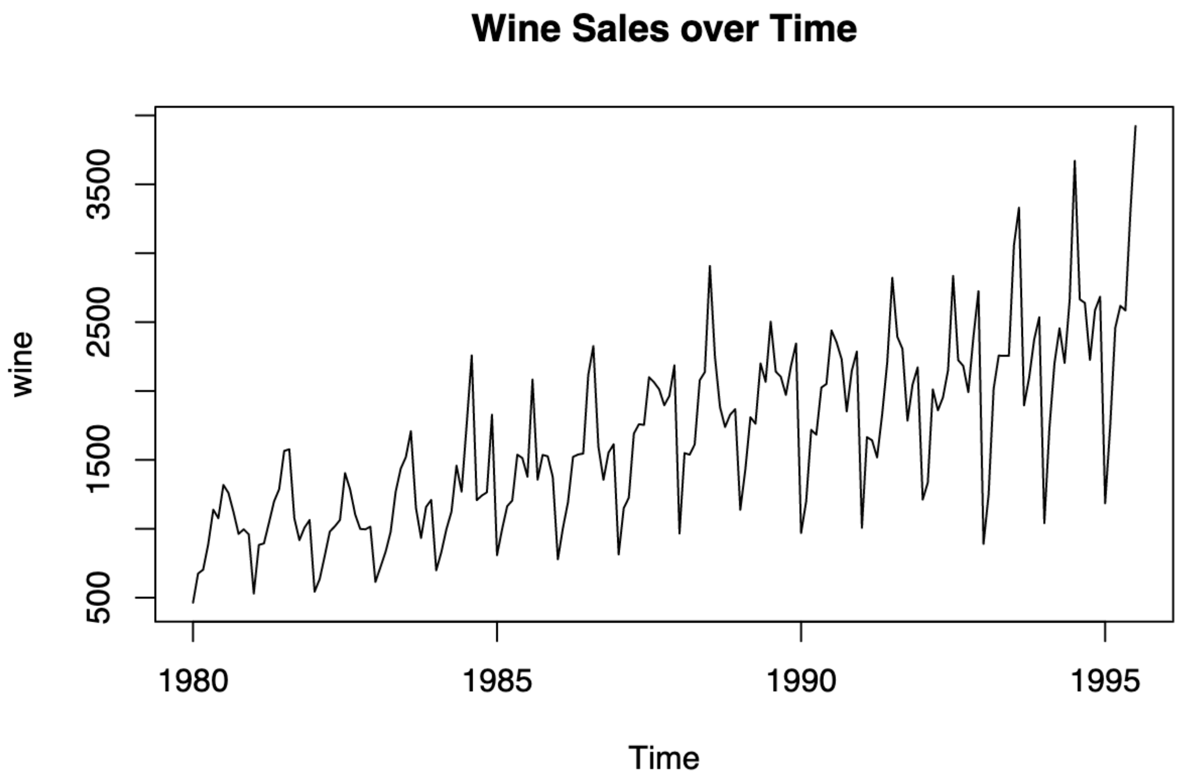
The sample ACF for the AR(2) process is signifcant for h<=3 and exhibits positive and negative values, while it is only significant for h<=2 for the MA(2) and is only positive (for the significant values). The sample ACF for the AR(2) process slowly goes towards zero while the drop for MA(2) is very abrupt. This is consistent with how the models are defined.

Conversely, the sample PACF is only significant for h<=2 and only negative for the significant values, while the sample PACF is significant for h=1,2 and 4, and shows a slower decay towards zero, while exhibiting both positive and negative significant values.

Question 3

In [ ]:
```
data = pd.read_csv('monthly-australian-wine-sales-th.csv')
```

```
data.drop(index=data.index[-1], inplace=True)
```



From this, we can see:

- This data appears to be increasing
- The data variability changes over time
- There appears to be seasonality

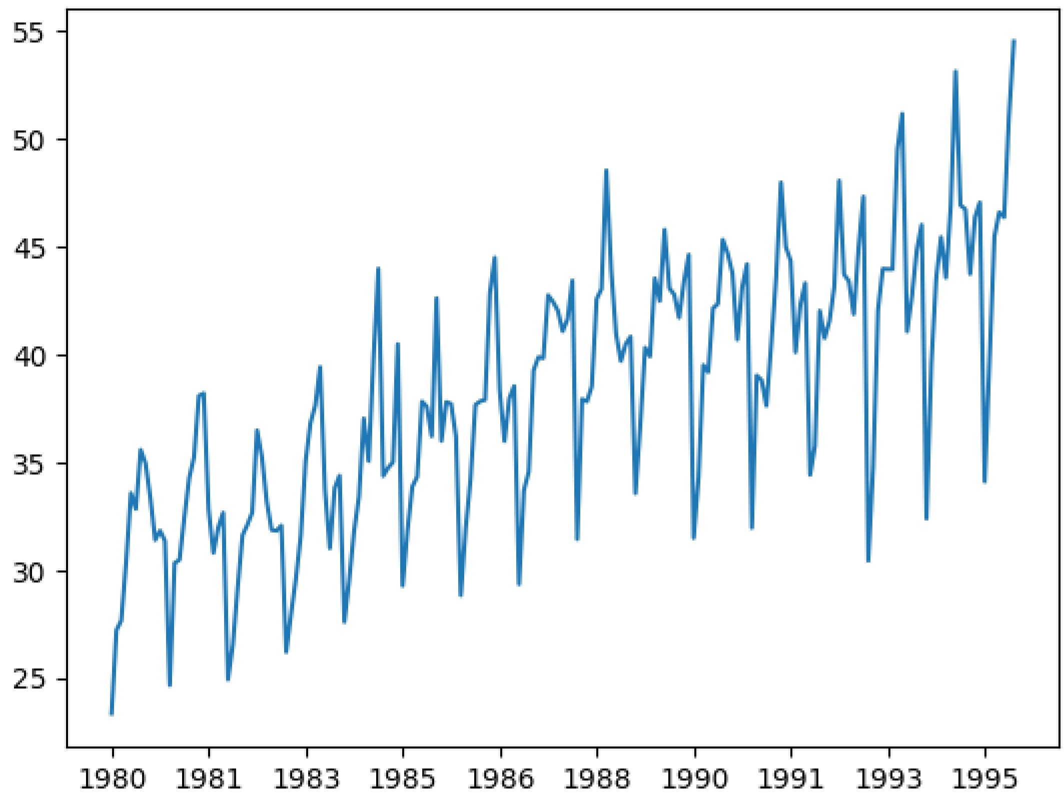
Let's explore how to deal with this.

a. One thing we could try is to transform our data, as we often do in typical linear regressions. Doing so may or may not yield a way for us to make our data stationary. Try a few transformations of the response variable and see if that removes any of the trends. You can even try Box-Cox if you want. This won't be our focus, so don't look at too many transformations.

The boxcox transformaiton is very versatile but does not seem to fix the seasonality.

In [ ]:

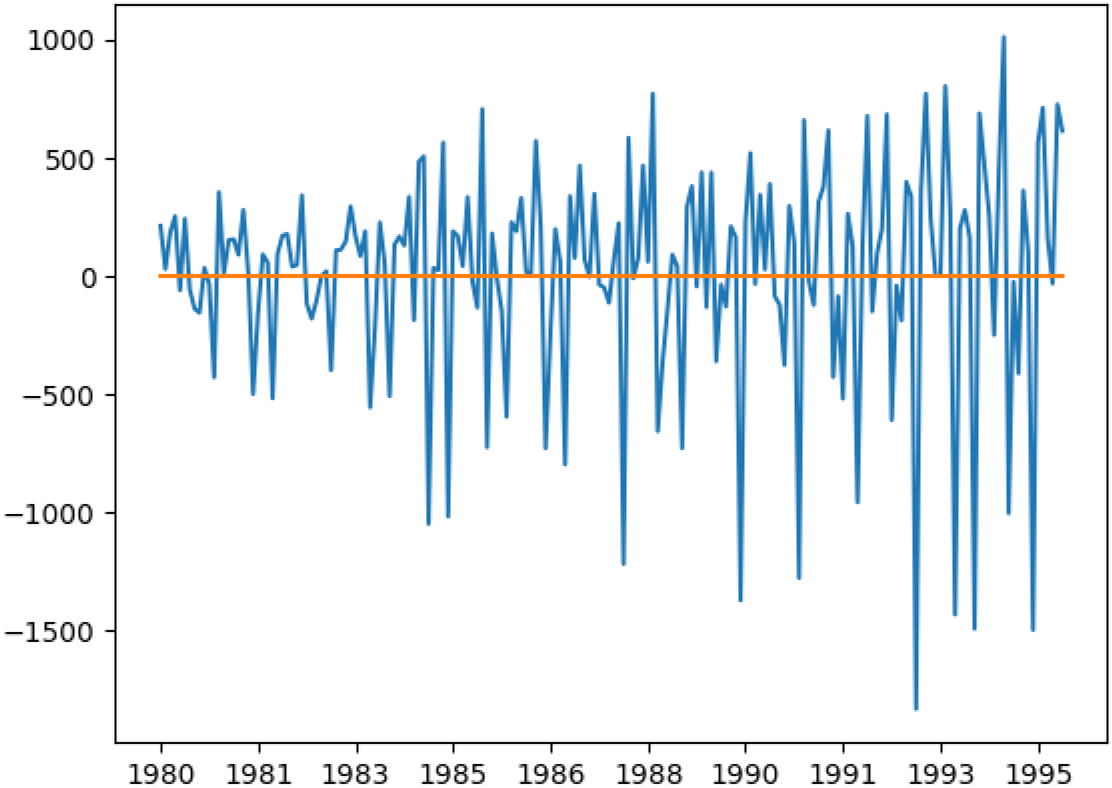
```
data_tf = sp.boxcox(data['Values'][:-1])

plt.plot(data['Month'][:-1],data_tf[0])
plt.xticks(data['Month'][::20], [str(x)[:-3] for x in data['Month'][::20]])
plt.show()
```



b. Let's try a simple differencing, and see what that gives us. If X_t represents our wine sales, differencing a series creates a new time series $Y_t = \nabla X_t = X_t - X_{t-1}$. Create a difference series, plot that, and plot a line through 0 so we can see if we're close to de-trending.

This does not seem to fix the issue either.

In [ ]:
```
Y_t = []

for t in range(1,len(data)):
    Y_t.append(data['Values'][t] - data['Values'][t-1])

plt.plot(data['Month'][1:], Y_t)
plt.plot(range(len(data)-1), [0]*(len(data)-1))
plt.xticks(data['Month'][1::20], [str(x)[:-3] for x in data['Month'][1::20]])
plt.show()
```

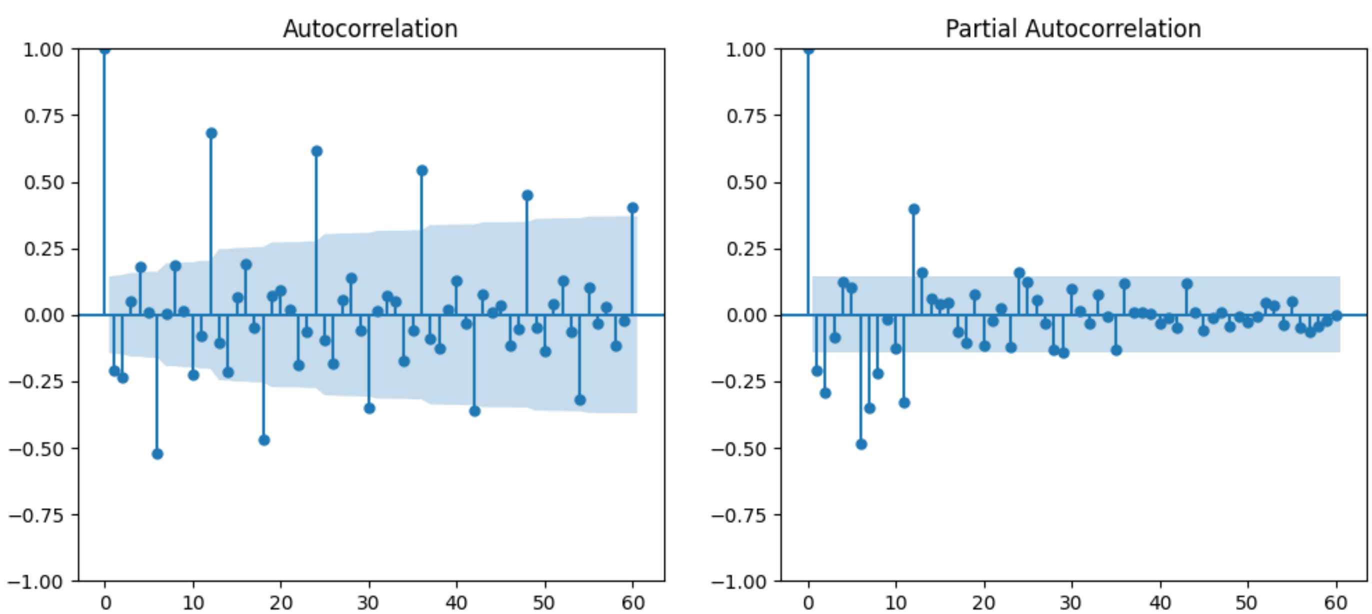


c. You should notice that we have removed our trend, but we still have some cyclic behavior. Plot the sample ACF and PACF of the differenced series, with 60 lags. What do they tell us? Why is there still a repeating pattern appearing?

There seems to be a cyclical pattern with length of 12 months for the ACF. However, the PACF decays relatively fast towards zero. This is indicative of an AR(12) model. There is still a repeasting pattern because of the 12-month seasonal cycle inherent to the data.

In [ ]:

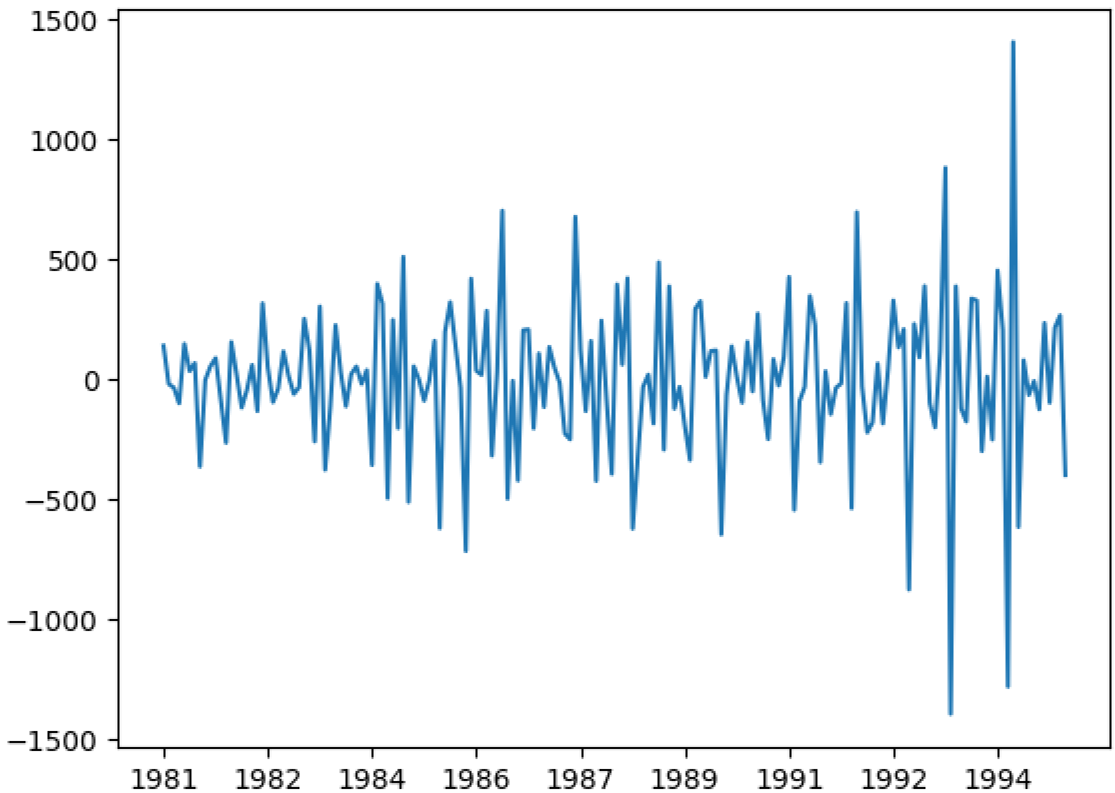
```
fig, (ax1, ax2) = plt.subplots(1,2,figsize=(12,5))
plot_acf(Y_t, lags=60, ax=ax1);
plot_pacf(Y_t, lags=60, ax=ax2);
```

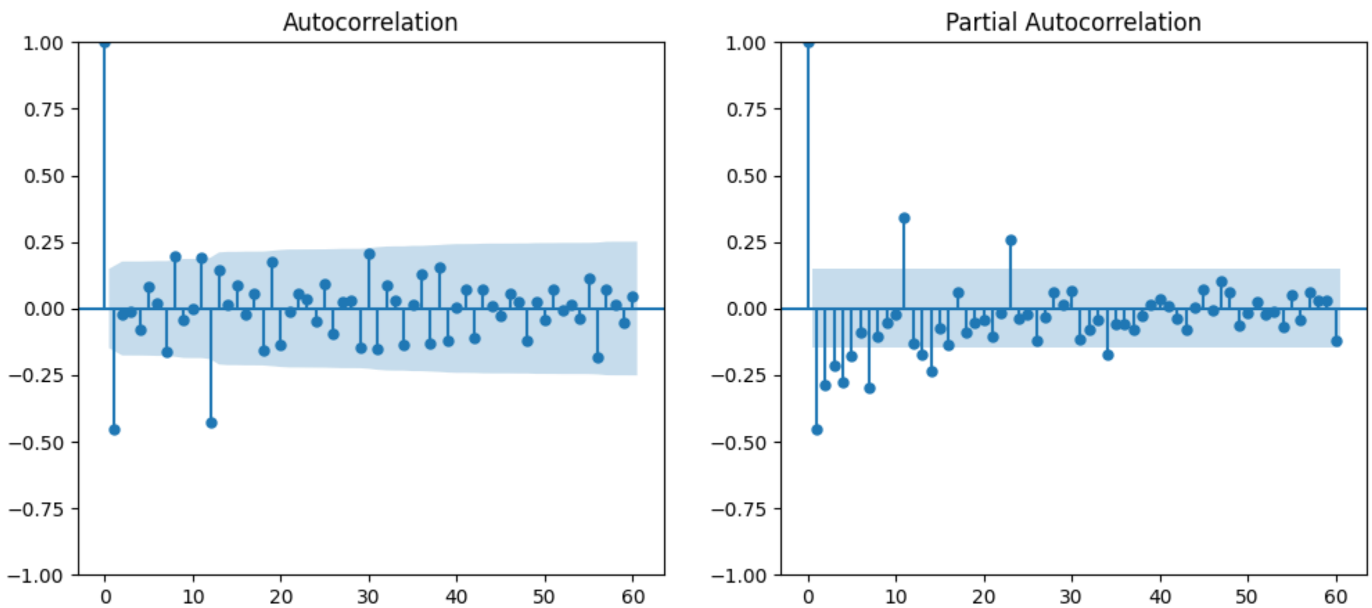


d. You have reason to believe that there is a 12-month cycle to sales. Difference the series (the one we already differenced) again, but for 12 steps this time. Plot the time-series, the ACF and the PACF.

We can see that the AR(12)-like behavior show in the ACF is now gone.

In [ ]:
```
W_t = []

for i in range(12,len(Y_t)):
    W_t.append(
        Y_t[i] - Y_t[i-12]
    )

plt.plot(data['Month'][13:], W_t)
plt.xticks(data['Month'][13::20], [str(x)[:-3] for x in data['Month'][13::20]])
plt.show()

fig, (ax1, ax2) = plt.subplots(1,2, figsize=(12,5))
plot_acf(W_t, lags=60, ax=ax1);
plot_pacf(W_t, lags=60, ax=ax2);
```




Autocorrelation
Partial Autocorrelation
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
0
10
20
30
40
50
60

Appendix

1.a

a. Let $X_1 = \delta + Z_1 + X_0 = \delta + Z_1$

$X_2 = \delta + Z_2 + X_1 = 2\delta + \sum_{i=1}^{2} Z_i$

Suppose $X_t = \delta t + \sum_{i=1}^{t} Z_i$

then $X_{t+1} = \delta + X_t = (t+1)\delta + \sum_{i=1}^{t+1} Z_i$

d. A process is not stationary if $\gamma_X(t, t+h) = F(t)$

Let $h = 0$

$\gamma_X[X_t, X_{t+h}] = \text{cov}[X_t, X_t] = \text{cov}[\delta + X_{t-1} + Z_t, \delta + X_{t-1} + Z_t]$

$= \text{cov}[X_{t-1} + Z_t, X_{t-1} + Z_t] = \text{cov}[X_{t-1}, X_{t-1}] + \text{cov}[Z_t, Z_t]$

$\text{cov}[X_{t-1}, X_{t-1}] = \text{cov}[X_{t-2}, X_{t-2}] + \text{cov}[Z_{t-1}, Z_{t-1}]$

1.d

d. A process is not stationary if $\gamma_X(t, t+h) = f(t)$

Let $h=0$

$\gamma_X[X_t, X_{t+h}] = cov[X_t, X_t] = cov[\delta + X_{t-1} + Z_t, \delta + X_{t-1} + Z_t]$

$= cov[X_{t-1} + Z_t, X_{t-1} + Z_t] = cov[X_{t-1}, X_{t-1}] + cov[Z_t, Z_t]$

$cov[X_{t-1}, X_{t-1}] = cov[X_{t-2}, X_{t-2}] + cov[Z_{t-1}, Z_{t-1}]$

$\vdots$

$cov[X_{t-(t-1)}, X_{t-(t-1)}] = \underbrace{cov[X_0, X_0]}_{0} + cov[Z_{t-(t-1)}, Z_{t-(t-1)}]$

we get one σ_Z^2 term for each (of t) iterations:

$\gamma_X(t, t+h) = t \cdot \sigma_Z^2 = t = f(t)$

hence not stationary, independent of δ

2.a

29. It is required that $\phi(z) = 1 - \phi_1 z - \phi_2 z^2 = 0$, $|z| > 1$

This is equivilent to $z^2 - \phi_1 z - \phi_2 = 0$, $|\lambda| < 1$

By the quadratic eqn $\left| \dfrac{\phi_1 \pm \sqrt{\phi_1^2 + 4\phi_2}}{2} \right| < 1$

If $\phi_1^2 \geq 4\phi_2$ (Real Roots)

$$-1 < \frac{\phi_1 - \sqrt{\phi_1^2 + 4\phi_2}}{2} < \frac{\phi_1 + \sqrt{\phi_1^2 + 4\phi_2}}{2} < 1$$

$$\left[-2 < \phi_1 - \sqrt{\phi_1^2 + 4\phi_2}\right] < \left[\phi_1 + \sqrt{\phi_1^2 + 4\phi_2} < 2\right]$$

$$\Downarrow$$

$$\sqrt{\phi_1^2 + 4\phi_2} < \phi_1 + 2 \implies \phi_1^2 + 4\phi_2 < \phi_1^2 + 4\phi_1 + 4 \implies \phi_2 - \phi_1 < 1$$

$$\sqrt{\phi_1^2 + 4\phi_2} < 2 - \phi_1 \implies \phi_1^2 + 4\phi_2 < 4 - 4\phi_1 + \phi_1^2$$

$$\phi_2 + \phi_1 < 1$$

If $\phi_1^2 + 4\phi_2 < 0$ (complex case)

The roots are complex conjugates iff $|z_1| = |z_2|$, $|z_1|^2 < 1$

and it can be shown this implie $|\phi_2| < 1$

Thus we have the conditions:

- $\phi_1 + \phi_2 < 1$
- $\phi_2 - \phi_1 < 1$
- $|\phi_2| < 1$