정상성 유무의 판단은 분석마다 다르지만, 가져온 출처에서는 (b), (c), (d), (h)라고 합니다.

++'(b)가 어떻게 정상성을 갖느냐?' 라 생각할 수도 있지만 약간의 튄 값은 융통성으로 무마해준다고 합니다.

정상성을 만족하기 위해선 시계열의 확률적인 성질들이 "시간의 흐름에 따라 불변"해야 하는 조건을 만족해야 하는데요.

1. 뚜렷한 추세가 없음. 즉, 시계열의 평균이 시간 축에 평행
2. 시계열의 진폭(변동)이 시간의 흐름에 따라 일정

정상성(stationarity)을 나타내는 시계열은 시계열의 특징이 해당 시계열이 관측된 시간에 무관합니다.[15] 따라서, 추세나 계절성이 있는 시계열은 정상성을 나타내는 시계열이 아닙니다 — 추세와 계절성은 서로 다른 시간에 시계열의 값에 영향을 줄 것이기 때문입니다. 반면에, 백색잡음(white noise) 시계열은 정상성을 나타내는 시계열입니다 — 언제 관찰하는지에 상관이 없고, 시간에 따라 어떤 시점에서 보더라도 똑같이 보일 것이기 때문입니다.

몇 가지 경우는 헷갈릴 수 있습니다 — 주기성 행동을 가지고 있는 (하지만 추세나 계절성은 없는) 시계열은 정상성을 나타내는 시계열입니다. 왜냐하면 주기가 고정된 길이를 갖고 있지 않기 때문에, 시계열을 관측하기 전에 주기의 고점이나 저점이 어디일지 확실하게 알 수 없습니다.

일반적으로는, 정상성을 나타내는 시계열은 장기적으로 볼 때 "예측할 수 있는 패턴을 나타내지 않을 것"입니다. (어떤 주기적인 행동이 있을 수 있더라도) 시간 그래프는 시계열이 일정한 분산을 갖고 대략적으로 평평하게 될 것을 나타낼 것입니다.

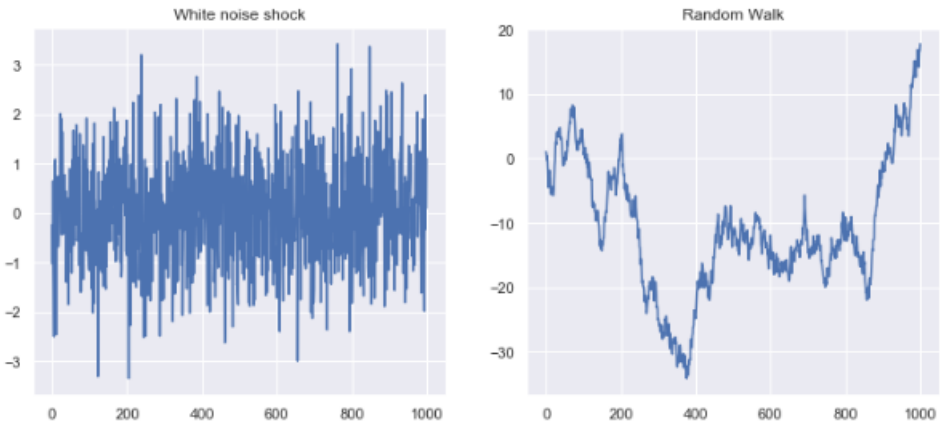
> Stationary(정상성) vs. Non-stationary(비정상성)

시계열 데이터가 가진 중요한 특징이다. 시계열 분석을 진행하기 위해서는 시계열 데이터의 특성을 파악해야 한다.

Stationary(정상성): 시간이 변해도 통계적 특성이 일정한 시계열이다. 통계적 특성이 일정한 정도에 따라 Strongly Stationary(강정상)과 Weakly Stationary(약정상)으로 구분된다. 일반적으로 약정상 시계열 정도만 되어도 정상성을 띈다고 본다.

1. "평균", "분산", "왜도", "첨도" 등 모든 통계적 특성이 동일하면 강정상;
2. 평균과 분산의 통계적 특성이 동일하면 약정상으로 구분한다.

Non-stationary(비정상성): 시간에 따라 통계적 특성이 변한다. 시계열 데이터가 non-stationary 하다면 평균, 분산, 공분산은 시간의 함수가 될 수 없다.



<좌: Stationary / 우: Non-stationary>

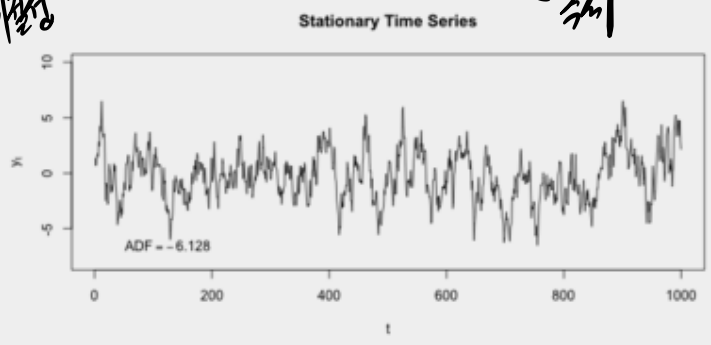
데이터가 정상성을 띄는지 비정상성을 띄는지에 따라 활용하는 모델이 다르기 때문에, 이를 파악하는 것은 필수적이다.

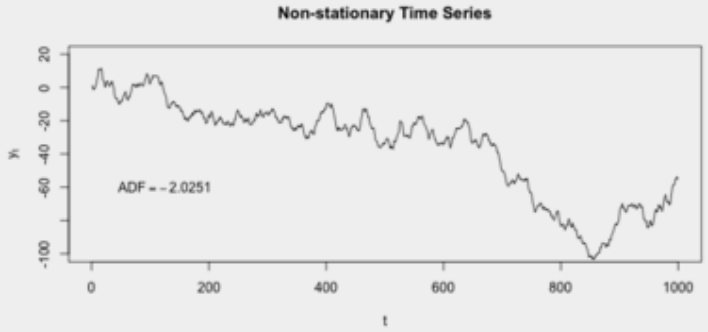
우선 정상성은 무엇일까요?

평균과 분산 같은 통계적 특성이 시간에 대해 일정한 성질을 정상성이라고 합니다. 따라서 전에 포스팅 했던 랜덤 과정을 기억하시나요? 정상성이 있는 랜덤 과정을 정상과정이라고 합니다.

time series의 stationarity를 체크해봅시다.

- 말로 하기는 좀 귀찮으니까, 그림을 봅시다 허허. 아래 그림을 보시면, stationary time series와 non-stationary time series가 나타나 있습니다. 간단히 말하면, stationary의 경우는 시간이 변해도, 일정한 분포를 따르는 경우를 말하고, non-stationary 의 경우는 시간이 변해도, 일정한 분포를 따르지 않는 경우를 말합니다.
- time series를 분석할 때, stationarity, 간단히 말하면 통계적 일관성이라고 말할 수도 있겠네요. 이게 지켜지면 얼마나 좋겠냐만, 보통은 지켜지지 않습니다. "seasonality"가 들어가거나, "trend"(값이 계속 증가하는 추세이거나) 등등의 변화로 인해 이 성질은 지켜지는 것이 어렵죠.





Checks for Stationarity

There are many methods to check whether a time series (direct observations, residuals, otherwise) is stationary or non-stationary.

1. **Look at Plots**: You can review a time series plot of your data and visually check if there are any obvious trends or seasonality.
2. **Summary Statistics**: You can review the summary statistics for your data for seasons or random partitions and check for obvious or significant differences.
3. **Statistical Tests**: You can use statistical tests to check if the expectations of stationarity are met or have been violated.

'시간' 설정을 여러번 해본 뒤, 각 시간의 통계적 특성을 비교해본다.

check for stationarity

- 제가 참고한 포스트에서는 우리가 가진 time series가 stationarity 성질을 갖추고 있는지 확인하기 위해서는 다음 세 가지 정도의 방법이 있다고 합니다.
  1. 눈으로 보기: 직접 plotting해서 시간에 따라 변하는지 볼 것
  2. 간단한 평균 내보기: 대략 반으로 쪼개서 앞의 평균과 뒤의 평균이 얼마나 다른지 볼 것
  3. statistical test: 통계적 검정하기

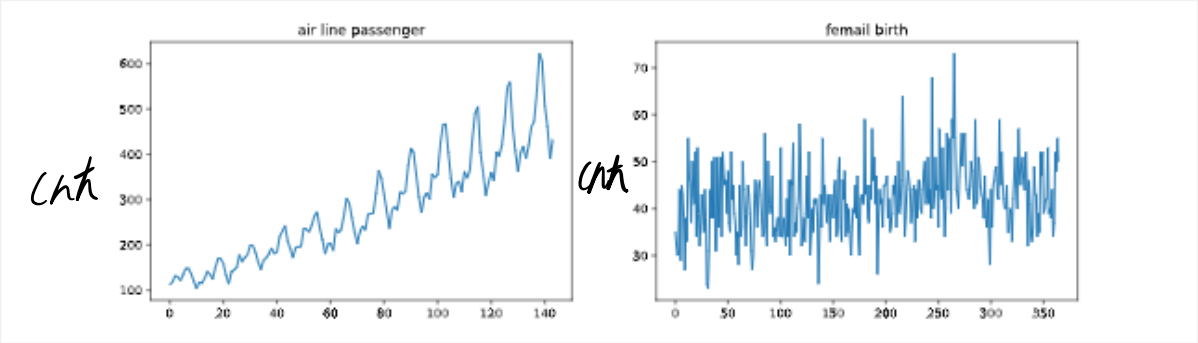
do it.

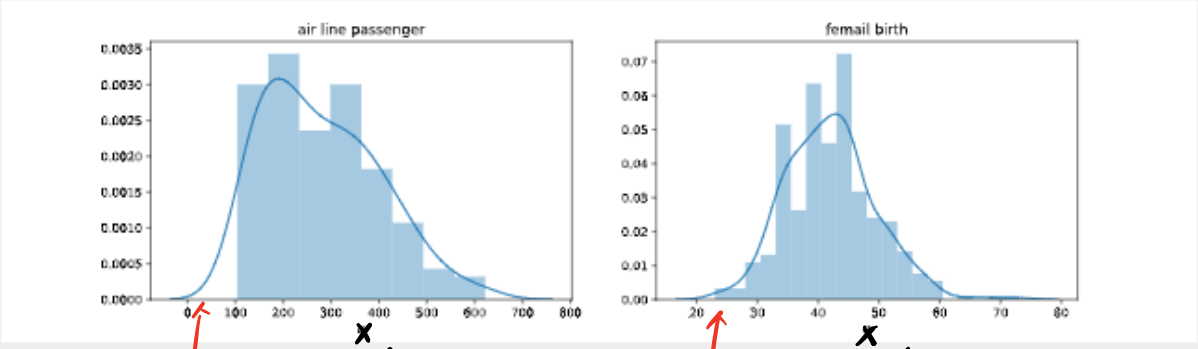
- 직접 해보도록 합니다. 여기서는 두 데이터 set을 사용합니다.

just plotting

- 일단 그냥 그림으로도 airline passenger는 확실한 trend가 보입니다. 또 seasonal effect도 분명하게 보입니다. 그래서 non-stationary하죠.
- 다만 female birth는 trend는 없는데, seasonal effect가 있는건지 아직은 잘 모르겠어요.
- 또한, histogram을 봐도, airline passenge는 gaussian dist를 따르지 않는 것처럼 보이죠. female birth는 약간 따르는 것처럼 보입니다. gaussian을 따르는지 여부 또한, 중요한 stationarity check 기법 중 하나입니다.
- 또한, data를 두 그룹으로 나누어(시간상 앞쪽인 그룹과, 뒤쪽인 그룹) 평균과 표준편차를 비교해보아도 비슷한 결과가 나오죠.

```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np

df_airline = pd.read_csv('/Users/frhyme/Downloads/international-airline-passengers.csv
df_airline.columns = ['m', 'q']
df_female = pd.read_csv('/Users/frhyme/Downloads/daily-total-female-births-in-cal.csv'
df_female.columns = ['m', 'q']

## 1) just plotting
f, axes = plt.subplots(1, 2, figsize=(14, 4))
axes[0].plot(df_airline['q'])
axes[0].set_title("air line passenger")
axes[1].plot(df_female['q'])
axes[1].set_title("femail birth")
plt.savefig("../../assets/images/markdown_img/180612_1811_just_plotting.svg")
plt.show()

## 2) plot histogram to check to follow gaussian dist.
f, axes = plt.subplots(1, 2, figsize=(14, 4))
sns.distplot(df_airline['q'], ax=axes[0])
axes[0].set_title("air line passenger")
sns.distplot(df_female['q'], ax=axes[1])
axes[1].set_title("femail birth")
plt.savefig("../../assets/images/markdown_img/180612_1815_hist_comp.svg")
plt.show()

## 3) mean variance comparision
print("---airline passenger---")
print("mean of left group, right group: {}, {}".format(
    df_airline['q'][:len(df_airline)//2].mean(), df_airline['q'][len(df_airline)//2:].
)
print("std of left group, right group: {}, {}".format(
    df_airline['q'][:len(df_airline)//2].std(), df_airline['q'][len(df_airline)//2:].s
)

print("---female birth---")
print("mean of left group, right group: {}, {}".format(
    df_female['q'][:len(df_female)//2].mean(), df_female['q'][len(df_female)//2:].mean
)
print("std of left group, right group: {}, {}".format(
    df_female['q'][:len(df_female)//2].std(), df_female['q'][len(df_female)//2:].std()
)
```





"정규분포"가 아닌 것 같음

"정규분포"인 것 같음.

첫 사람부터 중간 사람까지의 시간

중간 사람부터 끝 사람까지의 시간.

```
---airline passenger---
mean of left group, right group: 182.90277777777777, 377.69444444444446
std of left group, right group: 47.7042413215282, 86.4392058427729
---female birth---
mean of left group, right group: 39.76373626373626, 44.185792349726775
std of left group, right group: 7.034579412457393, 6.998305548491794
```