الفصل الخامس

الارتباط التلقائي والارتباط التلقائي الجزئي

Autocorrelation and Partial Autocorrelation

محتويات الفصل الخامس

- مقدمة Introduction
- الارتباط الاحصائي Statistical Correlation
- مفهوم الارتباط التلقائي Autocorrelation
- إيجاد الارتباط التلقائي
- الارتباط التلقائي لبيانات سلسلة زمنية حقيقية
- مفهوم الارتباط التلقائي الجزئي Partial Autocorrelation
- الارتباط التلقائي الجزئي لبيانات سلسلة زمنية حقيقية
- تفسير مخططات الارتباط التلقائي والارتباط التلقائي الجزئي

لتحميل ملحقات هذا الفصل من الأكواد البرمجية والشروحات التوضيحية زيارة المستودع التالي على صفحتي في موقع GitHub :

- السلاسل الزمنية باللغة العربية

مقدمة Introduction

في هذا الفصل سنتعرف على مفهوم الارتباط التلقائي Autocorrelation والذي يستخدم بشكل أساسي للمساعدة في تحليل السلاسل الزمنية Times Series Analysis واختيار مُعاملات النماذج models' parameters الخاصة بالتنبؤ forecasting، حيث سنستخدم دالتين أساسيتين وهما :

1- دالة الارتباط التلقائي Autocorrelation Function (ACF)

2- دالة الارتباط التلقائي الجزئي Partial Autocorrelation Function (PACF)

المسؤولتان عن إنشاء رسومات Plots تلخص بيانياً قوة العلاقة correlation بين قيمة (نقطة مشاهدة) observation في سلسلة زمنية مع قيم أخرى other observations في خطوات زمنية سابقة. يمكن أن يكون الفرق بين الارتباط التلقائي والارتباط التلقائي الجزئي صعبًا ومربكًا قليلًا في التنبؤ بالسلسلة الزمنية ولكن كلاهما يعتمد على مفهوم الارتباط Correlation.

الارتباط الاحصائي Statistical Correlation

الارتباط الاحصائي : بكلمات بسيطة هو دراسة قوة العلاقة relationship بين متغيرين two variable

يمكننا أن نفترض أن توزيع كل متغير يتناسب مع توزيع Gaussian (منحنى الجرس) إذا كانت هذه هي الحالة ، فيمكننا استخدام معامل ارتباط بيرسون Pearson's Coefficient لتلخيص الارتباط بين المتغيرات.

معامل ارتباط بيرسون هو رقم يتراوح بين -١ و ١ حيث يمكن تلخيص هذا الارتباط بين المتغيرين على أحد الأشكال التالية :

١- قيمة المعامل تساوي ١ أو تقترب من ١ وتشير إلى ارتباط إيجابي أو تناسب طردي (أي إذا ازدادت قيمة أحد المتغيرين تزيد الأخرى والعكس صحيح)

2- قيمة المعامل تساوي 0 أو قريبة من الصفر وتشير إلى عدم وجود أي ارتباط أو ارتباط شبه معدوم بين المتغيرين

3- قيمة المعامل تساوي -١ أو تقترب من -١ وتشير إلى ارتباط سلبي أو تناسب عكسي (أي إذا ازدادت قيمة أحد المتغيرين تنقص الأخرى)

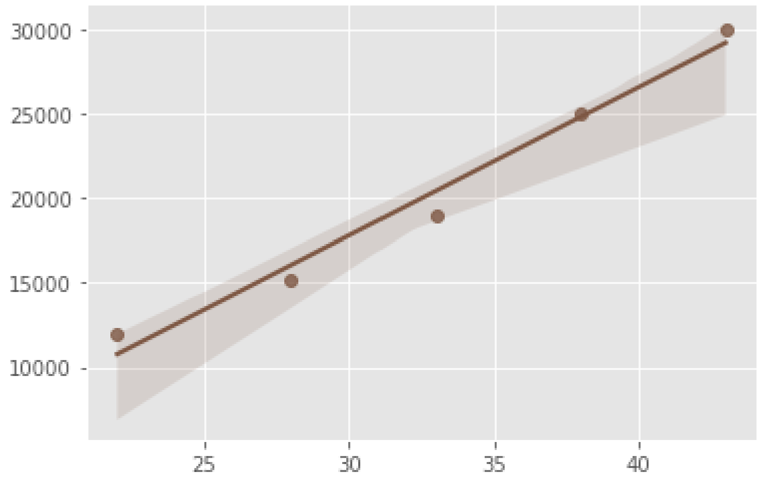
معامل الارتباط لتوزيعات مختلفة لبيانات متغيرين

الارتباط الاحصائي Statistical Correlation

لنقل أننا نريد دراسة قوة الارتباط بين سمتين attributes وهما العمر age والراتب الشهري salary :

```
age = np.array([22, 28, 33, 38, 43])
salary = np.array([12000, 15200, 19000, 25000, 30000])

sns.regplot(x=age, y=salary, color='#7F5741')
plt.show()
```




```
attributes_corr = np.corrcoef(age, salary)
pd.DataFrame(attributes_corr, columns={'age','salary'},
                              index={'age','salary'})
```

	salary	age
salary	1.000000	0.988386
age	0.988386	1.000000

نلاحظ عند استخدام الدالة np.corrcoef أنها تقوم باستخراج مصفوفة لقياس الارتباط حيث أنها تدرس الارتباط بين السمة ونفسها والسمة و السمة الأخرى بطريقة تبديلية حيث أن قطر المصفوفة يشكل دراسة ارتباط السمة مع نفسها وبطبيعة الحال ستكون أقصى ما يمكن أي 1 بينما عند دراسة العلاقة بين العمر و الراتب الشهري نلاحظ قوة الارتباط التي تصل إلى 0.988

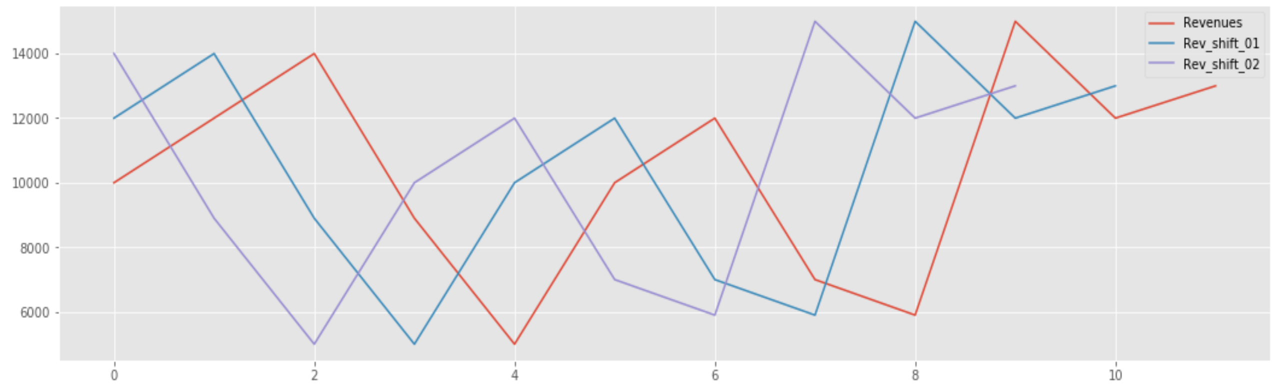
مفهوم الارتباط التلقائي Autocorrelation

الارتباط التلقائي Autocorrelation يختلف قليلًا عن المفهوم السابق حيث أنه يقوم على دراسة السلسلة الزمنية ومقارنتها مع نفسها في فترات زمنية أو خطوات سابقة يطلق عليها اسم التأخيرات Lags والذي يعتمد بشكل رئيسي على مفهوم الإزاحة shifting وقد تم شرحه في الفصل الثاني حيث يتم إزاحة (تأخير) السلسلة الزمنية بمقدار محدد للخلف ومن ثم يتم مقارنة السلسلة الأصلية مع السلسلة المتأخرة ومن هذا المنطلق تم تسميته بالارتباط التلقائي أو التسلسلي !

```python
Revenues = np.array([10000, 12000, 14000, 8900, 5000, 10000,
                     12000, 7000, 5900, 15000, 12000, 13000])
Revenues_shifted_1 = Revenues[1:]
Revenues_shifted_2 = Revenues[2:]

print(f"Revenues_00: {Revenues}\n
        Revenues_01: {Revenues_shifted_1}\n
        Revenues_02: {Revenues_shifted_2}")

ax, fig = plt.subplots(figsize=(18,5))
ax.plot(Revenues)
ax.plot(Revenues_shifted_1)
ax.plot(Revenues_shifted_2)
ax.legend(['Revenues', 'Rev_shift_01', 'Rev_shift_02'])
plt.show()
```




```
Revenues_00: [10000 12000 14000  8900  5000 10000 12000  7000  5900 15000 12000 13000]
Revenues_01: [12000 14000  8900  5000 10000 12000  7000  5900 15000 12000 13000]
Revenues_02: [14000  8900  5000 10000 12000  7000  5900 15000 12000 13000]
```

نشاهد هنا نموذج لسلسلة زمنية لإيرادات أحد المتاجر على مدى 12 شهرًا وقد قمنا بتأخيرها (إزاحتها) مرة بمقدار 1 و مرة أخرى بمقدار 2 ثم تم تصويرها بيانيًا لملاحظة الفرق بين السلسلة الأصلية والسلاسل المتأخرة Lags والتي سيتم تطبيق مفهوم الارتباط التلقائي عليها.

إيجاد الارتباط التلقائي Autocorrelation

الآن سنقوم بتطبيق الارتباط التلقائي بشكل عملي عن طريق دراسة العلاقة بين :

1- السلسلة الأصلية و ذاتها بدون أي تعديلات

2- السلسلة الأصلية و نظيرتها المتأخرة بمقدار فترة واحدة Revenues vs Revenues_shifted_1

3- السلسلة الأصلية و نظيرتها المتأخرة بمقدار فترتين Revenues vs Revenues_shifted_2

```
# إيجاد الارتباط بين السلسلة الأصلية ونظيراتها المتأخرة

lag_0 = np.corrcoef(Revenues, Revenues)[0,1]
lag_1 = np.corrcoef(Revenues[:-1], Revenues_shifted_1)[0,1]
lag_2 = np.corrcoef(Revenues[:-2], Revenues_shifted_2)[0,1]

print(f"Lag 0: {lag_0}\nlag 1: {lag_1}\nlag 2: {lag_2}")
```

```
Lag 0: 1.0
lag 1: 0.09342343040046394
lag 2: -0.4621229967699638
```

نلاحظ أن قيم الارتباط هي على الهيئة التالية :

1- ارتباط قوي 100% وهو أمر محتوم لأننا نقارن السلسلة مع نفسها !
2- ارتباط ضعيف بين السلسلة الأصلية و نظيرتها المتأخرة بمقدار فترة زمنية واحدة !
3- ارتباط سلبي متوسط القوة بين السلسلة الأصلية و نظيرتها المتأخرة بمقدار فترتين !

إيجاد الارتباط التلقائي Autocorrelation

لكن ماذا لو أردنا إيجاد قيم الارتباط التلقائي لأكثر من تأخير في السلسلة الزمنية ؟

من الممكن تصميم Loop يقوم بهذا الغرض لكن لحسن الحظ يوجد من سبقنا بفعل ذلك ، حيث بالإمكان استخدام دالة acf و plot_acf من مكتبة statsmodels لإيجاد وتصوير الارتباط التلقائي لعدد معين من التأخيرات Lags وفي حال عدم تحديد عدد التأخيرات فإن الدالة تقوم بطباعة أقصى عدد منها بشكل تلقائي !

```
from statsmodels.graphics.tsaplots import acf, plot_acf

ac_values = np.round(acf(Revenues, nlags=5,  fft=False),2)
ac_values
```

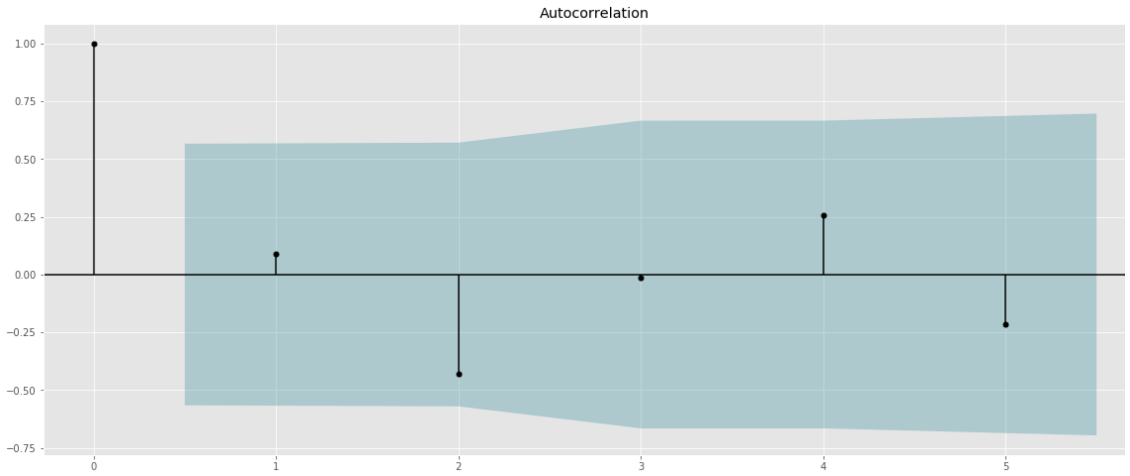
```
array([ 1.  ,  0.09, -0.43, -0.01,  0.25, -0.21])
```

```
from matplotlib.collections import PolyCollection

fig, ax = plt.subplots(figsize=(20,8))
plot_acf(Revenues, ax=ax, lags=5,
         color='#000000',
         vlines_kwargs={"colors": '#000000'})

for item in ax.collections:
    if type(item)==PolyCollection:
        item.set_facecolor('#007687')

plt.show()
```



نلاحظ أنه تم إيجاد وتصوير قيم الارتباط التلقائي حتى التأخير رقم خمسة lag5 وبطبيعة الحال يوجد

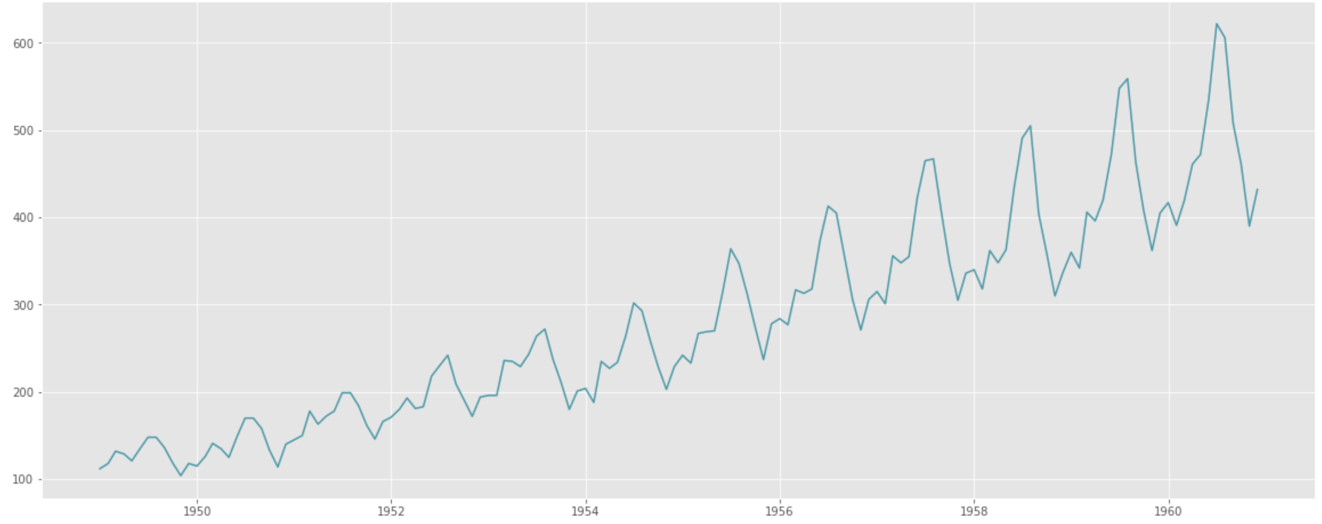
الارتباط التلقائي لبيانات سلسلة زمنية حقيقية

الآن سنقوم بالعمل على بيانات سلسلة زمنية تتضمن أعداد الركاب (المسافرين) على متن أحد الخطوط الجوية شهريًا من سنة 1949 إلى سنة 1960 ونطبق عليه مفهوم الارتباط التلقائي.

```
from IPython.display import display

# سلسلة زمنية تحتوي على أعداد المسافرين على أحد خطوط الطيران
airline_passengers = pd.read_csv('datasets/airline-passengers.csv',
                                 index_col='Month', parse_dates=True)
display(airline_passengers)

# التصوير البياني للسلسلة الزمنية
fig, ax = plt.subplots(figsize=(20,8))
ax.plot(airline_passengers, c='#489AAA');
```



Month	Passengers
1949-01-01	112
1949-02-01	118
1949-03-01	132
1949-04-01	129
1949-05-01	121
...	...
1960-08-01	606
1960-09-01	508
1960-10-01	461
1960-11-01	390
1960-12-01	432

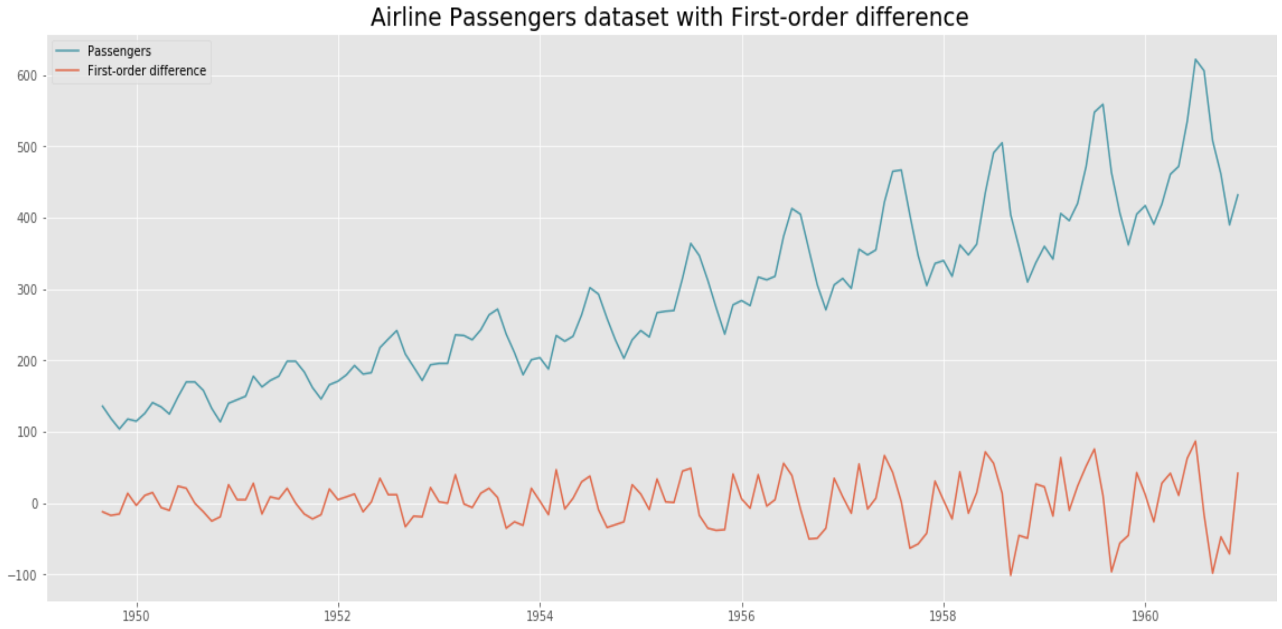
144 rows × 1 columns

الارتباط التلقائي لبيانات سلسلة زمنية حقيقية

قبل أن نقوم بتطبيق دالة الارتباط التلقائي plot_acf يجب أن نجعل السلسلة الزمنية ثابتة stationary وهو مفهوم سندرسه بشكل موسع في الفصل القادم كل ما علينا الآن هو تطبيق المفهوم عمليًا بأسهل طريقة وهي عن طريق إيجاد الفرق من الرتبة الأولى First-Order-Differencing للسلسلة الزمنية وهذا المفهوم قد تم شرحه بالفصل الثاني :

```
# إيجاد الفرق من الرتبة الأولى للسلسلة الزمنية
airline_passengers['Passengers_Diff'] =
airline_passengers['Passengers'].diff(periods=1)
airline_passengers = airline_passengers.dropna()

# التصوير البياني للسلسلة قبل وبعد إيجاد الفرق من الرتبة الأولى
fig, ax = plt.subplots(figsize=(20,8))
ax.plot(airline_passengers['Passengers'], label='Passengers', c='#489AAA')
ax.plot(airline_passengers['Passengers_Diff'], label='First-order-difference',
        color='#E46B48')
ax.legend()
plt.title('Airline Passengers dataset with First-order difference', size=20)
plt.show()
```



لا يمكن القول بأن السلسلة أصبحت ثابتة بشكل كلي لوجود انحراف معياري كبير في نهاية السلسلة الجديدة ولكن ستفي بالغرض حاليًا حيث كما أشرنا سابقًا أن الهدف هو فهم الارتباط التلقائي والتعمق بموضوع الثبات Stationarity سيكون في الفصل التالي !

الارتباط التلقائي لبيانات سلسلة زمنية حقيقية

يجيب الارتباط التلقائي Autocorrelation على السؤال التالي: ما مدى ارتباط عدد الركاب هذا الشهر بعدد الركاب في الشهر السابق؟". هنا ، يشير الشهر السابق إلى سلسلة زمنية متأخرة بمقدار شهر واحد أو Lag 1 وكذلك الأمر مع التأخر بشهرين Lag 2 حيث يشير إلى مدى ارتباط الركاب هذا الشهر بعدد الركاب قبل شهرين من الآن وهكذا دواليك !

يمكنك إعادة صياغة السؤال من خلال السؤال عن مدى ارتباط عدد الركاب هذا الشهر بعدد الركاب قبل عام. بعد ذلك ، ستكون قيمة التأخر 12 شهرًا. وهذا سؤال رائع ، حيث يمكن رؤية الموسمية السنوية من الرسم البياني.

```
# قيم الارتباط التلقائي للسلسلة الزمنية
acf(airline_passengers['Passengers_Diff'], fft=False)
```

Lag 12

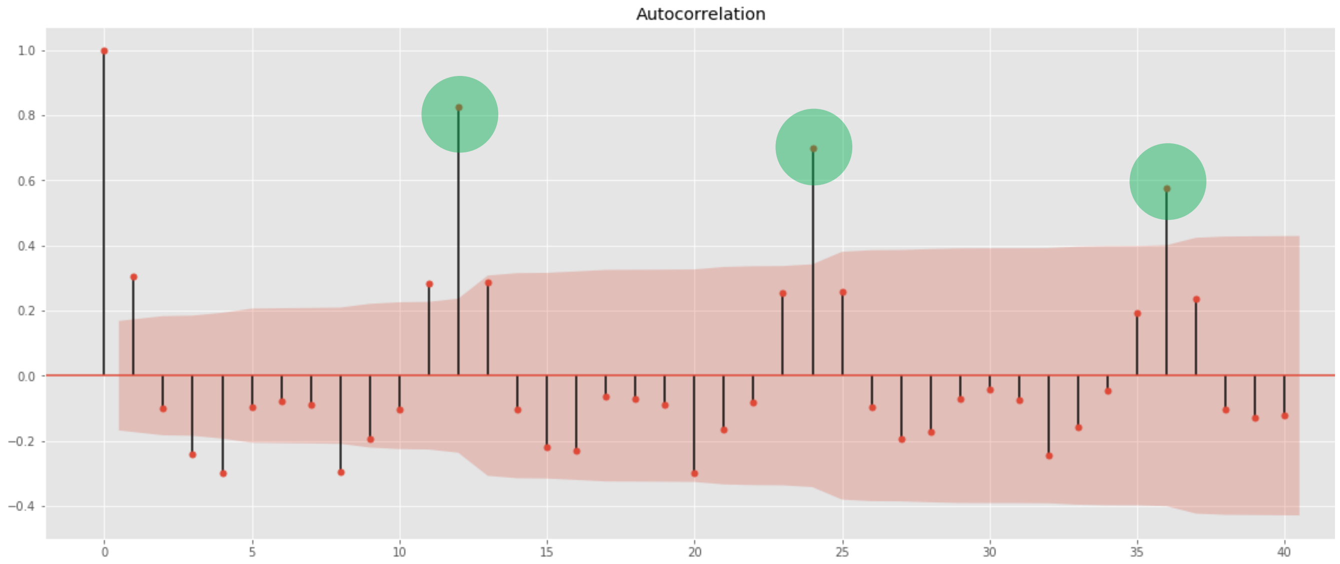
```
array([ 1.        ,  0.30352854, -0.09967008, -0.2403157 , -0.30055333,
       -0.095556  , -0.07960294, -0.09072591, -0.29559894, -0.19302286,
       -0.10236829,  0.28295691,  0.82692358,  0.28529547, -0.10236375,
       -0.22105555, -0.23080827, -0.06479552, -0.06934284, -0.0875783 ,
       -0.29713157, -0.16487266, -0.08148927,  0.25532906,  0.69977368,
        0.25747719, -0.09630781, -0.19399232, -0.17380209, -0.07152462,
       -0.0437688 , -0.07603641, -0.24635355, -0.15770001, -0.04520619,
        0.19297623,  0.57716557,  0.23663458, -0.10289863, -0.12993312,
       -0.12263335])
```

Lag 24

نلاحظ أن الارتباطات القوية تحدث بين السلسلة الزمنية الأصلية و السلسلة المتأخرة بمضاعفات العدد 12 أي كل سنة و أيضًا نلاحظ أن قوة الارتباط تنخفض بشكل تدريجي مع ازدياد التأخيرات وهذا ما سنصوره بيانيًا في الشريحة التالية .

الارتباط التلقائي لبيانات سلسلة زمنية حقيقية

```
fig, ax = plt.subplots(figsize=(20,8))
plot_acf(airline_passengers['Passengers_Diff'], lags=40, ax=ax)
plt.show()
```



- يؤكد المخطط ما استنتجناه في الشريحة السابقة حول الارتباط في التأخر Lag 12. يظهر الشيء نفسه في التأخر Lag 24، وكذلك في التأخر Lag 36 لكن قوة الارتباط تنخفض بازدياد عدد التأخيرات.

- أمر آخر يجب ملاحظته هو المنطقة المظللة حيث أن أي شيء بداخلها لا يعد ذو دلالة إحصائية و ما يخرج منها ستعتبره ذو دلالة إحصائية (يوجد ارتباط) مع السلسلة الأصلية.

مفهوم الارتباط التلقائي الجزئي Partial Autocorrelation

الارتباط التلقائي Autocorrelation يدرس العلاقة بين مشاهدات observations السلسلة الزمنية الأصلية ومشاهداتها في خطوات زمنية سابقة حيث تتضمن كل من الارتباط المباشر والارتباطات غير المباشرة. هذه الارتباطات غير المباشرة هي دالة خطية لعلاقة المشاهدات مع المشاهدات في الخطوات الزمنية المتداخلة.

أما الارتباط التلقائي الجزئي Partial Autocorrelation هو ملخص للعلاقة (الارتباط) بين بيانات (مشاهدات) السلسلة الزمنية الأصلية مع مشاهداتها في خطوات زمنية سابقة مع إزالة علاقات المشاهدات المتداخلة بينهما .

أي أنه يعمل بنفس طريقة الارتباط التلقائي العادي من حيث أنه يوضح ارتباط تسلسل مع نفسه متأخرًا بعدد من الوحدات الزمنية. لكن هناك منعطف وهو إظهار التأثير المباشر فقط و إزالة جميع التأثيرات الوسيطة.

على سبيل المثال : لو عدنا إلى مثال عدد الركاب الشهري و أردنا معرفة العلاقة المباشرة (الارتباط المباشر) بين عدد الركاب في هذا الشهر و عددهم قبل 12 شهرًا مضت فإننا لن نهتم بأي شيء حدث بينهما (lag 2-lag 11) على عكس الارتباط التلقائي الذي يأخذ بالاعتبار جميع التأخيرات المتداخلة بينهم لن نخوض في المنطق الرياضي وراء وذلك ويكفي معرفة أن هذه الارتباطات غير المباشرة تسعى دالة الارتباط التلقائي الجزئي إلى إزالتها.

الارتباط التلقائي الجزئي لبيانات سلسلة زمنية حقيقية

الآن سنقوم بإيجاد الارتباط التلقائي الجزئي Partial Autocorrelation على البيانات السابقة الخاصة بعدد الركاب على متن الخطوط الجوية

من خلال دالتي pacf() و plot_pacf() المضمنة في مكتبة statsmodels :

```python
from statsmodels.graphics.tsaplots import pacf, plot_pacf

pacf_values = pacf(airline_passengers['Passengers_Diff'], nlags=30)
np.round(pacf_values, 2)
```

```
array([ 1.  ,  0.3 , -0.22, -0.16, -0.23,  0.01, -0.2 , -0.16, -0.49,
       -0.28, -0.67, -0.42,  0.61, -0.05, -0.3 ,  0.02,  0.14,  0.09,
       -0.12,  0.01, -0.22, -0.12,  0.  , -0.07, -0.16, -0.17, -0.15,
       -0.02, -0.16, -0.48, -0.41])
```

Lag 24

Lag 10

Lag 12

ما نلاحظه أنه يوجد ارتباط سلبي في التأخر Lag 10 وأن مقدار الارتباط في التأخر Lag 12 قد انخفض من 0.82 إلى 0.61 مما يشير أن العلاقة المباشرة أضعف بقليل عما كان عليه في الارتباط التلقائي كما أن الارتباط في التأخر Lag 24 قد أصبح ارتباط سلبي ضعيف وهذا بفعل أن الارتباط التلقائي الجزئي يزيل جميع الارتباطات المتداخلة (الارتباطات الغير المباشرة) كما أشرنا سابقًا ويركز فقط على الارتباط المباشر !

الارتباط التلقائي الجزئي لبيانات سلسلة زمنية حقيقية

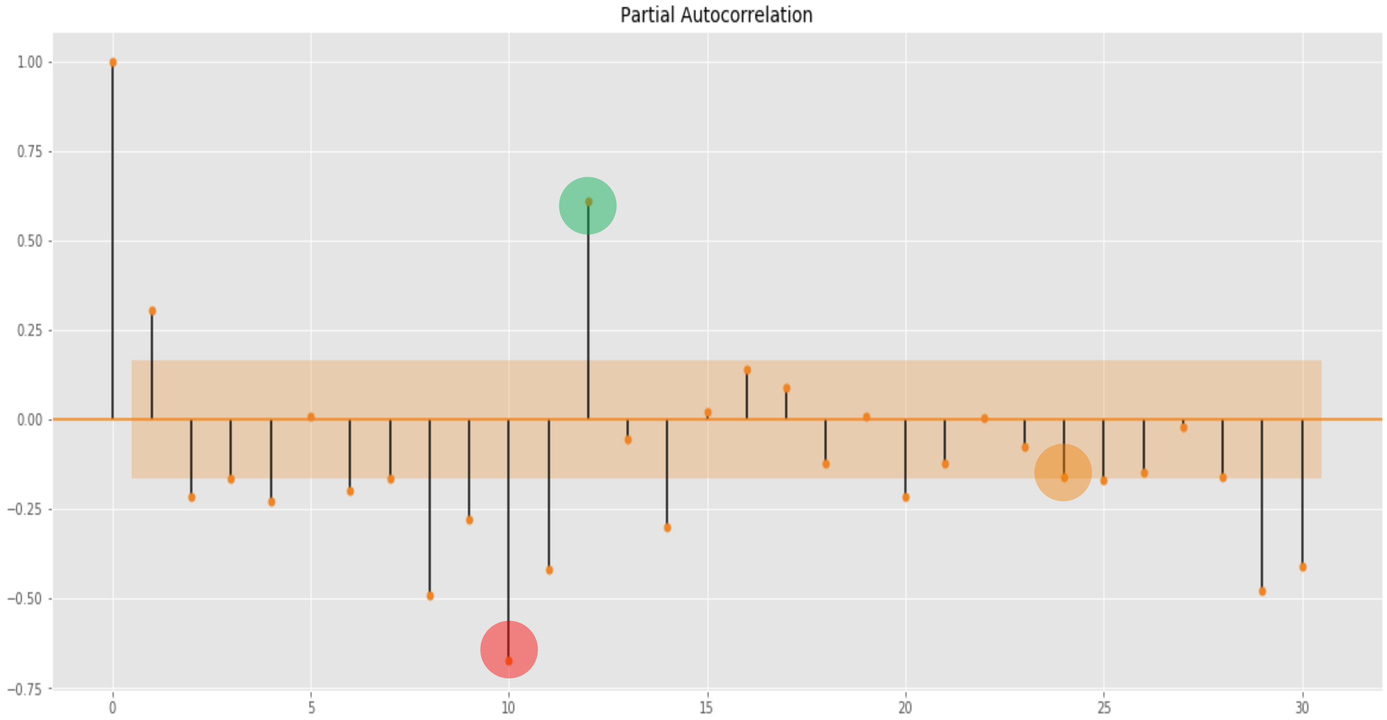
تصوير الارتباط التلقائي الجزئي

```python
# Plot partial autocorrelation
fig, ax = plt.subplots(figsize=(20,8))

plot_pacf(airline_passengers['Passengers_Diff'],
          lags=30, ax=ax, color='#F38307',
          vlines_kwargs={"colors": '#000000'})

for item in ax.collections:
    if type(item)==PolyCollection:
        item.set_facecolor('#F38307')

plt.show()
```



لا يزال هناك سؤال مهم متبقي - كيف نستخدم ونفسر مخططات الارتباط التلقائي والارتباط التلقائي الجزئي ACF و PACF للتنبؤ بالسلاسل الزمنية؟.

سنجيب عن هذا السؤال فيما يلي !

تفسير مخططات الارتباط التلقائي والارتباط التلقائي الجزئي

تتطلب منك نماذج السلاسل الزمنية التي ستتعرف عليها قريبًا : مثل الانحدار التلقائي Auto Regression (AR) و المتوسطات المتحركة Moving Averages(MA) أو تركيباتها المختلفة ARMA تحديد علامة أو أكثر يمكن استنتاجها من مخططات الارتباط التلقائي والارتباط التلقائي الجزئي pacf و acf من ضمن هذه العلامات !

- إذا انخفض مخطط ACF بشكل تدريجي وانخفض مخطط PACF بشكل فوري فاستخدم نموذج الانحدار التلقائي AR.
- إذا انخفض مخطط ACF بشكل فوري وانخفض PACF تدريجيًا ، فاستخدم نموذج المتوسط المتحرك MA.
- إذا انخفض كل من ACF و PACF تدريجيًا ، فقم بدمج نماذج الانحدار التلقائي والمتوسط المتحرك ARMA .
- إذا انخفض كل من ACF و PACF بشكل فوري (بدون فترات تأخير كبيرة) ، فمن المحتمل أنك لن تكون قادرًا على نمذجة السلاسل الزمنية والتنبؤ بها.

ومع ذلك ، فإن قراءة مخططات ACF و PACF تمثل تحديًا ، وأنت كعالم أو محلل بيانات تتمتع بقدرة متميزة من استخدام البحث على الشبكة العنكبوتية للعثور على قيم المعلمات المثلى. تحتوي مجموعة المعلمات المثلى على أدنى خطأ مثل مقياس MAPE أو أقل مقدار للجودة العامة مثل AIC والتي تعتبر كمقاييس للسلاسل الزمنية سنغطيها في هذه السلسلة قريبًا ، لذا ترقبوا ذلك.