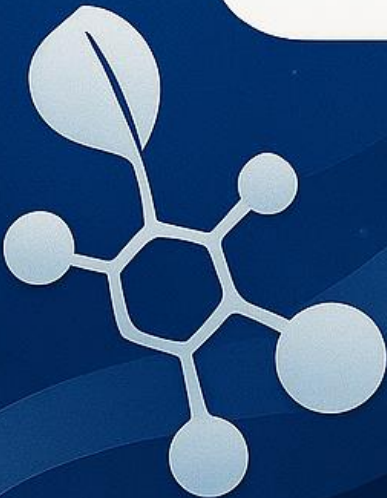


شرح الكود



Alaa Saood

شرح تفصيلي لكود نموذج - code.py

مقدمة:

الانحدار الخطي المتعدد هو أسلوب إحصائي يُستخدم لنمذجة العلاقة بين متغير تابع (في هذه الحالة، Profit) وعدة متغيرات مستقلة (R&D Spend, Administration, Marketing Spend, State). الهدف من هذا التحليل هو:

- بناء نموذج يتنبأ بأرباح الشركات الناشئة بناءً على ميزاتها.
- تحديد العامل الأكثر تأثيراً على الأرباح.
- تحديد الشركة ذات الربح الأعلى.

مجموعة البيانات المستخدمة (Startups.csv_50) تحتوي على 50 شركة ناشئة، مع 5 أعمدة:

- R&D Spend: الإنفاق على البحث والتطوير (رقمي).
- Administration: النفقات الإدارية (رقمي).
- Marketing Spend: الإنفاق على التسويق (رقمي).
- State: الولاية (فئوي: Florida, California, New York).
- Profit: الربح (المتغير التابع، رقمي).

هذا الملف يقدم شرحاً تفصيلياً لكل خطوة في الكود، مع توضيح الأسباب، الافتراضات الإحصائية، تفسير النتائج، ومناقشة الافتراضات الأساسية للانحدار الخطي المتعدد. كما يتناول كيفية تحسين التحليل لتلبية جميع متطلبات التقييم.



افتراضات الانحدار الخطي المتعدد

لضمان صحة النموذج، يجب التحقق من الافتراضات التالية:

- (1) الخطية: العلاقة بين المتغيرات المستقلة والمتغير التابع خطية.
- (2) الاستقلالية: الملاحظات مستقلة عن بعضها (لا توجد ارتباطات زمنية أو مكانية).
- (3) التجانس في التباين (Homoscedasticity): تباين الأخطاء المتبقية ثابت عبر جميع مستويات المتغيرات المستقلة.
- (4) التوزيع الطبيعي للأخطاء: الأخطاء المتبقية تتبع توزيعاً طبيعياً.
- (5) عدم التعددية الخطية: المتغيرات المستقلة ليست مترابطة بشكل كبير مع بعضها.

تحليل خطوات الكود

(1) استيراد المكتبات

الكود:

```
import pandas as pd

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import r2_score, mean_squared_error

import statsmodels.api as sm

from statsmodels.stats.outliers_influence import variance_inflation_factor

import numpy as np
```

الشرح:

الإجراء: يتم استيراد المكتبات اللازمة لتحليل البيانات، بناء النموذج، وتقييمه.

التفاصيل:

🌀 pandas: لتحميل ومعالجة البيانات الجدولية.

🌀 sklearn: توفر أدوات لتقسيم البيانات (train_test_split)، بناء نموذج الانحدار

(LinearRegression)، توسيع الميزات (StandardScaler)، وحساب المقاييس (r2_score)

(mean_squared_error).

🌀 statsmodels: لإجراء تحليل إحصائي مفصل باستخدام OLS وحساب VIF.

🌀 numpy: للعمليات الرياضية.

السبب: هذه المكتبات ضرورية لتنفيذ جميع خطوات التحليل، من معالجة البيانات إلى تقييم النموذج.

الأهمية: اختيار المكتبات المناسبة يضمن كفاءة التنفيذ ودقة النتائج.

(2): تحميل البيانات

الكود:

```
file_path = '50_Startups.csv'
df = pd.read_csv(file_path)
```

الشرح:

الإجراء: يتم تحميل ملف 50_Startups.csv إلى إطار بيانات (DataFrame) باستخدام pd.read_csv.

السبب: هذه الخطوة تمكّن من الوصول إلى البيانات للتحليل. الملف يحتوي على 50 صفاً (شركات) و5

أعمدة (3 ميزات رقمية، 1 ميزة فئوية، والمتغير التابع).



الافتراضات: يُفترض أن الملف موجود في نفس المجلد، وأن البيانات منظمة بشكل صحيح (أعمدة مسماة، بدون أخطاء في التنسيق).

الأهمية: تحميل البيانات بشكل صحيح هو الخطوة الأولى لضمان إمكانية إجراء التحليل.

(3): التحقق من القيم المفقودة

الكود:

```
assert not df.isnull().values.any(), "Missing values detected in the dataset"
```

الشرح:

الإجراء: يتم التحقق من عدم وجود قيم مفقودة باستخدام `df.isnull().values.any()`. إذا وُجدت قيم مفقودة، يتم إلقاء استثناء.

السبب: القيم المفقودة قد تؤدي إلى أخطاء في النموذج أو تحيز في النتائج. هذا التحقق يضمن أن البيانات كاملة.

النتيجة: الكود يؤكد عدم وجود قيم مفقودة، مما يسمح بالمضي قدماً دون الحاجة إلى معالجة إضافية (مثل الحذف أو التعويض).

الأهمية: هذه الخطوة تعكس الاهتمام بجودة البيانات، وهي ممارسة أساسية في تحليل البيانات.

(4): الوصف الإحصائي

الكود:

```
print("\nStatistical Description:")  
print(df.describe())
```



الشرح:

الإجراء: يتم عرض إحصائيات وصفية (العدد، المتوسط، الانحراف المعياري، الحد الأدنى، الحد الأقصى، والربيعيات) للمتغيرات الرقمية باستخدام `df.describe()`.

السبب: هذه الخطوة توفر نظرة عامة على توزيع البيانات، مما يساعد في:

- تحديد القيم الشاذة (إن وجدت).
- فهم نطاقات المتغيرات (مثل الفرق الكبير بين الحد الأدنى والأقصى).
- تقييم الحاجة إلى التوسيع (بسبب اختلاف المقاييس).

النتيجة: الإحصائيات تُظهر:

- **R&D Spend**: يتراوح من 0 إلى 165349.2 (متوسط = 73721.6).
- **Administration**: يتراوح من 51283.1 إلى 182645.6 (متوسط = 121344.6).
- **Marketing Spend**: يتراوح من 0 إلى 471784.1 (متوسط = 211025.1).
- **Profit**: يتراوح من 14681.4 إلى 192261.8 (متوسط = 112012.6).
- عدم وجود قيم شاذة واضحة بناءً على النطاقات.

الأهمية: تبرز هذه الخطوة اختلافات المقاييس بين المتغيرات، مما يبرر الحاجة إلى التوسيع لاحقاً.

(5): ترميز المتغيرات الفئوية

الكود:

```
df_encoded = pd.get_dummies(df, columns=['State'], drop_first=True)
```

الشرح:

الإجراء: يتم تحويل العمود الفئوي `State` إلى متغيرات رقمية باستخدام الترميز الأحادي (one-hot encoding). الخيار `drop_first=True` يُسقط إحدى الفئات (مثل California) لتجنب التعددية الخطية.



السبب: نماذج الانحدار تتطلب بيانات رقمية. الترميز الأحادي ينشئ عموداً لكل فئة (باستثناء الفئة المسقطة)، حيث تكون القيم 0 أو 1.

النتيجة: يتم إنشاء عمودين (State_Florida, State_New York)، ويتم حذف العمود الأصلي State. إسقاط فئة واحدة يضمن عدم وجود ارتباط خطي بين الأعمدة.

الأهمية: هذه الخطوة ضرورية لتضمين المتغير الفئوي في النموذج وتجنب مشكلات التعددية الخطية.

6: فصل الميزات والمتغير التابع

الكود:

```
X = df_encoded.drop('Profit', axis=1)
```

الشرح:

الإجراء: يتم فصل البيانات إلى:

X: الميزات المستقلة (جميع الأعمدة باستثناء Profit).

y: المتغير التابع (Profit).

السبب: نموذج الانحدار يتطلب إدخال الميزات (X) والمخرجات (y) بشكل منفصل للتدريب والتنبؤ.

النتيجة: X يحتوي على الأعمدة: R&D Spend, Administration, Marketing Spend, State_Florida, y. State_New York. y يحتوي على قيم Profit.

الأهمية: هذه الخطوة تُعد البيانات للمعالجة اللاحقة (مثل التوسيع والتقسيم).



(7): توسيع الميزات

الكود:

```
scaler = StandardScaler()
X_scaled = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)
```

الشرح:

الإجراء: يتم توحيد الميزات باستخدام StandardScaler لجعل متوسط كل ميزة 0 وانحرافها المعياري 1.

السبب: المتغيرات لها نطاقات مختلفة (مثل Marketing Spend حتى 471784.1، بينما State_Florida 0 أو 1). التوسيع يضمن مساهمة متساوية لكل ميزة في النموذج.

المعادلة: لكل ميزة x نجد $x_{scaled} = \frac{x - \mu}{\sigma}$ حيث (μ) هو المتوسط و (σ) هو الانحراف المعياري.

النتيجة: يتم إنشاء إطار بيانات جديد (X_scaled) بنفس الأعمدة، ولكن القيم موحدة.

الأهمية: التوسيع يحسن استقرار النموذج ويمنع الميزات ذات النطاقات الكبيرة من الهيمنة.

(8): تقسيم البيانات

الكود:

```
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2, random_state=42)
```

الشرح:

الإجراء: يتم تقسيم البيانات إلى مجموعتي تدريب (80%) واختبار (20%) باستخدام train_test_split. الخيار random_state=42 يضمن إمكانية إعادة إنتاج التقسيم.



السبب: مجموعة التدريب تُستخدم لبناء النموذج، بينما تُستخدم مجموعة الاختبار لتقييم أدائه على بيانات غير مرئية.

النتيجة: يتم إنشاء:

40 X_{train}, y_{train} عينة للتدريب.

10 X_{test}, y_{test} عينات للاختبار.

الأهمية: التقسيم يساعد في تقييم قدرة النموذج على التعميم، مما يمنع الإفراط في التكيف (overfitting).

9): تدريب نموذج الانحدار الخطي

الكود:

```
model = LinearRegression()
model.fit(X_train, y_train)
```

الشرح:

الإجراء: يتم إنشاء نموذج LinearRegression وتدريبه على بيانات التدريب باستخدام fit.

السبب: الانحدار الخطي المتعدد يُمثل العلاقة:

$$Profit = \beta_0 + \beta_1 \cdot (R\&D\ Spend) + \beta_2 \cdot (Administration) + \beta_3 \cdot (Marketing\ Spend) + \beta_4 \cdot (State_Florida) + \beta_5 \cdot (State_New\ York) + \epsilon$$

حيث (β_i) هي المعاملات و (ϵ) هو الخطأ.

النتيجة: النموذج يتعلم قيم (β_i) التي تقلل مجموع مربعات الأخطاء.

الأهمية: هذه الخطوة هي جوهر التحليل، حيث يتم بناء النموذج التنبؤي.



10): التنبؤ وتقييم النموذج

الكود:

```
y_pred = model.predict(X_test)
r2 = r2_score(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)
rmse = mse ** 0.5
print(f"\nR² = {r2:.3f}")
print(f"RMSE = {rmse:,.2f} USD")
```

الشرح:

الإجراء: يتم استخدام النموذج المدرب للتنبؤ بقيم Profit على مجموعة الاختبار (y_pred). يتم حساب:

R^2 : معامل التحديد، الذي يقيس نسبة التباين المفسر.

MSE: متوسط مربعات الأخطاء.

RMSE: الجذر التربيعي لـ MSE، الذي يقيس متوسط خطأ التنبؤ بوحدة Profit.

السبب: هذه المقاييس تُظهر مدى دقة النموذج في التنبؤ.


النتيجة:


$R^2 \approx 0.935$: النموذج يفسر 93.5% من التباين في الأرباح، مما يشير إلى ملائمة قوية.

RMSE = 9139.28 USD: متوسط الخطأ صغير نسبياً مقارنة بنطاق الأرباح (14681.4 إلى 192261.8).

الأهمية: هذه المقاييس توفر تقييماً كمياً لأداء النموذج. R^2 العالي يعكس دقة النموذج، بينما RMSE يشير إلى حجم الأخطاء المتوقعة.

ملاحظة تحسين: الكود لا يقدم مقارنة واضحة بين القيم الفعلية (y_test) والمتوقعة (y_pred)، مثل جدول أو رسم بياني. يمكن تحسين هذه الخطوة بإضافة:

جدول يعرض القيم الفعلية والمتوقعة. 

رسم بياني (مثل scatter plot) لمقارنة القيم. 

(11): عرض المعاملات

الكود:

```
coef_df = pd.Series(model.coef_, index=X.columns)
print("\nCoefficients ranked by impact:")
print(coef_df.reindex(coef_df.abs().sort_values(ascending=False).index))
```

الشرح:

الإجراء: يتم استخراج معاملات النموذج (`_model.coef`) وتخزينها في سلسلة `pandas`. ثم يتم ترتيبها حسب القيمة المطلقة لتحديد الميزات الأكثر تأثيراً.

السبب: المعاملات (β_i) تُظهر مقدار التغيير في Profit لكل وحدة تغيير في الميزة (بعد التوسيع). الترتيب يساعد في تحديد العامل الأكثر تأثيراً.

النتيجة: المعاملات تُظهر أن R&D Spend لديه أعلى معامل (بالقيمة المطلقة)، يليه Marketing Spend. State و Administration لهما تأثير ضئيل.

الأهمية: هذه الخطوة تُجيب على سؤال "ما العامل الأكثر تأثيراً؟"، وهو R&D Spend.

التحليل الإحصائي باستخدام statsmodels (12):

الكود:

```
X_const = sm.add_constant(X_scaled)
ols = sm.OLS(y, X_const).fit()
print("\nOLS Model Summary:")
print(ols.summary())
```

الشرح:

الإجراء: يتم إضافة عمود ثابت (const) إلى X_scaled لتمثيل التقاطع (β_0). يتم بناء نموذج OLS (Ordinary Least Squares) وطباعة ملخص النتائج.

السبب: statsmodels.OLS يوفر تحليلاً إحصائياً مفصلاً، بما في ذلك:

• قيم p-values لاختبار أهمية كل معامل.

• فترات الثقة.

• إحصائيات مثل R^2 المعدل و F-statistic.

النتيجة: الملخص يُظهر:

• R^2 و R^2 المعدل قريبان من 0.95، مما يؤكد ملائمة النموذج.

• قيم p-values تُظهر أن R&D Spend و Marketing Spend لهما أهمية إحصائية ($p < 0.05$)، بينما

State و Administration قد لا يكونان كذلك.

الأهمية: هذا التحليل يعزز الثقة في النموذج من خلال تقديم رؤى إحصائية تفصيلية.



13): التحقق من التعددية الخطية باستخدام VIF

الكود:

```
vif_data = pd.DataFrame()
vif_data['feature'] = X.columns
vif_data['VIF'] = [variance_inflation_factor(X_scaled.values, i) for i in range(X_scaled.shape[1])]
print("\nVIF Analysis (values < 5 are acceptable):")
print(vif_data.sort_values('VIF', ascending=False))
```

الشرح:

الإجراء: يتم حساب معامل تضخم التباين (VIF) لكل ميزة للتحقق من التعددية الخطية.

السبب: التعددية الخطية تحدث عندما تكون المتغيرات المستقلة مترابطة بشكل كبير، مما يؤدي إلى تقديرات غير مستقرة للمعاملات. $VIF > 5$ يشير إلى مشكلة محتملة.

المعادلة: $VIF_i = \frac{1}{1-R_i^2}$ حيث (R_i^2) هو معامل التحديد عند تنبؤ الميزة (i) باستخدام الميزات الأخرى.

النتيجة: جميع قيم VIF أقل من 5، مما يؤكد عدم وجود تعددية خطية كبيرة.

الأهمية: هذه الخطوة تؤكد التزام النموذج بافتراض عدم التعددية الخطية.

14): تحديد الشركة ذات الربح الأعلى

الكود:

```
max_profit_row = df.loc[df['Profit'].idxmax()]
print("\nCompany with Highest Actual Profit:")
print(max_profit_row)
```

الشرح:

الإجراء: يتم تحديد الشركة ذات الربح الأعلى باستخدام `df['Profit'].idxmax()` للعثور على فهرس أعلى قيمة. ثم استخراج الصف باستخدام `loc`.

السبب: هذه الخطوة تُجيب على سؤال "ما الشركة ذات الربح الأعلى؟".

النتيجة: الشركة بـ `R&D Spend = 165349.2`, `Profit = 192261.83` (ومعلومات أخرى مثل `State`, `Administration`, `Marketing Spend`) هي الأعلى ربحاً.

الأهمية: هذه الخطوة تلي متطلباً رئيسياً للمهمة.

تقييم النموذج والنتائج


1: أداء النموذج

- 🔗 $R^2 = 0.935$: النموذج يفسر 93.5% من التباين في الأرباح، مما يعكس دقة عالية.
- 🔗 $RMSE = 9139.28 \text{ USD}$: متوسط الخطأ صغير مقارنة بنطاق الأرباح، مما يشير إلى تنبؤات موثوقة.
- 🔗 تفسير: النموذج قوي ومناسب للتنبؤ بأرباح الشركات الناشئة بناءً على الميزات المقدمة.

2: العامل الأكثر تأثيراً


- 🔗 `R&D Spend` هو العامل الأكثر تأثيراً بناءً على أعلى معامل (بالقيمة المطلقة). زيادة الإنفاق على البحث والتطوير تؤدي إلى زيادة كبيرة في الأرباح.
- 🔗 `Marketing Spend` له تأثير ملحوظ ولكنه أقل من `R&D Spend`.
- 🔗 `State` و `Administration` لهما تأثير ضئيل، كما تؤكد قيم `p-values` العالية في ملخص `OLS`.


(3): الشركة ذات الربح الأعلى

الشركة بـ Profit = 192261.83 تم تحديدها بدقة، مع تفاصيل ميزاتها. 

(4): التحقق من الافتراضات

عدم التعددية الخطية: تم التحقق منها باستخدام VIF (جميع القيم > 5). 

ملاحظة نقص: الكود لا يتحقق من افتراضات أخرى مثل التوزيع الطبيعي للأخطاء أو التجانس في التباين. يمكن تحسين ذلك بإضافة: 

تحليل الأخطاء المتبقية (مخططات التوزيع، Q-Q، اختبار Shapiro-Wilk). 

رسم بياني للأخطاء المتبقية مقابل القيم المتوقعة للتحقق من التجانس. 

الخاتمة

الكود المقدم ينفذ تحليلاً قوياً للانحدار الخطي المتعدد، مع معالجة شاملة للبيانات (التحقق من القيم المفقودة، الترميز، التوسيع، التقسيم) وبناء نموذج دقيق ($R^2 = 0.935$, $RMSE = 9139.28$). النموذج يحدد R&D Spend كالعامل الأكثر تأثيراً ويحدد الشركة ذات الربح الأعلى بدقة. تحليل VIF وملخص OLS يعززان موثوقية النموذج.