Code Step by Steep Guide

Title: Predicting GPA Scores of University Students

Introduction:

In this project, we aim to predict GPA scores of university students based on their SAT scores and attendance. The data used for this analysis is a dummy dataset named '1.03. Dummies.csv'. We will perform regression analysis using the statsmodels library in Python and present the results along with recommendations.

Import relevant packages.

```
In [20]: import pandas as pd
         import numpy as np2
         import matplotlib.pyplot as plt
         import statsmodels.api as sm
         import seaborn as sns
         sns.set()
```

Data Exploration and Pre-processing:

In this section, we load the data into a pandas DataFrame called 'raw_data'.

```
In [2]: raw_data = pd.read_csv ('1.03. Dummies.csv')
        raw_data
```

Out[2]:

	SAT	GPA	Attendance
0	1714	2.40	No
1	1664	2.52	No
2	1760	2.54	No
3	1685	2.74	No
4	1693	2.83	No
...	...	...	...
79	1936	3.71	Yes
80	1810	3.71	Yes
81	1987	3.73	No
82	1962	3.76	Yes
83	2050	3.81	Yes

84 rows × 3 columns

Regression Analysis:

Import statsmodels.api as sm

- Pre-process the data.
- Mapping attendance into Yes =1 and No =0

```
In [15]: data = raw_data.copy()

In [16]: data['Attendance'] = data['Attendance'].map ({'Yes':1, 'No':0})
         data
```

Out[16]:

	SAT	GPA	Attendance
0	1714	2.40	0
1	1664	2.52	0
2	1760	2.54	0
3	1685	2.74	0
4	1693	2.83	0
...	...	...	...
79	1936	3.71	1
80	1810	3.71	1
81	1987	3.73	0
82	1962	3.76	1
83	2050	3.81	1

84 rows × 3 columns

- Define the predictor and target variables

```
In [23]: y = data ['GPA']
         x1 = data [['SAT', 'Attendance' ]]
```

- Add a constant term to the predictors

```
In [24]: x = sm.add_constant (x1)
         results = sm.OLS(y,x). fit()
         results.summary()
```

We have preprocessed the data by mapping the 'Attendance' column values to 1 for 'Yes' and 0 for 'No'. Then, we define the target variable ('GPA') and the predictor variables ('SAT' and 'Attendance'). Next, we add a constant term to the predictors using `**sm.add_constant()**`. Finally, we fit an Ordinary Least Squares (OLS) model to the data using `**sm.OLS()**` and obtain the results.

Results and Interpretation:

We print the summary of the regression results to analyse the coefficients, their statistical significance, and other relevant statistics.

```
C:\Users\CHUKS\Antonda3\lib\site-packages\numpy\core\fromnumeric.
and will be removed in a future version. Use numpy.ptp instead.
  return ptp(axis=axis, out=out, **kwargs)
```

Out[24]:

OLS Regression Results

Dep. Variable:	GPA	**R-squared:**	0.565
Model:	OLS	**Adj. R-squared:**	0.555
Method:	Least Squares	**F-statistic:**	52.70
Date:	Wed, 19 Aug 2020	**Prob (F-statistic):**	2.19e-15
Time:	13:24:38	**Log-Likelihood:**	25.798
No. Observations:	84	**AIC:**	-45.60
Df Residuals:	81	**BIC:**	-38.30
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.6439	0.358	1.797	0.076	-0.069	1.357
SAT	0.0014	0.000	7.141	0.000	0.001	0.002
Attendance	0.2226	0.041	5.451	0.000	0.141	0.304

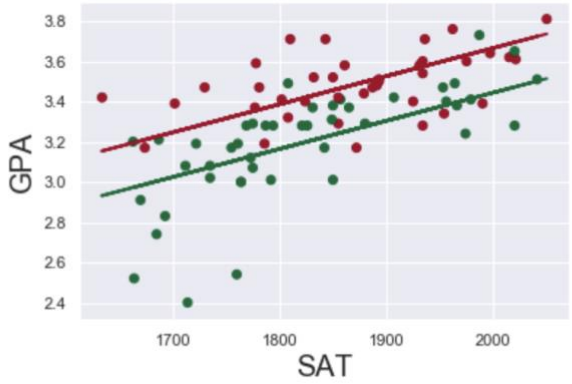
Omnibus:	19.560	**Durbin-Watson:**	1.009
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	27.189
Skew:	-1.028	**Prob(JB):**	1.25e-06
Kurtosis:	4.881	**Cond. No.**	3.35e+04

Visualization:

- Scatter plot with regression lines

We create a scatter plot to visualize the relationship between SAT scores, GPA scores, and attendance. We also plot regression lines separately for attendance values of 0 and 1.

```
In [36]: #models with dummie variables(Attendance)
         plt.scatter(data ['SAT'],y, c=data['Attendance'], cmap ='RdYlGn_r')
         yhat_no = 0.6439 + 0.0014*data['SAT']
         yhat_yes = 0.866 + 0.0014*data['SAT']
         fig = plt.plot(data['SAT'],yhat_no, lw=2, c = '#006837')
         fig = plt.plot(data['SAT'], yhat_yes, lw =2, c = '#a50026')
         plt.xlabel ('SAT', fontsize = 20)
         plt.ylabel('GPA', fontsize = 20)
         plt.show()
```



Making Predictions:

- Create new data for predictions

```
In [41]: #we will predict values based on attendance

In [51]: new_data = pd.DataFrame ({'const':1, 'SAT':[1700,1670], 'Attendance': [0,1]})
         new_data = new_data [['const', 'SAT', 'Attendance']]
         new_data
```

Out[51]:

	const	SAT	Attendance
0	1	1700	0
1	1	1670	1

```
In [52]: new_data.rename(index ={0:'Bob', 1: 'Alice'})
```

Out[52]:

	const	SAT	Attendance
Bob	1	1700	0
Alice	1	1670	1

- Predict GPA scores

new_data['Predicted_GPA'] = results.predict(new_data)

```
In [53]: predictions = results.predict(new_data)
         predictions

Out[53]: 0    3.023513
         1    3.204163
         dtype: float64
```

- Print the predictions

```
In [54]: predictionsdf =pd.DataFrame({'Predictions': predictions})
         joined = new_data.join(predictionsdf)
         joined.rename(index={0:'Bob',1:'Alice'})
```

Out[54]:

	const	SAT	Attendance	Predictions
Bob	1	1700	0	3.023513
Alice	1	1670	1	3.204163

We create new data with two instances, 'Bob' and 'Alice', and their corresponding SAT scores and attendance values. We then use the fitted regression model to predict their GPA scores and display the predictions.

Recommendations

Based on the regression analysis, we can provide the following recommendations:
1. Encourage students to focus on improving their SAT scores, as it positively impacts their GPA.
2. Emphasize the importance of regular attendance to maximize academic performance.
3. Provide support and resources to help students improve their SAT scores and create an environment that promotes regular attendance.

Conclusion

The regression analysis helps predict GPA scores of university students using SAT scores and attendance. By understanding the impact of these factors, educational institutions can devise strategies to improve student outcomes and create a conducive learning environment.

Happy coding ☺
Feel free to connect on linkedIn