

**Assignment 2: Estimating a bicycle exposure model (15 points)**

**Submitted by: Chun-hin (Anson) Sit, Jesus Hinojosa**

**Feb 2025**

In this assignment, you will estimate a exposure model, selecting the appropriate explanatory variables (features) and the model that performs the best, i.e. predicts the counts most accurately. The dependent variable (target) in the data is Annual Pedestrian Traffic at intersections on California state highway system and the explanatory variables (features) consist of infrastructure, demographic, climate, network connectivity, transit, employment/land use and other characteristics. Some of these characteristics are available at different radii of catchment area (buffer) around the intersection, half-mile, quarter-mile, and one-tenth of a mile. This dataset might include variables that would not be useful in the modeling process and miss some that could potentially have been useful.

Please answer the following questions:

1. Based on the columns, please list some initial hypotheses on how each variable might affect pedestrian traffic, that is whether it would have an increasing or decreasing effect, or no influence for increases in that explanatory variable. Please do not revise this after estimating your model. It is always good to note what your initial hypotheses were and how your result might conflict with some of those, since your results are dependent on the variations captured in your data set **(2)**.

**Answer for 1:**

For our hypothesis, we are focused on the variables Annual, Int4way, SchoolsH, Signal, AllTransitH, Jobs30H, and PopT. We believe that Int4way has high pedestrian traffic because it provides pedestrians with more crossing options. SchoolsH was selected because a higher number of schools means more students, which could increase pedestrian traffic. AllTransitH was chosen because greater transit availability may incentivize more people to use it, thereby increasing pedestrian traffic. Jobs30H was included because if jobs are within 30 minutes of transit, pedestrian traffic may rise. PopT was selected because a higher population near the buffer area could lead to increased pedestrian traffic. Overall, we believe these variables will positively influence pedestrian traffic.

- Perform initial data descriptive analysis. Include mean, standard deviations for the dependent variable (target) and the explanatory variables (features). Display correlations among them and plot a few dependent variables with the explanatory variable to observe the functional form. How does your data descriptive analysis and your knowledge of pedestrian traffic influence the variables you select for your model **(3)**?

### **Answer for 2:**

For our analysis, we focused on the variables Annual, Int4way, SchoolsH, Signal, AllTransitH, Jobs30H, and PopT. The Descriptive Analysis and Correlation Heat map show that Jobs30H, AllTransitH, and SchoolsH have a strong or moderate correlation. Jobs30H and PopT being 0.63 and AllTransitH and PopT being 0.62. The moderate correlations are SchoolsH and PopT being 0.47 and Annual and Jobs30H being 0.57. The weakest are Singal, Int4way, and Snow10in have a weak correlation with PopT. This influences the variable we would select by making sure we include Jobs30H, AllTransitH, SchoolsH and exclude Singal, Intway, and Snow10in.

	Annual	Int4way	Snow10in	SchoolsH	Signal	AllTransitH	Jobs30H	PopT
count	1.301000e+03	1301.000000	1301.000000	1301.000000	1301.000000	1301.000000	1.301000e+03	1301.000000
mean	7.328024e+05	0.694081	0.212452	1.882398	0.490392	5.669776	9.730359e+05	237.709650
std	2.029549e+06	0.460973	1.013796	1.874322	0.500100	3.105202	1.733345e+06	259.902302
min	0.000000e+00	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000e+00	0.000000
25%	3.550104e+04	0.000000	0.000000	0.000000	0.000000	3.099828	6.017384e+03	66.095605
50%	1.434238e+05	1.000000	0.000000	1.000000	0.000000	6.498840	1.591286e+05	167.359809
75%	5.436381e+05	1.000000	0.000000	3.000000	1.000000	8.408637	1.064683e+06	334.644143
max	2.579122e+07	1.000000	5.600000	9.000000	1.000000	9.986393	1.050488e+07	3799.904870

Figure 1: The Descriptive Analysis

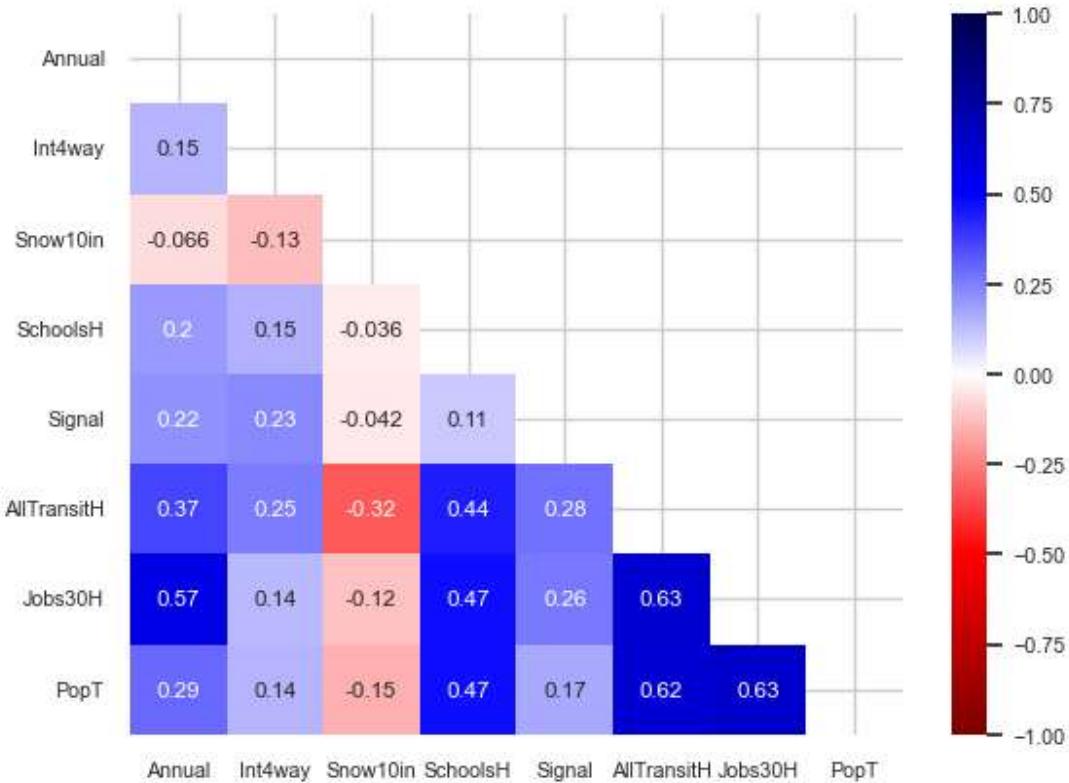


Figure 2: Correlation Heat Map

3. In the modeling stage, please begin with estimating a linear regression model, and then the count models, and any other model that you would like to estimate. For the initial model, you may want to include the explanatory variables one by one to observe the effect that each has on the significance of the other. Remember that the R-squared for LR models would mostly increase as you add more variables, whatever those may be (5).

### **Answer for 3:**

Before we start our analysis, we have divided the dataset into a training set and a test set. 20% of data is allocated for tests.

#### Linear regression model

First, we began with the simplest linear regression models. We have added the 7 selected explanatory variables one by one to observe the change of parameters.

Variables like Int4way, SchoolsH, Signal, Jobs30H, and PopT show significance, though the significance of Int4way decreases when more variables are added. In contrast, Snow10in and AllTransitH are often not significant or only borderline once additional predictors are included.

$R^2$  increases as more predictors are added—from 0.021 with only Int4way to 0.341 in the full model. This is expected as additional variables tend to explain more variation, though the contribution of each variable should be considered in the context of their statistical significance and potential multicollinearity.

The final model (with all 7 variables) shows a relatively high condition number ( $\approx 5.73e+06$ ), which could suggest multicollinearity issues among the predictors. This is likely due to the high correlation between Jobs30H, AllTransitH and PopT.

To further interpret some of the variables:

Int4way (Four-way intersection): Positive and statistically significant. Locations with four-way intersections are associated with an increase in annual pedestrian counts by approximately 292,400, holding other variables constant.

SchoolsH (Number of schools): Negative and statistically significant. An increase in the number of school units is associated with a decrease in annual pedestrian counts by about 118,800 per school unit.

AllTransitH (All transit metric): Not statistically significant. Suggests that transit metrics may not strongly influence pedestrian counts in this model.

### Poisson

Similar to the linear regression models, we have added the 7 selected explanatory variables one by one to observe the change of parameters. Every step reports z-statistics with p-values essentially 0 ( $p < 0.001$ ), indicating that each predictor is statistically significant in its respective model. As new variables are added, some coefficients change in magnitude and even direction.

To further interpret some of the variables:

Int4way (Four-way intersection): Positive. For each additional four-way intersection, the expected annual pedestrian count increases by about 73%.

SchoolsH (Number of schools): Negative. Each additional school unit is associated with about a 13% decrease in annual pedestrian count.

AllTransitH (All transit metric): Positive. For each unit increase in the transit metric, the pedestrian count is expected to be about 40% higher.

As a special note, the pseudo  $R^2$  in all models remains 1, which is weird and should be interpreted with caution.

### Negative Binomial

Again, we have added the 7 selected explanatory variables one by one to observe the change of parameters. In every model almost all predictors have

z-statistics with  $p < 0.001$ . In the full model, however, “SchoolsH” ( $p \approx 0.069$ ) is borderline and “PopT” is only marginally significant ( $p \approx 0.045$ ), suggesting that when all predictors are included, some effects become less clear or may be partially explained by other variables (potential collinearity).

To further interpret some of the variables:

Int4way (Four-way intersection): Positive. A one-unit increase in Int4way is associated with a doubling of the expected pedestrian count (an increase of about 109%).

SchoolsH (Number of schools): Low positive. Each additional school unit is associated with about a 4% increase in the expected count; however, with a p-value of 0.069, this predictor is marginally significant. It might not be a robust predictor at a 5% significance level.

AllTransitH (All transit metric): Positive. A one-unit increase in the transit metric correlates with about a 23% increase in pedestrian count.

4. Please display some of the models you estimated including your final model. What were the reasons for selecting this model as your final model **(2)**?

**Answer for 4:**

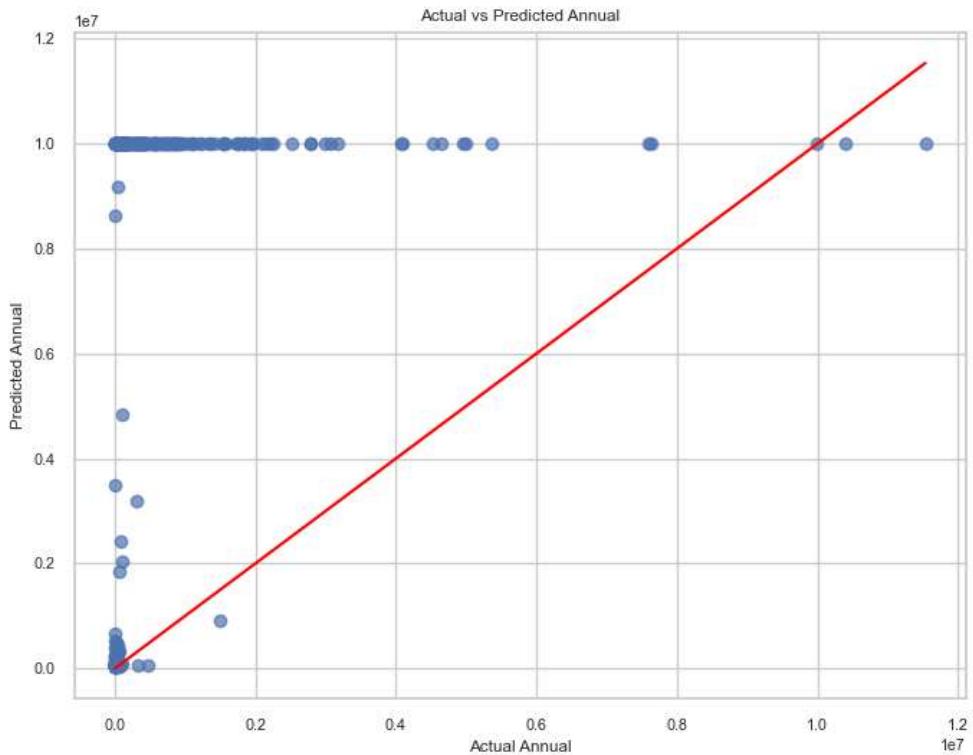
Negative Binomial would be the best model out of the 3 models tested. The annual pedestrian count is inherently a count variable, for which linear regression may perform poorly due to its assumptions of normally distributed errors and constant variance. Also, count data often exhibit variance greater than the mean. The Poisson model assumes the mean equals the variance and, as shown by its enormous deviance and Pearson chi<sup>2</sup> values, it fails to capture the observed variability in the data.

5. Please discuss any limitations in your model and how you could correct those. Discuss issues such as endogeneity, simultaneity etc. **(3)**.

**Answer for 5:**

To evaluate the performance of our Negative Binomial model using the selected predictors, we compared its estimated values with the actual test data. The results indicate that the model is performing very poorly. Key metrics are as follows:

MSE: 65909906125183.47  
RMSE: 8118491.6163769895  
MAE: 7104500.208496759  
R<sup>2</sup>: -26.5163803448109



Several potential issues could be contributing to this poor performance:

Endogeneity occurs when one or more predictors are correlated with the error term—often due to measurement error, omitted variables, or reverse causality. For instance, if the transit metric "AllTransitH" and pedestrian counts are determined simultaneously in certain urban settings, the model struggles to distinguish cause from effect.

Simultaneity arises when the dependent variable and one or more predictors are jointly determined. For example, areas with high pedestrian counts might attract better transit services, creating a two-way influence. This bidirectional relationship can lead to biased and inconsistent coefficient estimates.

Selecting highly correlated predictors may contribute to the model's poor performance. Overfitting or underfitting, along with improper scaling of variables, can also destabilize the model and lead to inaccurate predictions.

If given the opportunity to redo the model, it would be advisable to address these issues—by, for example, incorporating more uncorrelated variables, applying appropriate scaling or transformations, and using techniques to mitigate endogeneity (such as instrumental variables or structural modeling).

Please submit your responses in the form of a short report/write-up including your code. You could do this assignment on your own or team up with **one** other student.

## Appendix:

### 1) Hypothesis:

#### Demographics:

- Pop: A higher population may lead to an increase in pedestrian traffic because more people are living near the buffer area.
- HseHold: An increase in households may increase pedestrian activity.
- White: This could have an impact due to socioeconomic factors at play.
- TransCom: More transit commuters could mean an increase in pedestrian traffic at transit hubs.
- WalkCom: The more people walking could increase pedestrian traffic.
- Degree: Higher education levels may correlate to the ability to afford a car increased cars on the road
- NoVech: Households with no car are more likely to use transit or walk, this could increase the pedestrian traffic at hubs.
- Pop25: Older population may walk less which could decrease pedestrian traffic
- WhitePct: This could impact pedestrian traffic because white people tend to be hit less when walking, possibly increasing the number of pedestrians if the white traffic is high.
- TransComPct: The higher the number of shared modes like buses or trains could increase pedestrian traffic.
- WalkComPct: The higher the walk mode share this would likely increase pedestrian traffic.
- DegreePct: High education levels could increase the car ridership lessening pedestrian traffic.

#### Infrastructure:

- PrincArt: The principal arterials could have higher car traffic which would affect pedestrian traffic.
- MinAct: Minor arterial could have light traffic with minor effects on pedestrian traffic
- Collector: Collector streets could have fewer cars which would increase traffic for pedestrians.
- Int4way: This could have high pedestrian traffic because it gives pedestrians more options to cross.
- Singal: Intersections with signals could increase pedestrian traffic because pedestrians feel safer when using them.
- FC\_min, FC\_mac\_, FC\_sum: This could decrease pedestrian traffic because more cars are on the road.

#### Network Connectivity:

- StMetters: The increase in meters could increase traffic for pedestrians because they have more routes.
- StgSeg: More Street Segments could increase pedestrian traffic because the route is better connected

#### Transit (census Tract Level):

- AllTransit: With more transit, it could incentivize people to use it more increasing pedestrian traffic.
- Jobs30: If jobs are within 30 mins on transit there could be an increase in pedestrian traffic.
- Commuters: Having a higher number of transit commuters could increase pedestrian traffic at the transit hub.
- Tripswk: More transit trips per week could increase the pedestrian traffic at the transit hub.
- Routes: The more routes the transit has could increase pedestrian traffic.

#### Employment/Land Use:

- EmpSF: The more commercial spaces add new destinations for pedestrians increasing the traffic for them as well.
- Emp: The more employees in the area the higher chance of pedestrian traffic around the area.

#### Climate

- Precip05in: Maybe not enough rain to stop pedestrian traffic but enough to lessen it.
- Precip1in: Enough rain to decrease pedestrian traffic and maybe even stop it.
- Snow10in: Enough snow to decrease pedestrian traffic.
- Temp90: Enough to lower pedestrian traffic

#### Other:

- DistWater: Being close to a body of water could increase pedestrian traffic, think of beaches
- Schools: A higher number of schools means a higher number of students which could increase pedestrian traffic.
- MaxSlope: It could decrease traffic for pedestrians because it difficult for some

## 2) Code:

## 265Assng2\_achsit

February 19, 2025

```
[4]: ### Import libraries
import pandas as pd
import numpy as np
import statsmodels.api as sm
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import seaborn as sns
```

```
[5]: ### Create dataframe from csv file
exposure = pd.read_csv('Data for model.csv')
exposure.head()
```

```
[5]:    year      Annual ClimateZone Precip05in Precip1in Snow10in Temp90 \
0  2014  1.008715e+05            3       160        55     0.0   4.0
1  2014  1.160158e+06            3       160        55     0.0   4.0
2  2014  5.230030e+05            3       160        55     0.0   4.0
3  2014  5.453961e+05            3       160        55     0.0   4.0
4  2014  1.303617e+05            3       160        55     0.0   4.0

      FC_Min  FC_Max PrincArt ... HseHldH WalkComH TransComH \
0        4      7       1 ... 3442.366473  166.738041  1018.284763
1        3      3       2 ... 3162.386495  158.532153   773.437362
2        3      7       1 ... 4858.992498  916.155792  1430.823715
3        3      7       1 ... 2755.337596  111.032411   383.858091
4        3      4       1 ... 1338.560970   22.904996    44.223395

      DegreeH NoVehH WalkComPctH TransComPctH NonWhitePctH \
0 1099.640883  947.562614     0.044633     0.272576     0.226225
1 1590.916415  508.269792     0.036902     0.180034     0.326323
2 2858.854956  805.819303     0.155156     0.242317     0.723288
3  721.283123  147.926427     0.026330     0.091027     0.348611
4  574.407818   65.102869     0.017433     0.033658     0.334453

      DegreePctH NoVehPctH
0      0.193205  0.275265
1      0.264586  0.160723
2      0.373215  0.165841
3      0.119821  0.053687
```

```
4      0.208903   0.048636
```

```
[5 rows x 80 columns]
```

```
[6]: ### Select variables & check mean, std dev., min, & max value.  
df = exposure[['Annual', 'Int4way', 'Snow10in', 'SchoolsH', 'Signal',  
               'AllTransitH', 'Jobs30H', 'PopT']]  
df.describe()
```

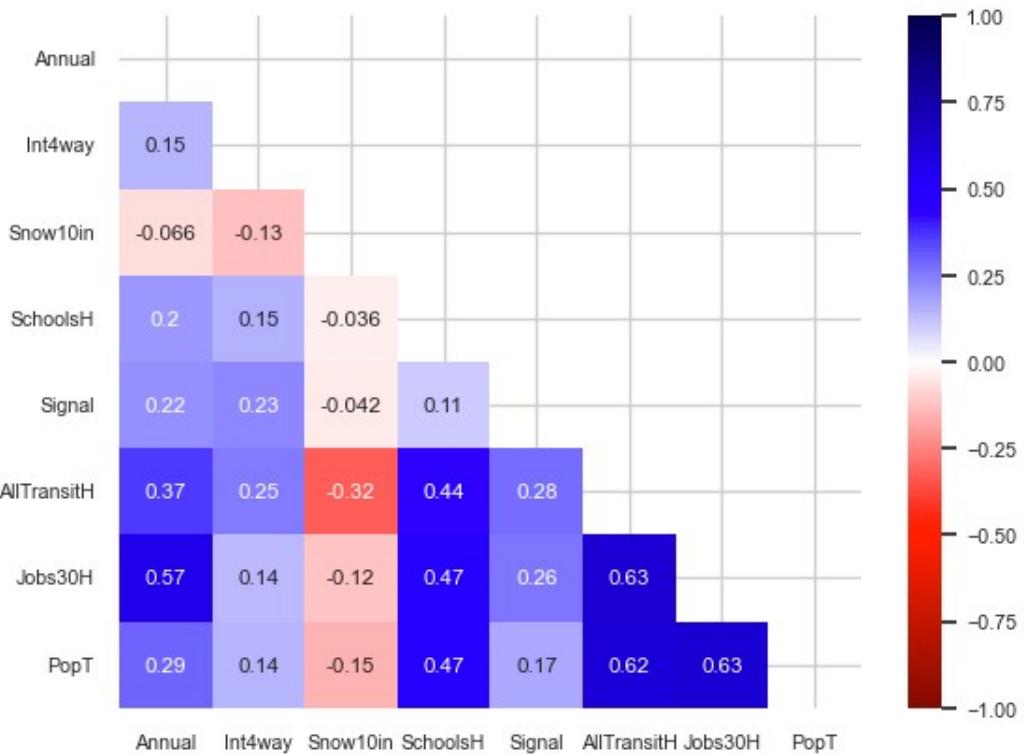
```
[6]:
```

	Annual	Int4way	Snow10in	SchoolsH	Signal	\
count	1.301000e+03	1301.000000	1301.000000	1301.000000	1301.000000	
mean	7.328024e+05	0.694081	0.212452	1.882398	0.490392	
std	2.029549e+06	0.460973	1.013796	1.874322	0.500100	
min	0.000000e+00	0.000000	0.000000	0.000000	0.000000	
25%	3.550104e+04	0.000000	0.000000	0.000000	0.000000	
50%	1.434238e+05	1.000000	0.000000	1.000000	0.000000	
75%	5.436381e+05	1.000000	0.000000	3.000000	1.000000	
max	2.579122e+07	1.000000	5.600000	9.000000	1.000000	

	AllTransitH	Jobs30H	PopT
count	1301.000000	1.301000e+03	1301.000000
mean	5.669776	9.730359e+05	237.709650
std	3.105202	1.733345e+06	259.902302
min	0.000000	0.000000e+00	0.000000
25%	3.099828	6.017384e+03	66.095605
50%	6.498840	1.591286e+05	167.359809
75%	8.408637	1.064683e+06	334.644143
max	9.986393	1.050488e+07	3799.904870

```
[7]: #Create heatmap of correlations among the variables  
sns.set(context='notebook', style='whitegrid', font_scale=0.7)  
upper = np.triu(df.corr()) # Here, we are looking at the upper triangle.  
# Optionally, you can just look at the lower triangle.  
sns.heatmap(df.corr(), cmap="seismic_r", annot=True, vmin=-1, vmax=1,  
            mask=upper);  
plt.savefig('heatmap.png')
```



```
[8]: #Splitting into training & test data
X_train, X_test, y_train, y_test = train_test_split(df[['Int4way', 'Snow10in',
    ↪'SchoolsH', 'Signal', 'AllTransitH', 'Jobs30H', 'PopT']], df['Annual'],
    ↪test_size=0.2, random_state=20)
```

```
[9]: ### Fit a linear regression model using statsmodels
```

```
import statsmodels.api as sm

# List of all explanatory variables in the order you'd like to add them:
predictors = ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH',
    ↪'PopT', 'Jobs30H']

# Fit models adding one predictor at a time
for i in range(1, len(predictors) + 1):
    current_vars = predictors[:i]
    X_curr = sm.add_constant(X_train[current_vars])
    model = sm.OLS(y_train, X_curr).fit()
    print(f"\nModel with predictors: {current_vars}")
    print(model.summary())
```

Model with predictors: ['Int4way']

OLS Regression Results

Dep. Variable:	Annual	R-squared:	0.021
Model:	OLS	Adj. R-squared:	0.020
Method:	Least Squares	F-statistic:	22.43
Date:	Wed, 19 Feb 2025	Prob (F-statistic):	2.48e-06
Time:	17:03:27	Log-Likelihood:	-16620.
No. Observations:	1040	AIC:	3.324e+04
Df Residuals:	1038	BIC:	3.325e+04
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
const	2.754e+05	1.17e+05	2.347	0.019	4.52e+04	5.06e+05
Int4way	6.697e+05	1.41e+05	4.736	0.000	3.92e+05	9.47e+05
Omnibus:	1314.896	Durbin-Watson:			2.041	
Prob(Omnibus):	0.000	Jarque-Bera (JB):			144842.697	
Skew:	6.715	Prob(JB):			0.00	
Kurtosis:	59.233	Cond. No.			3.35	

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model with predictors: ['Int4way', 'Snow10in']

OLS Regression Results

Dep. Variable:	Annual	R-squared:	0.023
Model:	OLS	Adj. R-squared:	0.021
Method:	Least Squares	F-statistic:	12.37
Date:	Wed, 19 Feb 2025	Prob (F-statistic):	4.93e-06
Time:	17:03:27	Log-Likelihood:	-16619.
No. Observations:	1040	AIC:	3.324e+04
Df Residuals:	1037	BIC:	3.326e+04
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P> t	[0.025	0.975]
const	3.172e+05	1.2e+05	2.633	0.009	8.08e+04	5.54e+05
Int4way	6.406e+05	1.43e+05	4.492	0.000	3.61e+05	9.2e+05
Snow10in	-9.472e+04	6.28e+04	-1.508	0.132	-2.18e+05	2.85e+04

Omnibus: 1315.226 Durbin-Watson: 2.037

Prob(Omnibus):	0.000	Jarque-Bera (JB):	145197.158
Skew:	6.717	Prob(JB):	0.00
Kurtosis:	59.305	Cond. No.	3.53

---

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH']  
OLS Regression Results

---

Dep. Variable:	Annual	R-squared:	0.046
Model:	OLS	Adj. R-squared:	0.043
Method:	Least Squares	F-statistic:	16.55
Date:	Wed, 19 Feb 2025	Prob (F-statistic):	1.65e-10
Time:	17:03:27	Log-Likelihood:	-16607.
No. Observations:	1040	AIC:	3.322e+04
Df Residuals:	1036	BIC:	3.324e+04
Df Model:	3		
Covariance Type:	nonrobust		

---

	coef	std err	t	P> t	[0.025	0.975]
const	7.182e+04	1.29e+05	0.556	0.578	-1.82e+05	3.25e+05
Int4way	5.299e+05	1.43e+05	3.710	0.000	2.5e+05	8.1e+05
Snow10in	-8.823e+04	6.21e+04	-1.420	0.156	-2.1e+05	3.37e+04
SchoolsH	1.693e+05	3.43e+04	4.935	0.000	1.02e+05	2.37e+05

---

Omnibus:	1343.359	Durbin-Watson:	2.043
Prob(Omnibus):	0.000	Jarque-Bera (JB):	164983.631
Skew:	6.938	Prob(JB):	0.00
Kurtosis:	63.123	Cond. No.	7.84

---

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal']  
OLS Regression Results

---

Dep. Variable:	Annual	R-squared:	0.075
Model:	OLS	Adj. R-squared:	0.072
Method:	Least Squares	F-statistic:	21.07
Date:	Wed, 19 Feb 2025	Prob (F-statistic):	1.02e-16
Time:	17:03:27	Log-Likelihood:	-16591.
No. Observations:	1040	AIC:	3.319e+04

Df Residuals:	1035	BIC:	3.322e+04			
Df Model:	4					
Covariance Type:	nonrobust					
<hr/>						
	coef	std err	t	P> t	[0.025	0.975]
const	-1.374e+05	1.32e+05	-1.039	0.299	-3.97e+05	1.22e+05
Int4way	3.441e+05	1.44e+05	2.384	0.017	6.09e+04	6.27e+05
Snow10in	-9.021e+04	6.12e+04	-1.474	0.141	-2.1e+05	2.99e+04
SchoolsH	1.561e+05	3.39e+04	4.609	0.000	8.96e+04	2.23e+05
Signal	7.574e+05	1.32e+05	5.752	0.000	4.99e+05	1.02e+06
<hr/>						
Omnibus:	1338.389	Durbin-Watson:	2.057			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	165260.557			
Skew:	6.886	Prob(JB):	0.00			
Kurtosis:	63.200	Cond. No.	7.92			
<hr/>						

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH']

#### OLS Regression Results

---

Dep. Variable:	Annual	R-squared:	0.146			
Model:	OLS	Adj. R-squared:	0.142			
Method:	Least Squares	F-statistic:	35.43			
Date:	Wed, 19 Feb 2025	Prob (F-statistic):	1.59e-33			
Time:	17:03:27	Log-Likelihood:	-16549.			
No. Observations:	1040	AIC:	3.311e+04			
Df Residuals:	1034	BIC:	3.314e+04			
Df Model:	5					
Covariance Type:	nonrobust					
<hr/>						
	coef	std err	t	P> t	[0.025	0.975]
const	-9.189e+05	1.53e+05	-6.024	0.000	-1.22e+06	-6.2e+05
Int4way	1.703e+05	1.4e+05	1.217	0.224	-1.04e+05	4.45e+05
Snow10in	1.054e+05	6.25e+04	1.686	0.092	-1.73e+04	2.28e+05
SchoolsH	1.257e+04	3.61e+04	0.349	0.727	-5.82e+04	8.33e+04
Signal	4.818e+05	1.3e+05	3.705	0.000	2.27e+05	7.37e+05
AllTransitH	2.256e+05	2.43e+04	9.270	0.000	1.78e+05	2.73e+05
<hr/>						
Omnibus:	1320.390	Durbin-Watson:	2.024			
Prob(Omnibus):	0.000	Jarque-Bera (JB):	159240.947			
Skew:	6.721	Prob(JB):	0.00			

Kurtosis: 62.111 Cond. No. 19.7

---

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', 'PopT']

OLS Regression Results

---

Dep. Variable:	Annual	R-squared:	0.151
Model:	OLS	Adj. R-squared:	0.146
Method:	Least Squares	F-statistic:	30.58
Date:	Wed, 19 Feb 2025	Prob (F-statistic):	6.67e-34
Time:	17:03:27	Log-Likelihood:	-16546.
No. Observations:	1040	AIC:	3.311e+04
Df Residuals:	1033	BIC:	3.314e+04
Df Model:	6		
Covariance Type:	nonrobust		

---

	coef	std err	t	P> t	[0.025	0.975]
const	-8.702e+05	1.54e+05	-5.665	0.000	-1.17e+06	-5.69e+05
Int4way	1.846e+05	1.4e+05	1.320	0.187	-8.98e+04	4.59e+05
Snow10in	1.007e+05	6.24e+04	1.614	0.107	-2.17e+04	2.23e+05
SchoolsH	-1.414e+04	3.77e+04	-0.375	0.708	-8.82e+04	5.99e+04
Signal	4.802e+05	1.3e+05	3.701	0.000	2.26e+05	7.35e+05
AllTransitH	1.894e+05	2.87e+04	6.592	0.000	1.33e+05	2.46e+05
PopT	848.7418	360.276	2.356	0.019	141.785	1555.699

---

Omnibus:	1330.467	Durbin-Watson:	2.027
Prob(Omnibus):	0.000	Jarque-Bera (JB):	165807.784
Skew:	6.802	Prob(JB):	0.00
Kurtosis:	63.343	Cond. No.	952.

---

Notes:

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', 'PopT', 'Jobs30H']

OLS Regression Results

---

Dep. Variable:	Annual	R-squared:	0.341
Model:	OLS	Adj. R-squared:	0.336
Method:	Least Squares	F-statistic:	76.25

---

```

Date: Wed, 19 Feb 2025 Prob (F-statistic): 4.78e-89
Time: 17:03:27 Log-Likelihood: -16415.
No. Observations: 1040 AIC: 3.285e+04
Df Residuals: 1032 BIC: 3.288e+04
Df Model: 7
Covariance Type: nonrobust
=====

              coef    std err      t    P>|t|    [0.025    0.975]
-----
const      -9.771e+04  1.43e+05  -0.685   0.493  -3.78e+05  1.82e+05
Int4way     2.924e+05  1.23e+05   2.369   0.018   5.02e+04  5.35e+05
Snow10in    3.436e+04  5.51e+04   0.623   0.533  -7.38e+04  1.43e+05
SchoolsH    -1.188e+05  3.38e+04  -3.517   0.000  -1.85e+05  -5.25e+04
Signal      2.265e+05  1.15e+05   1.965   0.050   287.366  4.53e+05
AllTransitH 4.018e+04  2.68e+04   1.501   0.134  -1.23e+04  9.27e+04
PopT        -922.8168  333.747   -2.765   0.006  -1577.717  -267.917
Jobs30H      0.7747    0.045    17.252   0.000   0.687   0.863
=====
Omnibus:          1252.630 Durbin-Watson:       1.974
Prob(Omnibus):    0.000 Jarque-Bera (JB): 152403.939
Skew:             6.074 Prob(JB):            0.00
Kurtosis:         61.047 Cond. No.        5.73e+06
=====

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 5.73e+06. This might indicate that there are strong multicollinearity or other numerical problems.

```

[10]: *Fit a Poisson regression model using statsmodels*

```

import statsmodels.api as sm

# List of explanatory variables in the order you'd like to add them:
predictors = ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', 'PopT', 'Jobs30H']

# Iteratively build Poisson regression models
for i in range(1, len(predictors) + 1):
    current_vars = predictors[:i]
    X_curr = sm.add_constant(X_train[current_vars])
    poisson_model = sm.GLM(y_train, X_curr, family=sm.families.Poisson()).fit()
    print(f"\nPoisson Model with predictors: {current_vars}")
    print(poisson_model.summary())

```

Poisson Model with predictors: ['Int4way']

Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1038
Model Family:	Poisson	Df Model:	1
Link Function:	Log	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-1.0230e+09
Date:	Wed, 19 Feb 2025	Deviance:	2.0459e+09
Time:	17:03:27	Pearson chi2:	5.88e+09
No. Iterations:	9	Pseudo R-squ. (CS):	1.000
Covariance Type:	nonrobust		

	coef	std err	z	P> z	[0.025	0.975]
const	12.5260	0.000	1.18e+05	0.000	12.526	12.526
Int4way	1.2331	0.000	1.09e+04	0.000	1.233	1.233

Poisson Model with predictors: ['Int4way', 'Snow10in']

Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1037
Model Family:	Poisson	Df Model:	2
Link Function:	Log	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-1.0069e+09
Date:	Wed, 19 Feb 2025	Deviance:	2.0138e+09
Time:	17:03:27	Pearson chi2:	5.67e+09
No. Iterations:	7	Pseudo R-squ. (CS):	1.000
Covariance Type:	nonrobust		

	coef	std err	z	P> z	[0.025	0.975]
const	12.6099	0.000	1.19e+05	0.000	12.610	12.610
Int4way	1.1784	0.000	1.05e+04	0.000	1.178	1.179
Snow10in	-0.4851	0.000	-3432.865	0.000	-0.485	-0.485

Poisson Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH']

Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1036
Model Family:	Poisson	Df Model:	3
Link Function:	Log	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-9.4573e+08
Date:	Wed, 19 Feb 2025	Deviance:	1.8915e+09
Time:	17:03:27	Pearson chi2:	5.81e+09

No. Iterations:	7	Pseudo R-squ. (CS):	1.000			
Covariance Type:	nonrobust					
=====						
	coef	std err	z	P> z	[0.025	0.975]
=====						
const	12.2790	0.000	1.1e+05	0.000	12.279	12.279
Int4way	1.0531	0.000	9303.296	0.000	1.053	1.053
Snow10in	-0.4814	0.000	-3385.736	0.000	-0.482	-0.481
SchoolsH	0.1833	1.56e-05	1.17e+04	0.000	0.183	0.183
=====						

Poisson Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal']  
Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040			
Model:	GLM	Df Residuals:	1035			
Model Family:	Poisson	Df Model:	4			
Link Function:	Log	Scale:	1.0000			
Method:	IRLS	Log-Likelihood:	-8.4457e+08			
Date:	Wed, 19 Feb 2025	Deviance:	1.6891e+09			
Time:	17:03:27	Pearson chi2:	4.57e+09			
No. Iterations:	8	Pseudo R-squ. (CS):	1.000			
Covariance Type:	nonrobust					
=====						
	coef	std err	z	P> z	[0.025	0.975]
=====						
const	11.7815	0.000	9.54e+04	0.000	11.781	11.782
Int4way	0.7907	0.000	6908.594	0.000	0.790	0.791
Snow10in	-0.4831	0.000	-3355.068	0.000	-0.483	-0.483
SchoolsH	0.1663	1.57e-05	1.06e+04	0.000	0.166	0.166
Signal	1.1608	8.91e-05	1.3e+04	0.000	1.161	1.161
=====						

Poisson Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH']

Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040			
Model:	GLM	Df Residuals:	1034			
Model Family:	Poisson	Df Model:	5			
Link Function:	Log	Scale:	1.0000			
Method:	IRLS	Log-Likelihood:	-4.8185e+08			
Date:	Wed, 19 Feb 2025	Deviance:	9.6368e+08			
Time:	17:03:27	Pearson chi2:	1.69e+09			
No. Iterations:	10	Pseudo R-squ. (CS):	1.000			
Covariance Type:	nonrobust					
=====						
	coef	std err	z	P> z	[0.025	0.975]
=====						

const	8.0254	0.000	3.06e+04	0.000	8.025	8.026
Int4way	0.4822	0.000	4220.666	0.000	0.482	0.482
Snow10in	0.4411	0.000	4401.196	0.000	0.441	0.441
SchoolsH	-0.0622	1.85e-05	-3360.128	0.000	-0.062	-0.062
Signal	0.6083	9.12e-05	6666.429	0.000	0.608	0.608
AllTransitH	0.6295	3.06e-05	2.06e+04	0.000	0.629	0.630

Poisson Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', 'PopT']

#### Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1033
Model Family:	Poisson	Df Model:	6
Link Function:	Log	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-4.7869e+08
Date:	Wed, 19 Feb 2025	Deviance:	9.5736e+08
Time:	17:03:27	Pearson chi2:	1.68e+09
No. Iterations:	9	Pseudo R-squ. (CS):	1.000
Covariance Type:	nonrobust		

	coef	std err	z	P> z	[0.025	0.975]
const	8.0438	0.000	3.05e+04	0.000	8.043	8.044
Int4way	0.5192	0.000	4506.366	0.000	0.519	0.519
Snow10in	0.4399	0.000	4386.472	0.000	0.440	0.440
SchoolsH	-0.0757	1.94e-05	-3903.966	0.000	-0.076	-0.076
Signal	0.5991	9.14e-05	6552.787	0.000	0.599	0.599
AllTransitH	0.6137	3.13e-05	1.96e+04	0.000	0.614	0.614
PopT	0.0003	1.25e-07	2589.512	0.000	0.000	0.000

Poisson Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', 'PopT', 'Jobs30H']

#### Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1032
Model Family:	Poisson	Df Model:	7
Link Function:	Log	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-3.9097e+08
Date:	Wed, 19 Feb 2025	Deviance:	7.8192e+08
Time:	17:03:27	Pearson chi2:	1.14e+09
No. Iterations:	8	Pseudo R-squ. (CS):	1.000
Covariance Type:	nonrobust		

	coef	std err	z	P> z	[0.025	0.975]
const	9.9901	0.000	4.11e+04	0.000	9.990	9.991
Int4way	0.5506	0.000	4717.431	0.000	0.550	0.551
Snow10in	0.1675	0.000	1656.381	0.000	0.167	0.168
SchoolsH	-0.1360	2.12e-05	-6421.272	0.000	-0.136	-0.136
Signal	0.4607	9.29e-05	4960.822	0.000	0.461	0.461
AllTransitH	0.3369	3.11e-05	1.08e+04	0.000	0.337	0.337
PopT	-0.0002	1.17e-07	-1337.542	0.000	-0.000	-0.000
Jobs30H	2.897e-07	2.16e-11	1.34e+04	0.000	2.9e-07	2.9e-07

[11]: *### Fit a Negative Binomial regression model using statsmodels*

```
import statsmodels.api as sm

# List of explanatory variables in the order you'd like to add them
predictors = ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', u
    ↪'PopT', 'Jobs30H']

# Iteratively build Negative Binomial models
for i in range(1, len(predictors) + 1):
    current_vars = predictors[:i]
    X_curr = sm.add_constant(X_train[current_vars])
    nb_model = sm.GLM(y_train, X_curr, family=sm.families.
    ↪NegativeBinomial(alpha=1)).fit()
    print(f"\nNegative Binomial Model with predictors: {current_vars}")
    print(nb_model.summary())
```

Negative Binomial Model with predictors: ['Int4way']									
Generalized Linear Model Regression Results									
Dep. Variable:	Annual	No. Observations:	1040						
Model:	GLM	Df Residuals:	1038						
Model Family:	NegativeBinomial	Df Model:	1						
Link Function:	Log	Scale:	1.0000						
Method:	IRLS	Log-Likelihood:	-14950.						
Date:	Wed, 19 Feb 2025	Deviance:	3844.4						
Time:	17:03:27	Pearson chi2:	9.81e+03						
No. Iterations:	7	Pseudo R-squ. (CS):	0.2362						
Covariance Type:	nonrobust								
	coef	std err	z	P> z	[0.025	0.975]			
const	12.5260	0.056	225.467	0.000	12.417	12.635			
Int4way	1.2331	0.067	18.416	0.000	1.102	1.364			

Negative Binomial Model with predictors: ['Int4way', 'Snow10in']  
 Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1037
Model Family:	NegativeBinomial	Df Model:	2
Link Function:	Log	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-14920.
Date:	Wed, 19 Feb 2025	Deviance:	3785.1
Time:	17:03:27	Pearson chi2:	9.08e+03
No. Iterations:	8	Pseudo R-squ. (CS):	0.2785
Covariance Type:	nonrobust		

	coef	std err	z	P> z	[0.025	0.975]
const	12.6149	0.057	220.982	0.000	12.503	12.727
Int4way	1.1526	0.068	17.056	0.000	1.020	1.285
Snow10in	-0.2700	0.030	-9.070	0.000	-0.328	-0.212

Negative Binomial Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH']  
 Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1036
Model Family:	NegativeBinomial	Df Model:	3
Link Function:	Log	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-14792.
Date:	Wed, 19 Feb 2025	Deviance:	3529.3
Time:	17:03:27	Pearson chi2:	1.05e+04
No. Iterations:	12	Pseudo R-squ. (CS):	0.4358
Covariance Type:	nonrobust		

	coef	std err	z	P> z	[0.025	0.975]
const	11.9479	0.062	193.158	0.000	11.827	12.069
Int4way	1.2412	0.068	18.139	0.000	1.107	1.375
Snow10in	-0.2473	0.030	-8.306	0.000	-0.306	-0.189
SchoolsH	0.2528	0.016	15.377	0.000	0.221	0.285

Negative Binomial Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal']

Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1035

Model Family:	NegativeBinomial	Df Model:	4
Link Function:	Log	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-14584.
Date:	Wed, 19 Feb 2025	Deviance:	3112.0
Time:	17:03:27	Pearson chi2:	7.97e+03
No. Iterations:	14	Pseudo R-squ. (CS):	0.6223
Covariance Type:	nonrobust		

	coef	std err	z	P> z	[0.025	0.975]
const	11.1935	0.064	173.981	0.000	11.067	11.320
Int4way	1.0236	0.070	14.580	0.000	0.886	1.161
Snow10in	-0.2014	0.030	-6.766	0.000	-0.260	-0.143
SchoolsH	0.2828	0.016	17.165	0.000	0.251	0.315
Signal	1.3286	0.064	20.743	0.000	1.203	1.454

Negative Binomial Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH']

#### Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1034
Model Family:	NegativeBinomial	Df Model:	5
Link Function:	Log	Scale:	1.0000
Method:	IRLS	Log-Likelihood:	-14172.
Date:	Wed, 19 Feb 2025	Deviance:	2288.1
Time:	17:03:27	Pearson chi2:	2.96e+03
No. Iterations:	21	Pseudo R-squ. (CS):	0.8290
Covariance Type:	nonrobust		

	coef	std err	z	P> z	[0.025	0.975]
const	9.7753	0.077	126.631	0.000	9.624	9.927
Int4way	0.6805	0.071	9.605	0.000	0.542	0.819
Snow10in	0.1434	0.032	4.532	0.000	0.081	0.205
SchoolsH	0.0824	0.018	4.515	0.000	0.047	0.118
Signal	0.6310	0.066	9.591	0.000	0.502	0.760
AllTransitH	0.3389	0.012	27.522	0.000	0.315	0.363

Negative Binomial Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', 'PopT']

#### Generalized Linear Model Regression Results

Dep. Variable:	Annual	No. Observations:	1040
Model:	GLM	Df Residuals:	1033
Model Family:	NegativeBinomial	Df Model:	6

Link Function: Log Scale: 1.0000  
 Method: IRLS Log-Likelihood: -14163.  
 Date: Wed, 19 Feb 2025 Deviance: 2269.9  
 Time: 17:03:27 Pearson chi2: 3.07e+03  
 No. Iterations: 22 Pseudo R-squ. (CS): 0.8319  
 Covariance Type: nonrobust

---

	coef	std err	z	P> z	[0.025	0.975]
const	9.7852	0.078	125.607	0.000	9.633	9.938
Int4way	0.7318	0.071	10.320	0.000	0.593	0.871
Snow10in	0.1466	0.032	4.633	0.000	0.085	0.209
SchoolsH	0.0603	0.019	3.152	0.002	0.023	0.098
Signal	0.6123	0.066	9.306	0.000	0.483	0.741
AllTransitH	0.3115	0.015	21.374	0.000	0.283	0.340
PopT	0.0006	0.000	3.542	0.000	0.000	0.001

---

Negative Binomial Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', 'PopT', 'Jobs30H']

#### Generalized Linear Model Regression Results

---

Dep. Variable: Annual No. Observations: 1040  
 Model: GLM Df Residuals: 1032  
 Model Family: NegativeBinomial Df Model: 7  
 Link Function: Log Scale: 1.0000  
 Method: IRLS Log-Likelihood: -14072.  
 Date: Wed, 19 Feb 2025 Deviance: 2087.9  
 Time: 17:03:27 Pearson chi2: 2.28e+03  
 No. Iterations: 13 Pseudo R-squ. (CS): 0.8589  
 Covariance Type: nonrobust

---

	coef	std err	z	P> z	[0.025	0.975]
const	10.1444	0.082	123.633	0.000	9.984	10.305
Int4way	0.7381	0.071	10.395	0.000	0.599	0.877
Snow10in	0.1114	0.032	3.511	0.000	0.049	0.174
SchoolsH	0.0353	0.019	1.816	0.069	-0.003	0.073
Signal	0.5260	0.066	7.929	0.000	0.396	0.656
AllTransitH	0.2055	0.015	13.348	0.000	0.175	0.236
PopT	0.0004	0.000	2.002	0.045	8.16e-06	0.001
Jobs30H	3.158e-07	2.58e-08	12.221	0.000	2.65e-07	3.66e-07

---

Negative Binomial Model with predictors: ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', 'PopT', 'Jobs30H']

#### Generalized Linear Model Regression Results

---

Dep. Variable:	Annual	No. Observations:	1040			
Model:	GLM	Df Residuals:	1032			
Model Family:	NegativeBinomial	Df Model:	7			
Link Function:	Log	Scale:	1.0000			
Method:	IRLS	Log-Likelihood:	-14072.			
Date:	Wed, 19 Feb 2025	Deviance:	2087.9			
Time:	17:03:27	Pearson chi2:	2.28e+03			
No. Iterations:	13	Pseudo R-squ. (CS):	0.8589			
Covariance Type:	nonrobust					
<hr/>						
	coef	std err	z	P> z	[0.025	0.975]
const	10.1444	0.082	123.633	0.000	9.984	10.305
Int4way	0.7381	0.071	10.395	0.000	0.599	0.877
Snow10in	0.1114	0.032	3.511	0.000	0.049	0.174
SchoolsH	0.0353	0.019	1.816	0.069	-0.003	0.073
Signal	0.5260	0.066	7.929	0.000	0.396	0.656
AllTransitH	0.2055	0.015	13.348	0.000	0.175	0.236
PopT	0.0004	0.000	2.002	0.045	8.16e-06	0.001
Jobs30H	3.158e-07	2.58e-08	12.221	0.000	2.65e-07	3.66e-07
<hr/>						

[12]: *### Evaluate the Negative Binomial model on test data*

```

import numpy as np
import matplotlib.pyplot as plt
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
import statsmodels.api as sm
from sklearn.model_selection import train_test_split

# Splitting data
predictors = ['Int4way', 'Snow10in', 'SchoolsH', 'Signal', 'AllTransitH', ↴
    ↴ 'Jobs30H', 'PopT']
X_train, X_test, y_train, y_test = train_test_split(
    df[predictors], df['Annual'], test_size=0.2, random_state=20
)

# Note: nb_model should be your final Negative Binomial model trained using ↴
# X_train and y_train.
# Prepare test set by adding a constant to X_test (using the same predictors)
X_test_const = sm.add_constant(X_test)
y_pred = nb_model.predict(X_test_const)

# Check for non-finite values in predictions
if not np.all(np.isfinite(y_pred)):
    print("Non-finite values found in y_pred!")
    print("max:", np.max(y_pred[np.isfinite(y_pred)]))

```

```

print("min:", np.min(y_pred[np.isfinite(y_pred)]))

# Clip predictions to avoid extreme values (example limits, adjust as necessary)
y_pred_clipped = np.clip(y_pred, a_min=None, a_max=1e7) # set a realistic  

maximum

# Then compute performance metrics on clipped predictions
mse = mean_squared_error(y_test, y_pred_clipped)
rmse = np.sqrt(mse)
mae = mean_absolute_error(y_test, y_pred_clipped)
r2 = r2_score(y_test, y_pred_clipped)

print(f"MSE: {mse}")
print(f"RMSE: {rmse}")
print(f"MAE: {mae}")
print(f"R^2: {r2}")

# Plot actual vs predicted values
plt.figure(figsize=(8, 6))
plt.scatter(y_test, y_pred_clipped, alpha=0.7)
plt.xlabel("Actual Annual")
plt.ylabel("Predicted Annual")
plt.title("Actual vs Predicted Annual")
plt.plot([min(y_test), max(y_test)], [min(y_test), max(y_test)], color="red")
plt.show()

```

Non-finite values found in y\_pred!

max: 1.2150073969165267e+308

min: 28080.9556408886

MSE: 65909906125183.47

RMSE: 8118491.6163769895

MAE: 7104500.208496759

R^2: -26.5163803448109

```
C:\Users\Anson\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.12_qbz5n  

2kfra8p0\LocalCache\local-packages\Python312\site-  

packages\statsmodels\genmod\families\links.py:527: RuntimeWarning: overflow  

encountered in exp  

    return np.exp(z)
```

