WALK WISE: A MACHINE LEARNING EXPLORATION OF URBAN WALKABILITY

Vaibhavi Satish, Akhil Katta, Taruni Katta, Santoshi, Laila

University of North Texas

Table of Contents

Table of Figures

Acknowledgment

We would like to extend our gratitude to Professor Dr. Sameh Shamroukh for distinguished guidance and continuous support in all our research work. His skill and commitment were the most important guideline in forming the direction of our study. Dr. Shamroukh has improved the rigor and depth of our research by providing insightful feedback and counsel. We attribute a considerable part of our intellectual and professional development to his exceptional mentorship, for which we are sincerely grateful.

We also thank the hardworking team for their dedication and spirit of teamwork in realizing this project. Each one of the individuals from the project provided unique skills and perceptions that collectively contributed to the success of the research.

Abstract

This research tries to identify the complex relationship among walkability, an urban environment, and pedestrian movement through the application of machine learning methods to tackle real-world urban planning problems. Walkability is a significant aspect of an urban area in contributing toward physical activities, reducing traffic congestion, and fostering sustainable communities. This review identifies existing biases and limitations in the literature and identifies the most important studies concerning urban planning theories, behavioral frameworks, and machine learning methods. This study based its analysis on the detailed dataset retrieved from Data.gov to understand the pedestrian movement pattern through tight analysis techniques using data preprocessing, an explorative study of the data, and the implementation of several kinds of different machine learning models and activation functions for selection and training. The results obtained point to invaluable insights for walkability and pedestrian behavior in urban settings that will be helpful in the efforts toward urban planning, particularly toward a pedestrian-friendly environment and sustainability. This study is going to add to the value of the decision-making body in urban planning and policymaking that seeks the amelioration of walkability and the betterment of the quality of life in the city.

Keywords:

Walkability index, Activation function, Hyper parameter tuning, Principle component analysis, Standard scaling, Rectified Linear Unit.

1. Introduction

Walkability, an essential element of urban planning, focuses on making walking safe and convenient in cities, with accessible transportation options nearby. A key aspect of building vibrant, sustainable cities is the comprehension of pedestrian movement and its connection to factors like transit access and destination accessibility. Urban planners, policymakers, and researchers benefit from a walkability index, which quantitatively measures pedestrian-friendliness, for prioritizing investments and informed land use decisions. Our research in this context involves analyzing urban walkability and its determining factors through machine learning methods. Our goal is to offer valuable insights for urban planners and communities seeking to improve walkability and create healthier, pedestrian-friendly spaces by analyzing job-housing balance, land use diversity, traffic measures, demographics, and employment factors.

1.1 Problem and Background

The significance of promoting walking as the main mode of transportation has been accentuated by rapid urbanization, calling for advancements in urban design and transportation infrastructure. Walkable cities require a detailed understanding of the complex relationships between urban factors. We are researching to bridge this gap by examining the intricate nature of urban walkability patterns and their interplay with diverse urban variables.

1.2 Project Goals

Enhancing urban design, sustainable transportation, and pedestrian-friendly environments is the goal of our project. We achieve this by analyzing population

dynamics, job structures, land use, and transit to provide actionable insights for urban planners, legislators, and communities.

1.3 Research Questions

Building upon our investigation of urban walkability factors, our goal is to further explore the intricacies of creating pedestrian-friendly environments. Thus, our research aims to understand the complex connections between these factors and how they affect walkability. Our goal is to provide valuable insights for urban planners, policymakers, and communities seeking to create vibrant, sustainable, and pedestrian-friendly cities by addressing key research questions.

The primary research questions for this project are:

1. Does neighborhood walkability correlate with job-housing balance?
2. What is the impact of combining retail, office, and residential land use on walkability scores?
3. Is it possible to use traffic measures as a predictor for areas with high walkability?
4. Can we find a correlation between demographics, like auto ownership rates, and observed walkability?
5. In what ways do employment variables influence neighborhood walk scores?

1.4 Research Problem and Context

Our research problem arises from the necessity of unraveling the intricacies of urban walkability patterns and their relationship to diverse urban factors. While previous tries might have concentrated on factors, a complete comprehension is necessary to enhance urban walkability.

1.5 Stakeholders, Decision-Making, and Predictive Modeling

Walkability's immense influence on quality of life, health, and economic vibrancy is recognized by key urban planning stakeholders, including legislators, city officials, and residents. Determining the functionality of urban space to enhance walkability is a complex process that requires considering urban planning, transportation, public health, and community development. To effectively analyze future walkability scores and highlight areas for improvement, the use of predictive modeling techniques is essential. Predictive modeling improves urban decision-making and enhances walkability by optimizing resources and strategies.

2. Literature Review

The next section of the current research work refers to the review of literature with theories, studies, and methodologies related to walkability in general and machine learning techniques in solving real-world complex problems. This research has tried to achieve that: give a comprehensive understanding of the organizational problem and its theoretical bases by integrating insights from urban planning with behavioral theories and research on machine learning.

Urban Design Theories:

Urban Design Theories. The urban walkability literature is indicative of much emphasis on the role of pedestrians in developing friendly infrastructure, mixed land uses, and nurturing cities that are livable and alive. The New Urbanism and Transit-Oriented Development (TOD) paradigms, therefore, espouse compact development, mixed-use, and easy accessibility to public transit, as in Seaside, Florida.

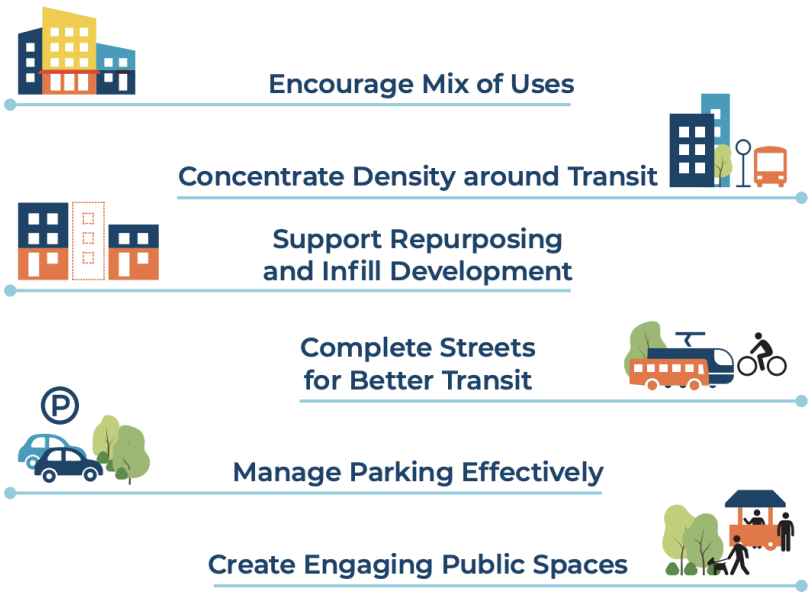


Figure 1: Transit- Oriented Development (TOD)

Behavioral Theories:

These theories are highly plausible to explain or clarify behavior, such as the Theory of Planned Behavior (TPB) or the Social-Ecological Model (SEM). TPB takes into consideration the subjective norms, perceived behavioral control, and attitude in the determination of walking intentions, while SEM investigates a wider framework, expanded to include wider environmental influences. The artificial neural network (ANN), as applied in research along the same line, would mean the one by Mas-Cabo et al. (2020) to determine timely labor, so that machine learning is applicable in relevant healthcare domains.

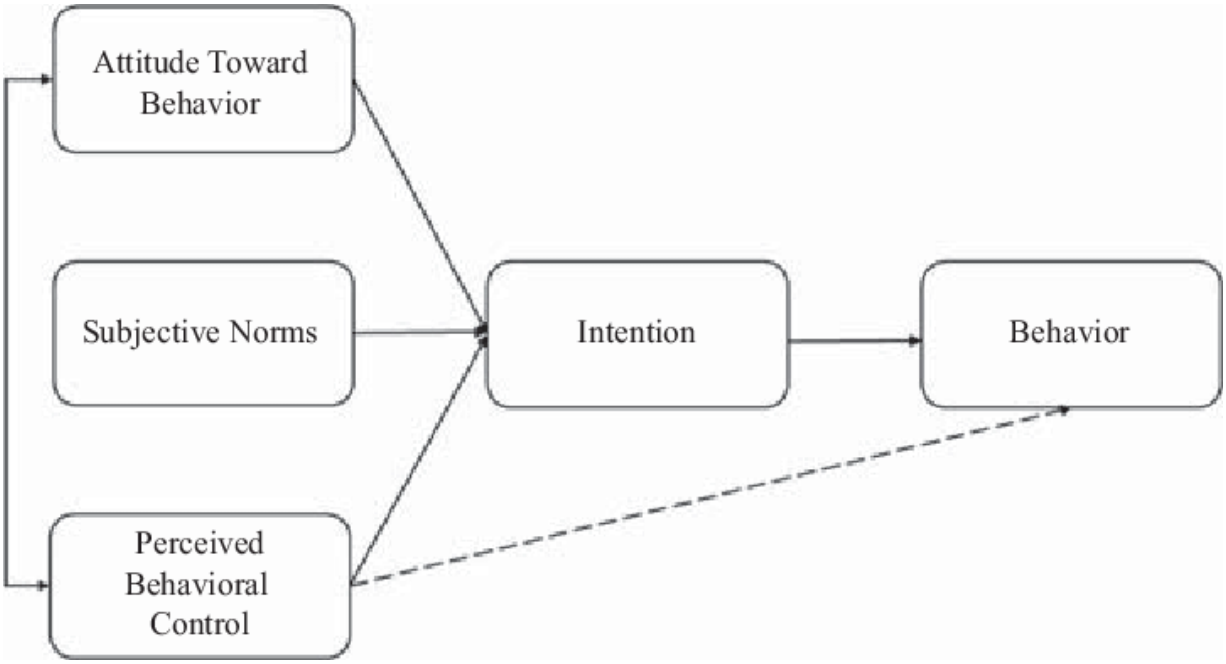


Figure 2: Theory of Planned Behavior (TPB) or the Social-Ecological Model (SEM)

2.1 Bias and Limitations in Literature

Biases and limitations still exist despite the abundance of literature on urban walkability and machine learning. The focus of urban walkability research may disproportionately favor developed urban areas, disregarding the obstacles faced by underserved communities. In addition, the lack of specificity in machine learning studies makes it hard to distinguish the specific research limitations. In their study, Ray et al. (2023) (Ray, 2023) emphasized the importance of thorough evaluation and clear features in machine learning applications, comparing ANN and response surface methodology (RSM) for predicting concrete strength.

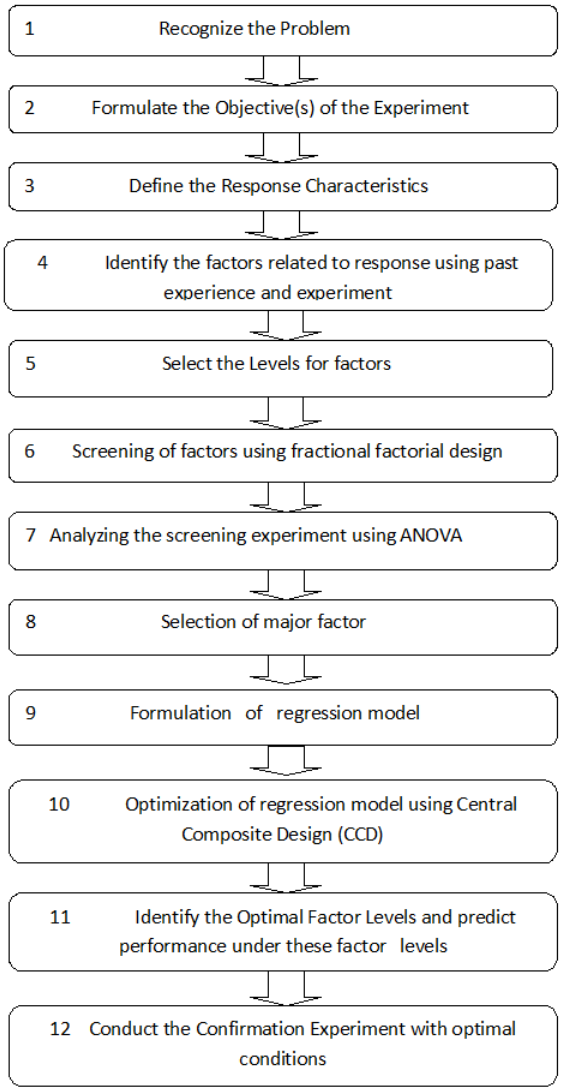


Figure 3: Steps in Response Surface Methodology (RSM)

2.2 Research Study

Hauer et al. (2021) (Hauer, 2021)presented groundbreaking approaches to natural language processing by exploring sense annotation through translations, both in semi-supervised and unsupervised settings. While it may not be specifically focused on urban walkability, this study showcases how innovative approaches can be used to tackle intricate research challenges, reflecting the demand for creative solutions in urban planning.

2.3 Similar Problems

The literature review by De Vos et al. (2022) introduced a conceptual model and research agenda, aiming to deepen our understanding of perceived walkability. Through the synthesis of existing research and the proposal of future directions, this study presents a roadmap for addressing comparable challenges in urban walkability research. Hijriyah et al. (2023) utilized systematic literature review techniques to examine trends in walkability research, highlighting the significance of comprehensive methodologies in analyzing complex urban phenomena.

2.4 Conclusion

By examining the literature, we uncover the various facets of urban walkability and the potential of machine learning to address challenging real-world problems. This review establishes a foundation for empirical analysis in urban walkability research by integrating insights from urban planning theories, behavioral frameworks, and machine learning methodologies. This review carefully examines biases, limitations, and relevant studies to establish the framework for informed decision-making and future research in urban planning and machine learning integration.

3. Research Methods

The dataset was selected after careful evaluation for its relevance and comprehensiveness. The dataset from Data.gov provides comprehensive information on urban variables essential for analyzing pedestrian activity patterns. The dataset considers multiple indicators, such as administration, demographics, employment, density, diversity, design, transit access, and destination accessibility, to provide a well-rounded view of urban environments. The wide range of information makes it perfect for studying the complex connection between urban features and pedestrian behavior.

3.1 Data and Procedures

The dataset contains 117 variables, providing detailed information on urban characteristics such as population, housing, employment, land use, transportation, and accessibility. The analysis incorporates one dependent variable that reflects pedestrian activity levels and multiple independent variables. Geographic representation in diverse urban areas is ensured with the dataset covering all 50 states, Puerto Rico, and other territories. By utilizing this dataset, the research methodology aimed to streamline the analysis process and implement rigorous analytical techniques to address the research questions and extract meaningful insights effectively.

3.2 Measures

Dependent Variable:

In this study, the dependent variable is represented by pedestrian activity levels. and Walkability Index as dependent variable. It gives a metric on overall walkability of a location based on a combination of factors like street connectivity, proximity to transit, land use mix, and density(D1 to D5).

Independent Variable:

The dataset includes multiple independent variables that cover diverse urban characteristics, like categories such as Administrative, Core-Based Statistical Area Measures , Area, Demographics, Employment, Density (D1) , Diversity (D2) , Design (D3), Transit Access (D4) , Destination Accessibility (D5) as independent variables These key variables are crucial for comprehending the factors that impact pedestrian behavior and urban walkability.

3.3 Analysis

The research employed a methodical analytical approach to derive useful findings from the data. The analysis process consisted of multiple important steps.

1. Data Pre-processing: The first step consisted of cleaning the dataset, removing inconsistencies, handling missing values, and ensuring data quality. Methods like imputation and outlier detection have been used to improve the dataset's reliability.
2. Exploratory Data Analysis (EDA): Preliminary insights into the dataset's characteristics were obtained through descriptive and visual analyses. This involved summarizing variable distribution, identifying patterns and trends, exploring correlations among variables, and visualizing spatial relationships through GIS tools.
3. Feature Engineering: We identified numeric features from the dataset and applied PCA to optimize the predictive modeling process by dealing with the multicollinearity problem faced above. The goal of PCA is to reduce the dimensionality of the dataset from 117 columns to 25 Principal components.
4. Model Selection: The choice of machine learning algorithms, including linear regression and artificial neural networks (ANN), was made considering the research objectives and dataset properties. Their proficiency influenced the choice of models in capturing and predicting pedestrian activity levels based on the independent variables.

Note. There is significant multicollinearity in the data, which we are going to deal with by using PCA.

5. Model Training: The prepared dataset was used to train the selected models to understand the connections between urban variables and pedestrian activity levels. To optimize performance, the training process involves fitting the models to the data and fine-tuning their parameters.
6. Model Evaluation: Evaluation of trained models involved the use of appropriate metrics, including Mean Squared Error (MSE) for regression tasks. The purpose of model evaluation is to ensure reliable and accurate analysis results by measuring predictive performance and identifying areas for improvement.

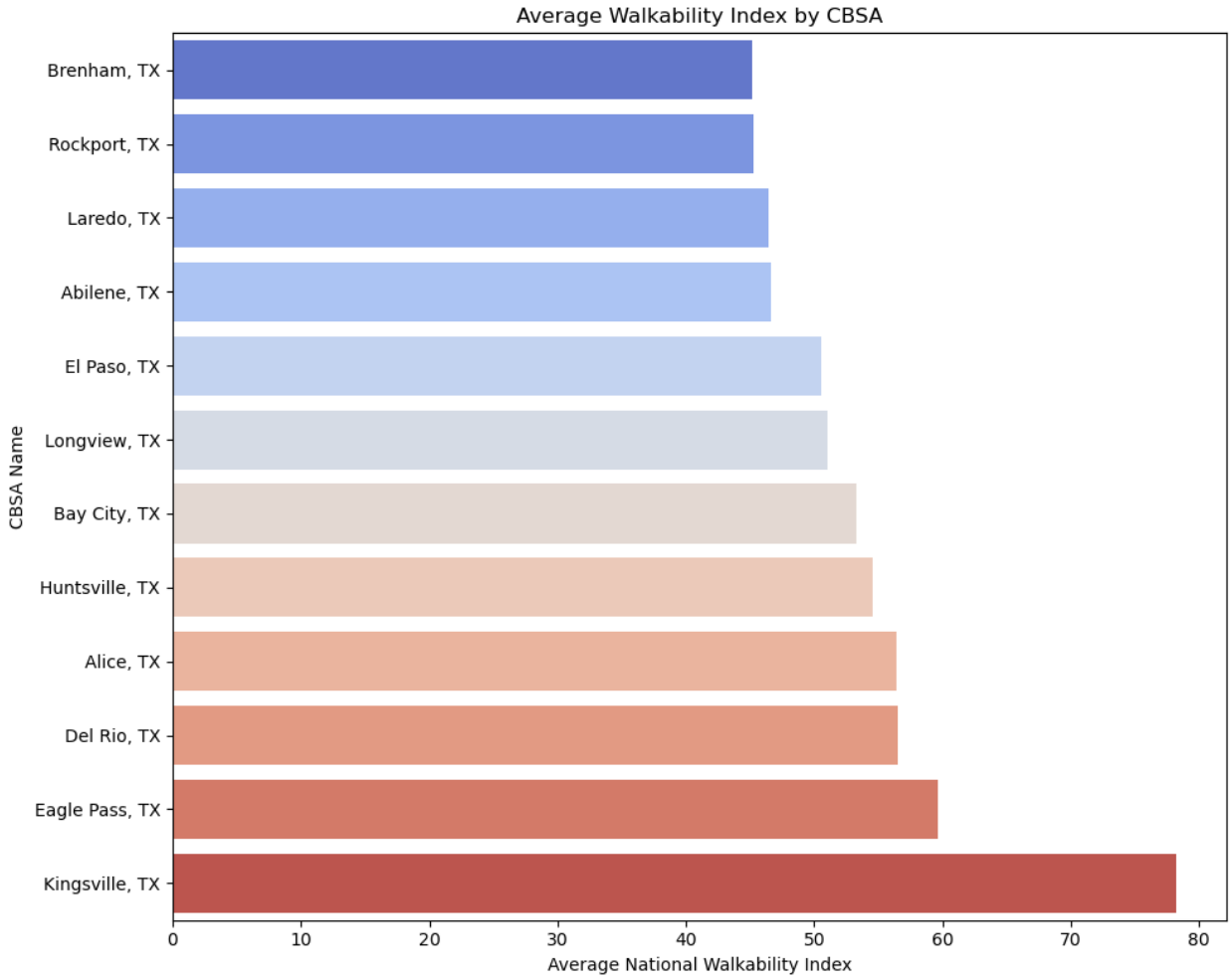


Figure 4: Average Walkability Index

Note. The figure has Average Natwalkind and CBSA Names on the X and Y axis respectively.

7. Interpretation and Insights: The interpretation of the analysis results provided valuable insights into urban walkability and pedestrian behavior factors. The knowledge gained from these insights influenced efforts to create pedestrian-friendly environments through urban planning and policymaking. Furthermore, the researchers examined the implications of the findings within the context of existing literature and theoretical frameworks to ensure a thorough comprehension of the research outcomes.

The study aimed to provide valuable insights for urban planning, promoting informed decisions to improve walkability and create sustainable communities.

3.4 Methods used

Imputation: we have used imputation to deal with null values in our dataset. The nulls in all the numeric variables have been imputed with mean and categorical variables with mode. We have found that more than 90% of the values in "D4A", "D4C", "D4D", "D4E", "D5BR", "D5BE", "D5DR", "D5DRI", "D5DE", "D5DEI" are “-0.99999” which has been used as a place holder value in these columns. So, they have been dropped.

Correlation: We have checked for correlations in the independent variable to check for any interesting correlations and found some very interesting things like how as populations grow, so does the infrastructure to support job commuting by car and how as total employment raises, the workers earning more than $3333 also increases disproportionately which indicates a high influx of high paying jobs.

PCA: Since we have 117 variables originally, we needed to do some dimensionality reduction. We have considered Forward selection, Backward selection and Principle component analysis

among which we ended up choosing PCA as it reduces the data leakage and deals with the multicollinearity problem we had with the dataset.

"PCA is a practical and valuable statistical technique that transforms large correlated data into small uncorrelated data using principal components (PCs) (Shang et al., 2017). PCs can be expressed as linear combinations of the original input variables, which retain the complete information of the original data."(Fan et al., 2024). Using PCA we have reduced our dataset to 25 columns.

Standard Scaling: Machine learning algorithms tend to give higher weightage to bigger numbers in the dataset no matter how insignificant they are so, to deal with this problem, we have scaled the entire dataset Using standardscaler() function from python which does the standardization by using the mean and standard deviation of each variable.

Linear Regression

We have chosen Linear regression as a baseline model for our analysis which fits a linear equation to the data and estimate coefficients to each variable depending on how much influence that independent variable has on the dependent variable.

Artificial Neural Networks

Building a shallow neural network to compare the performances of various activation functions has always been the core of our project. A neural network is a kind of model that is created to simulate the functioning of human brain consisting of neurons and nodes organized in layers.

For our analysis we have decided to use a shallow neural network because of the computational constraints with 1 hidden layer on which we will apply different activation functions and an output layer with one node with linear activation function which predicts the outcome.

Activation Functions used:

ReLU (Rectified Linear Unit):

Definition: If the input is positive, it comes from the input itself; If the input is negative, zero is the output.

LeakyReLU:

Definition: It is comparable to ReLU, but allows a slight, non-zero overhead when the input is less than zero and the unit is not operating.

PReLU (Parametric ReLU):

Definition: A variant of LeakyReLU where the leak coefficient are determined by training rather than by default.

RReLU (Randomized ReLU):

Definition: It is comparable to LeakyReLU, but with a random negative part slope during training and a fixed slope during testing.

SReLU (S-shaped ReLU):

Definition: Combining several blockwise linear functions yields an S-shaped function.

ELU (Exponential Linear Unit):

Definition: If the input is negative, negative gives an exponential decay to infinity; If input is positive, it generates only input.

PeLU (Parametric ELU):

Definition: ELU variant with parameters specifying the size of the output.

Selu (Scaled Exponential Linear Unit):

Definition: The ELU is scaled using a fixed scale factor to maintain accuracy.

Maxout:

Definition: It efficiently uses several affine functions to generalize ReLU without requiring a special method for nonlinearity.

ELiSH (Exponential Linear Sigmoid SquasHing):

Definition: Sigmoid-like output combined with ELU based on the range of inputs.

HardELiSH:

Definition: A piecewise linear approximation of ELiSH that sacrifices smoothness in order to be faster.

4. Data Analysis

Our data analysis is specifically focused on descriptive statistics and correlation analysis results, and we evaluated the performances using different activation functions in the Neural networks.

4.1 Descriptive Statistics

Descriptive Statistics provided an understanding of the size, structure, and data types. Used data.describe() for understanding the statistical details such as mean, minimum value, maximum value, and standard deviation.

The key components are count, mean, standard deviation, minimum, maximum values, and percentiles:

1. The count showed the number of non-null values per variable, this indicates that there are no missing values and the missing values have been filled up using the mode value.
2. Mean provided the information of the average value of each variable and this information gave he central tendency insights.
3. Standard deviation suggests the number of variations happening around the mean values. The data points are spared out in a wide range as we have a high range of std.
4. The minimum and maximum values helped us understand the range within the data variers and this has been calculated for each column.
5. The 25th, 50th, and 75th percentile of each column helped in understanding the data distribution. And median which is the 50th percentile provided the midpoint information. Overall, this information gave insights into the first quartile, median, and third quartile.

4.2 Analysis based on the Descriptive Statistics

1. Urban Planning and Economical Analysis, here we compared the demographical and economic variables for understanding urban planning and policymaking. Some of the variables used are TotPop, Workers, and R_LowWageWk.
2. Socio-economic research factors helped to analyze the correlation between the economic and demographic factors to identify the trends and potential area trends. Some of the variables used are AutoOwn0, Shape_Length, Shape_Area and D2A_JPHH
3. Geographical Analysis helped us understand the patterns related to development, land density, population, and employment distribution. Some of the variables used are GEOID10 and STATEFP.

4.3 Correlation Analysis

Correlation Analysis is a critical part of our analysis. It provided insights into how the different variables in the walk wise are related to each other. Overall, we have strong correlations in terms of the target variable for predicting the model. The stronger correlations are identified by seeing the red blocks which are directly related to each other, more populations and employment variables showed up in this area. The negative correlation is seen in the blue area of the graph, and we can see no correlation or weaker correlation on the white or light blue part of the graph.

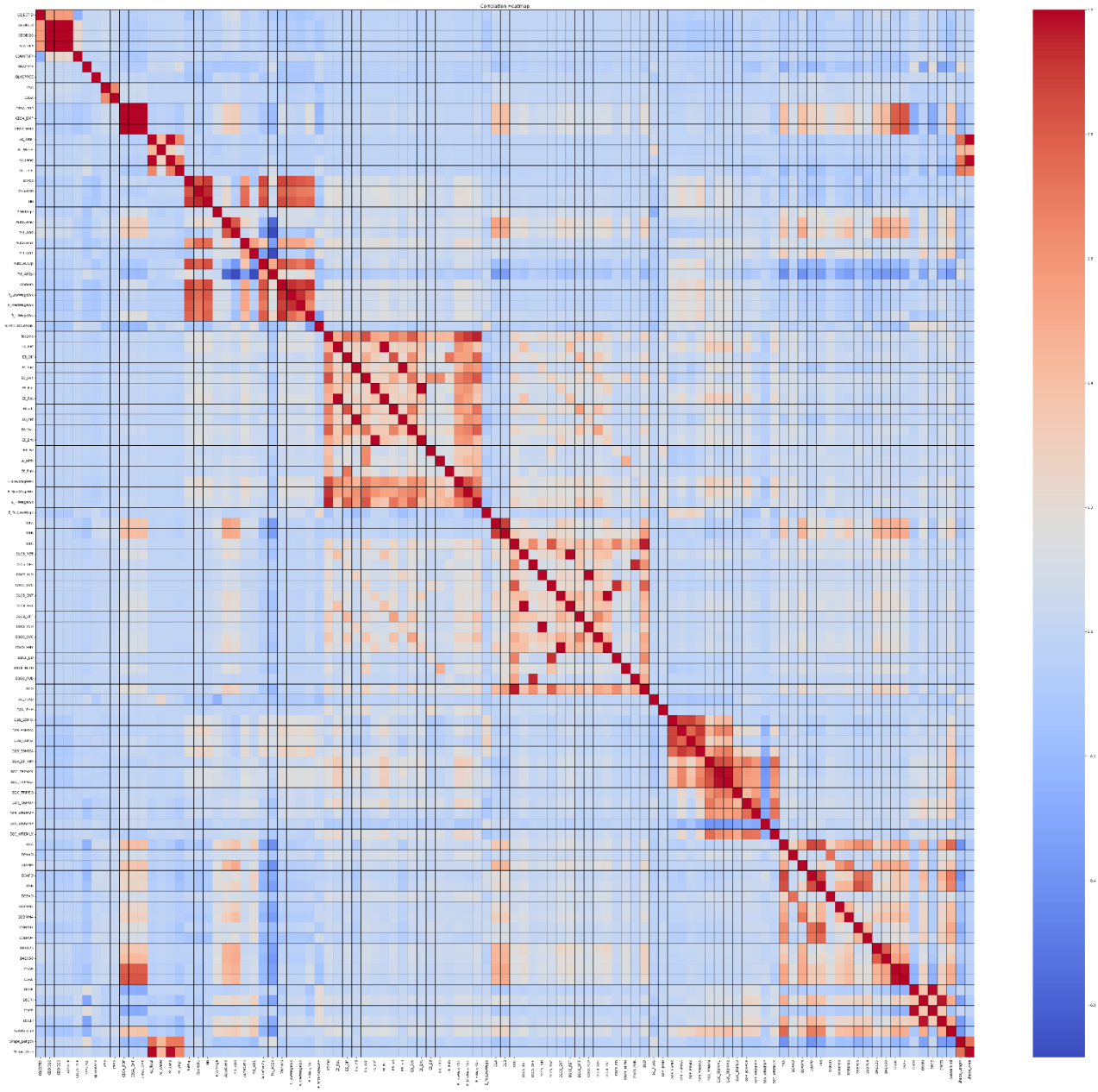


Figure 5: The Correlation Analysis

In the overall analysis, we can see a strong correlation with the target variable which is the net walkability index which we used for prediction. This heatmap also identified that multicollinearity can be shown as a problem.

The limitation of Correlation analysis is the graph only showed the linear relationship of the model, which can show us how strong that matrix is and how the positive pattern is shown in the graph. The major con is, that overall can identify the variable which indicates that the graph is clumsy and messy as we have a larger dataset.

4.4 Visualization

After EDA we have included a few scatter plots, which determined the relationship between variables, and we have given the threshold of 0.6 we got about 160 graphs showing the different relations and analyses among all we have a few interesting scatter plots that are listed below.

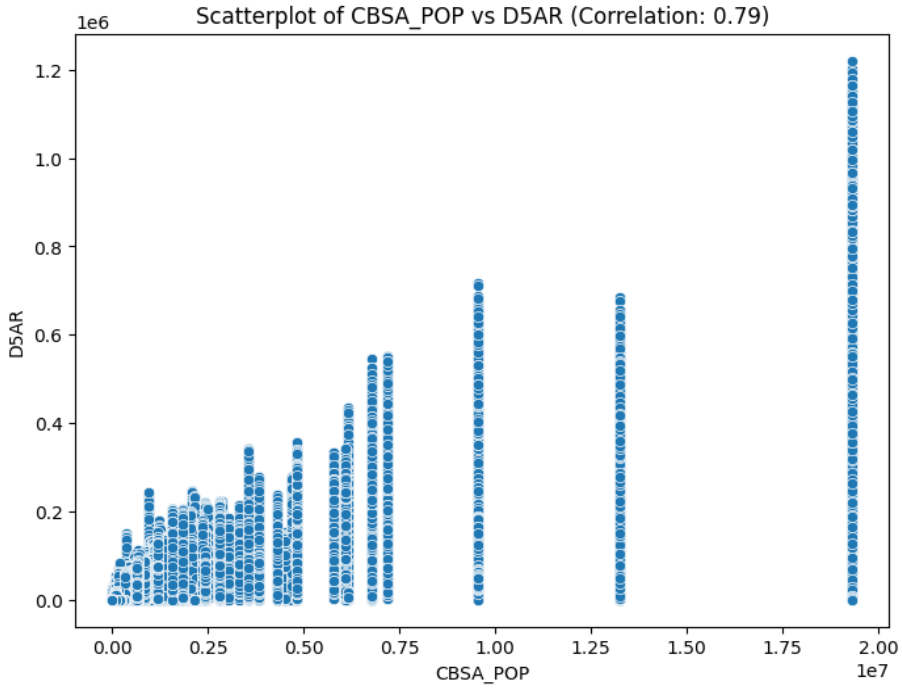


Figure 6: Total population in CBSA.

1. Relation between CBSA_POP and D5AR

The scatter plots show the relation between the Total population in CBSA and Jobs within 45 minutes of auto travel time. We can see that they are highly correlated with 0.794, by this we can explain the accessibility of jobs with 45 minutes of travel time is good and This indicates that as the population increases the infrastructure for the travel or commute is also growing.

2. Relation between Ac_Total and Ac_Land

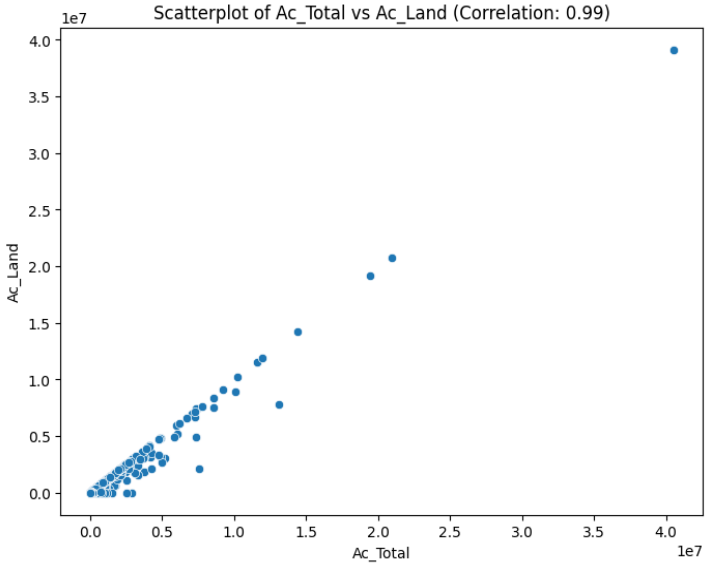


Figure 7: Total geometric area of the CBG and Total land area.

The relation between the Total geometric area of the CBG and the Total land area has a correlation of 0.986 which is good enough to explain the area measures and the proportion of land and non-land areas.

3. Relation between TotPop and Workers

The relation between Total Population and Count of workers in CBG has an interesting plot with 0.874 as a correlation, this highlights the economic activity of the population.

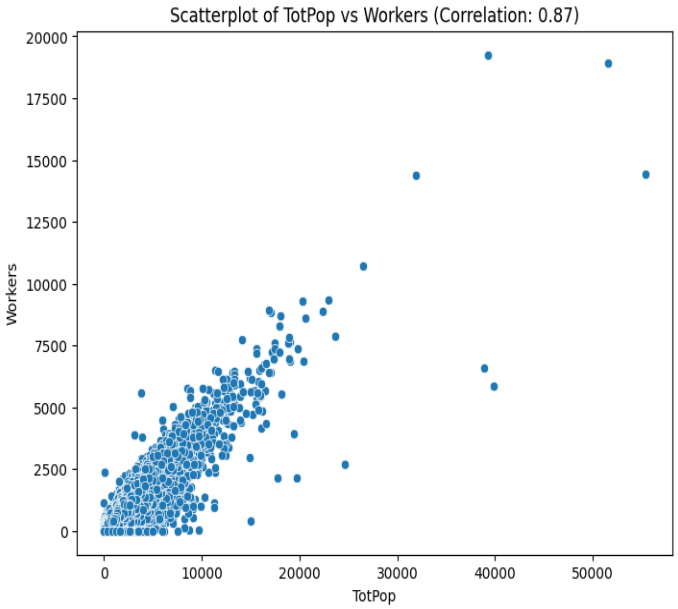


Figure 8: Total Population and Count of workers in CBG.

4. Relation between R_LowWageWk and R_HiWageWk

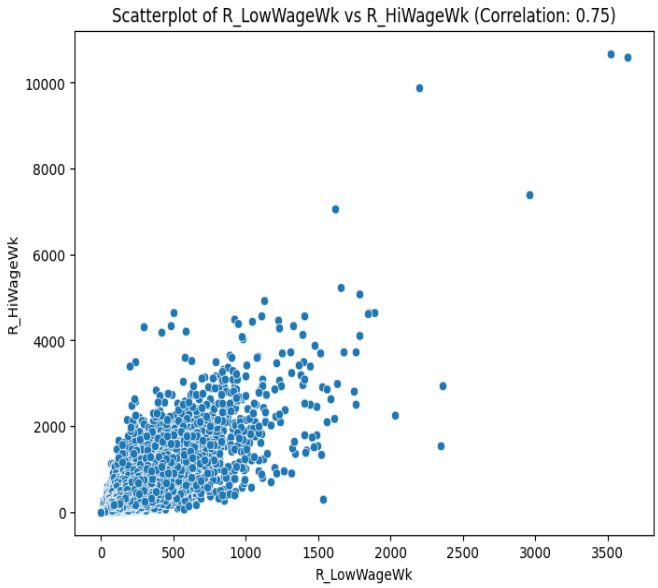


Figure 9: Count of low-wage workers and Count of high-wage.

The Count of low-wage workers and Count of high-wage workers relation explains the geographic area wage differences, this shows the mixed economic structure for development and the correlation is 0.751.

5. Relation between TotEmp and E_HiWageWk

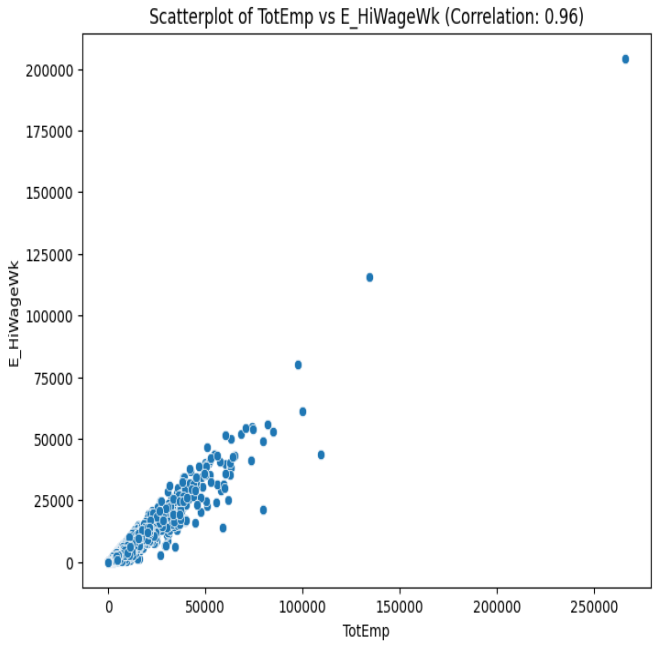


Figure 10: Total Employment and the number of workers earning.

The relation between Total Employment and the number of workers earning $3333/month or more is highly correlated with 0.9605 which indicates that high-paying jobs are on the rise.

6. Relation between D2B_E5MIX and D2B_E8MIX

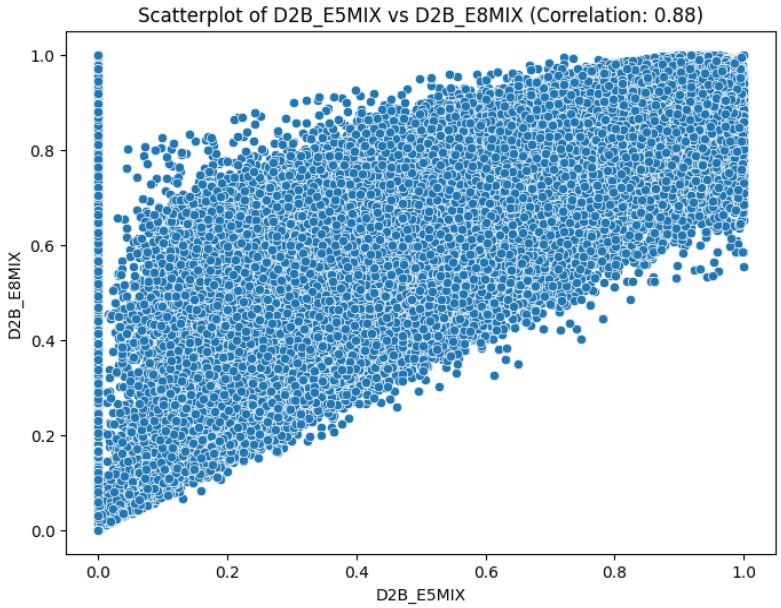


Figure 11: complexity of job markets in the different locations.

The 5-tier employment entropy and 8-tier employment entropy correlate 0.882 which explains the complexity of job markets in the different locations.

Research Question 1: Does neighborhood walkability correlate with job-housing balance?

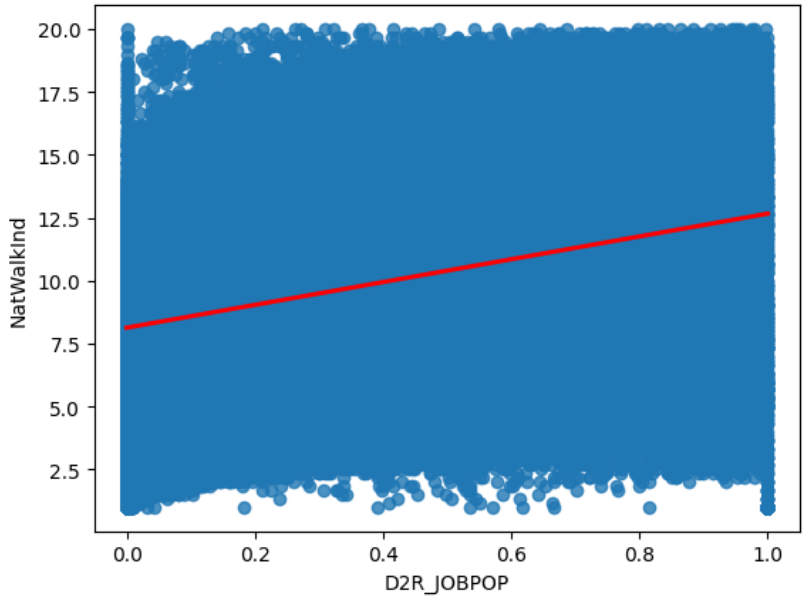


Figure 12: Scatter plot of NatWalkInd vs. D2R_JOBPOP

From the scatter plot of NatWalkInd vs. D2R_JOBPOP, the two variables have a slightly positive to no correlation. In other words, it means that when the ratio of jobs to population is high, the walkability score is likely to be high. Areas in which they have more jobs than people living in them could possibly have better walkability due to amenities supporting workers and residents in this regard.

Research Question 2: What is the impact of combining retail, office, and residential land use on walkability scores?

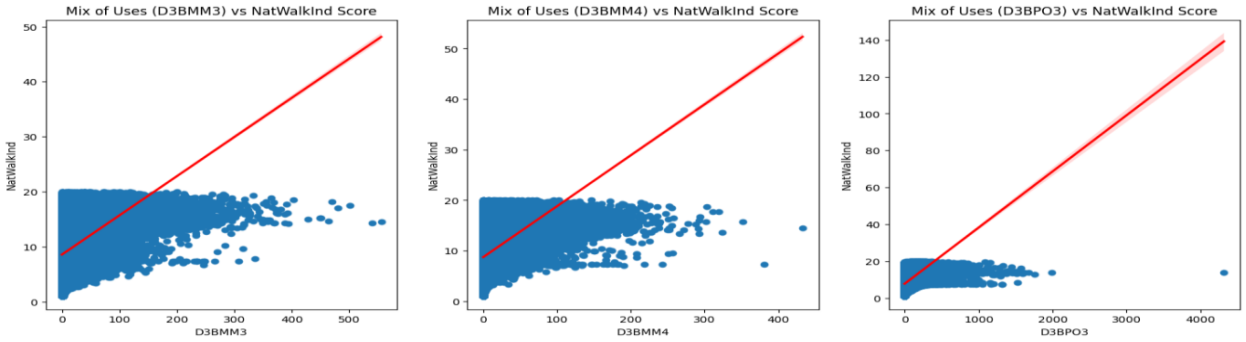


Figure 13: D3BMM3 versus NatWalkInd

There is a positive correlation between D3BMM3 and NatWalkInd. As D3BMM3 increases, indicating a denser mix of residential, multifamily, and commercial uses, the NatWalkInd score also tends to increase. This suggests that areas with a higher mix of residential and commercial functions are more likely to have better walkability, probably due to the availability of various amenities and services close to dwellings.

D3BMM4 versus NatWalkInd:

Similar to D3BMM3, D3BMM4 shows a positive correlation with NatWalkInd. As D3BMM4 increases, which indicates a higher mix of office and commercial uses, the NatWalkInd score also tends to increase. This indicates that areas zoned with a combination of office and commercial uses are likely to be more walkable, due to the presence of workplaces, shops, and services that attract foot traffic.

D3BPO3 versus NatWalkInd:

Unlike the first two plots, there is no clear correlation between D3BPO3 and NatWalkInd. The scatterplot shows a more dispersed pattern, suggesting that the mix of public open space and recreation areas does not consistently impact walkability scores. This implies that other factors,

such as the quality and accessibility of public spaces, may play a more significant role in determining walkability.

Research Question 3: Is it possible to use traffic measures as a predictor for areas with high walkability?

Traffic Measure 1 vs. NatWalkInd:

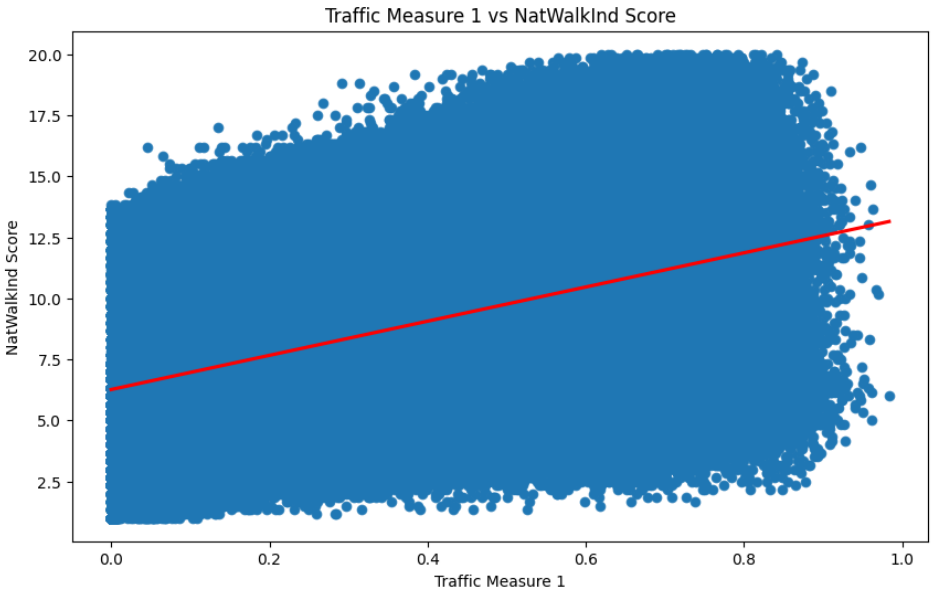


Figure 14: Traffic Measure 1 vs. NatWalkInd

- Traffic Measure 1 has a positive correlation with NatWalkInd.

- As Traffic Measure 1 increases, the NatWalkInd score also tends to increase. This suggests that areas with higher traffic measurements, such as traffic volume or density, are likely to have better walkability. This could be due to more transportation options available, including public transit and bike lanes, making walking a preferred option.

Traffic Measure 2 vs. NatWalkInd:

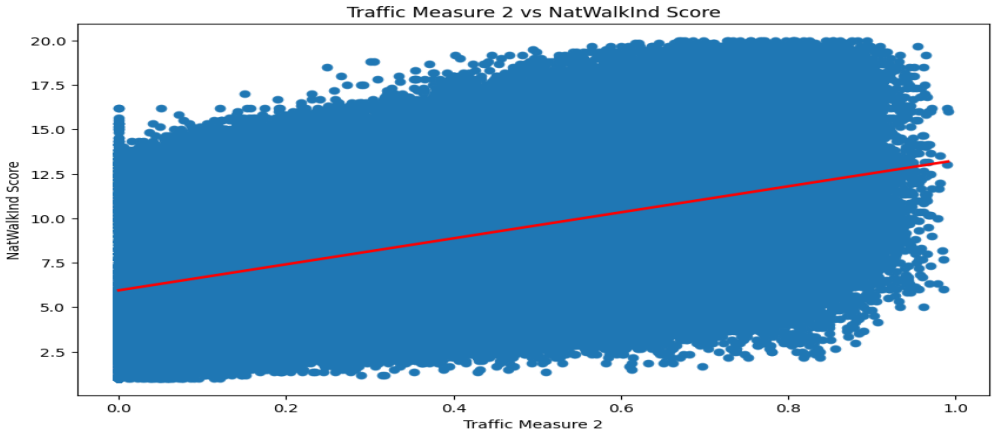


Figure 15: Traffic Measure 2 vs. NatWalkInd

-

Similarly, NatWalkInd shows a positive relationship with Traffic Measure 2.

- As Traffic Measure 2 increases, the NatWalkInd score also increases. This suggests that areas with higher traffic measures, such as greater traffic congestion or delay, may have better walkability. In these areas, the presence of mixed-use development and pedestrian-friendly infrastructure likely makes walking more convenient and enjoyable.

Research Question 4: Can we find a correlation between demographics, like auto ownership rates, and observed walkability?

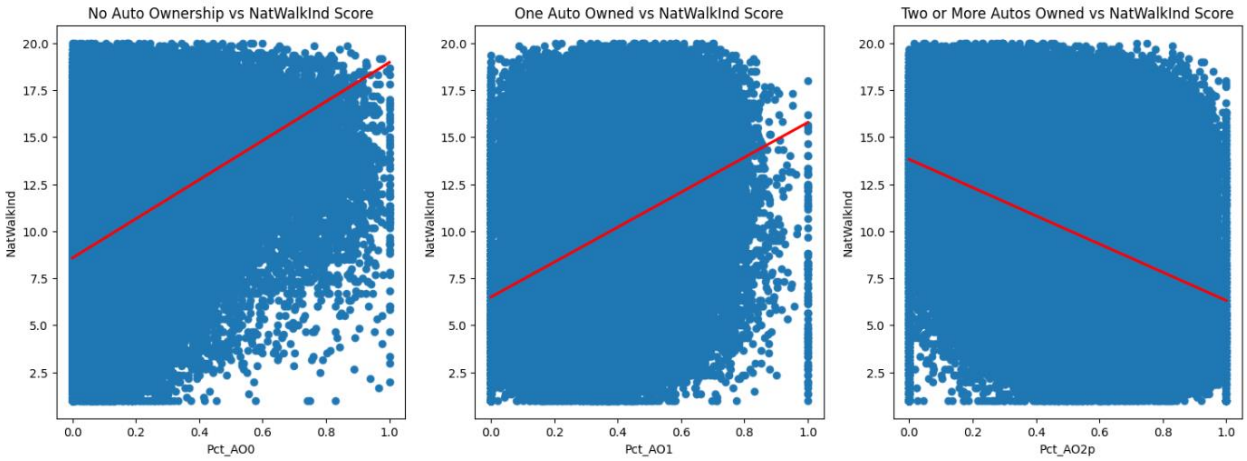


Figure 16:Correlation between demographics

There is no clear correlation between auto correlation rates and walkability. This would suggest that factors like accessibility to public transportation and a quality built environment may be very strong determinants in walkability for regions with greater auto ownership rates.

Research Question 5: How does the usage of various activation functions affect metrics such as error(MSE and MAE), R-Squared, Adj R-Squared when applied on a shallow neural network.

	Activation Function	MSE	MAE	R2	Adjusted R2
0	ReLU	0.730824	0.637704	0.961902	0.961880
1	LeakyReLU	0.766813	0.668507	0.960026	0.960003
2	PReLU	0.761451	0.656491	0.960305	0.960283
3	RReLU	0.568426	0.576477	0.970368	0.970351
4	SReLU	0.788996	0.679787	0.958869	0.958846
5	ELU	1.054497	0.738320	0.945029	0.944998
6	PeLU	1.314620	0.845946	0.931469	0.931430
7	Selu	0.816229	0.688758	0.957450	0.957426
8	Maxout	1.112729	0.786692	0.941993	0.941960
9	ELiSH	1.131057	0.754601	0.941038	0.941004
10	HardELiSH	1.032446	0.728436	0.946178	0.946148

The activation function of neural networks determines the output of a node in the neural network. It plays a crucial role in setting the smoothness of the fitting curve that predicts the output. Metrics such as mean square error (MSE), mean absolute error (MAE), R-square error (R2), adjusted R-square can be used to evaluate the efficiency of the neural network model when using different activation functions. We analyzed the impact of the implementation of these functions using the metrics provided:

- Mean Squared Error (MSE):

In the model we can see that the MSE is low which indicates that its shows a smaller average squared error while comparing the actual and predicted values. Based on the graphs, we can say that ReLU, LeakyReLU, PReLU, RReLU, SReLU are performing well, where ad RRelu is showing the lower values of MSE. Thus, RRelu is on average and closer to the true value.

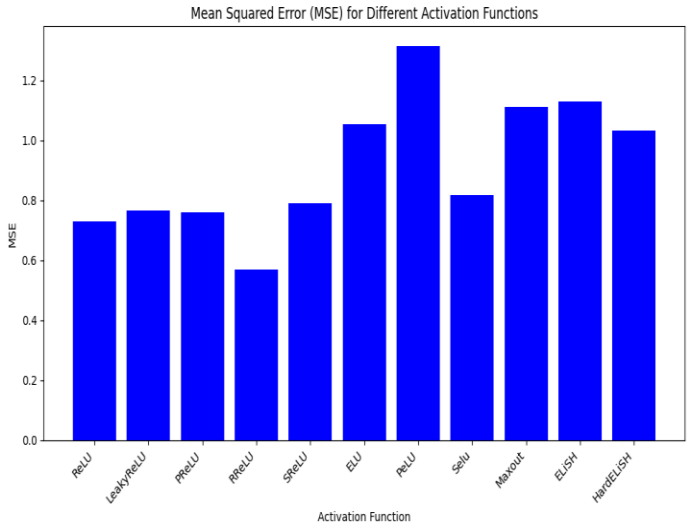


Figure 17: Mean Squared Error (MSE):

- Mean Absolute Error (MAE):

The MAE value is also low, which indicates that the predicted values are closer to the actual values. The RRelu function performs the best metrics with few errors on average without deviations squaring.

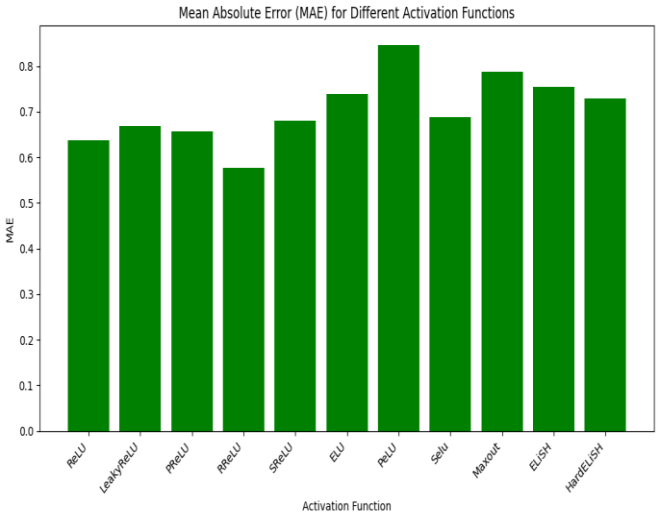


Figure 18: Mean Absolute Error (MAE)

- R-Squared (R^2):

The R-Squared shows the proportion of variances for the dependent variables which explains the independent variables in regression model. The higher the R-Squared the more variance which gives us a desirable model. In most activation functions R-Squared is mostly similar and in RRelu we can see a slight edge. Overall, we can say that different activation functions are relatively comparable.

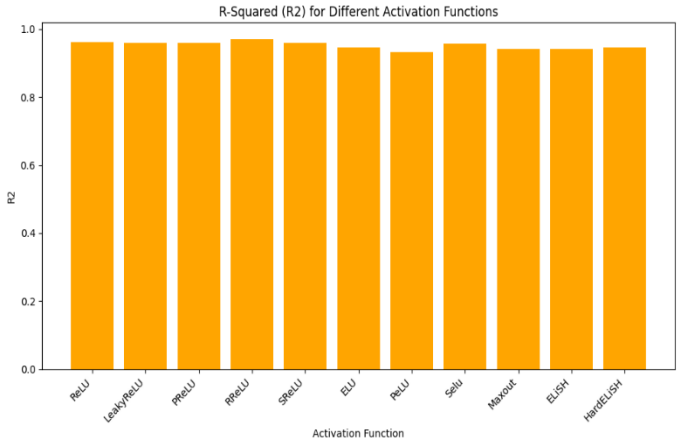


Figure 19: R-Squared (R^2)

- Adjusted R-Squared:

This will modify the R-Squared values to predict the model. The higher the better measures which gives us a good model. Here we can again see that RRelu has the higher value which indicated the predicted values and perform robust.

In summary, the RReLU activation function appears to perform well for shallow neural networks in all given metrics, based on the given data if the adjusted R2 shows a larger percentage of captures it is a variance that is not too normal, thus balancing the bias-variance trade-off. Variants such as ReLU also perform well, consistent with the literature. This is because ReLU-type functions, due to their computational efficiency and gradient propagation properties, are well known to train effective deep correlations.

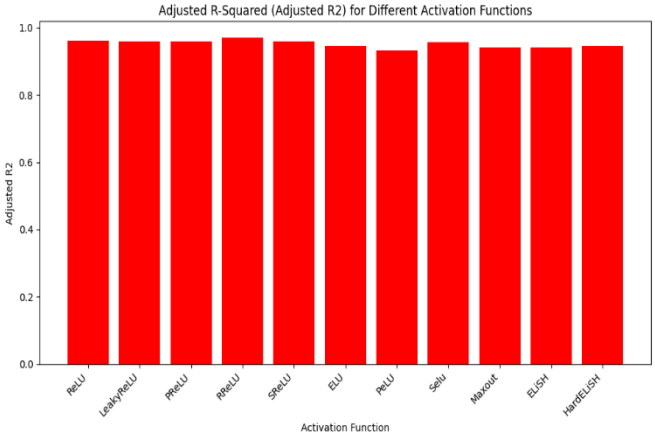


Figure 20: Adjusted R-Squared

5. Conclusions and Recommendations

Conclusions:

In Conclusion, our study investigates the intricate relationship between urban walkability and various socioeconomic and urban planning factors with the help of machine learning. We have identified the main features of the scrutinized dataset that reveal the most relevant information related to pedestrian behaviour and determinants of walkability in urban areas. The major correlations our findings pointed out are related to the factors of job-housing balance, land use diversity, traffic measures, demographics, and employment variables; they thus influence walkability scores. To use descriptive statistics, correlation analysis, and visualization techniques to tease out patterns and trends within urban data with the intent of providing subtle perspectives back to the urban planner and policymaker. Besides, the model was used to generate a predictive model analysis framework, which is improving the understanding of pedestrian activity levels in making urban planning decisions. Our results strive to contribute toward the search for vibrant, sustainable, pedestrian-friendly cities by trying to bridge the gap between theoretical and practical application. Thus, based on these insights, the current discussion moves forward to provide a guide for designing interventions and policies that foster walkable urban environments in the name of healthier and more livable communities for all residents.

Recommendations:

1.Mixed land use: The uses that are residential should mix with those of commercial and office in an urban setting. High positive correlations between the mixed land use and walkability scores show that places with a diverse mix of functions would generally be more pedestrian-friendly in nature. Urban planners and policy makers need to adopt land use planning techniques that are in

line with zoning requirements and incentives that encourage mixed-use development, which provides vibrant, pedestrian-friendly neighborhoods.

2.Invest in Transportation Infrastructure: Invest in transportation infrastructure supportive of pedestrian movement and reducing automobile dependence. The positive high correlation between the two means that better transportation options, on the whole, bring about more walkability, and efforts such as public transit improvements, friendlier streetscapes for pedestrians, and the inclusion of bike lanes would bring this about.

3.Enhancing Public Spaces: Quality and access to public spaces should be improved to make them more attractive and people-friendly places for walking. Even though the association with public open space was not significant, investments in well-designed parks, plazas, and recreation areas would help to create attractive and inviting pedestrian environments. Another way is to increase walkability in streetscapes and to provide amenities such as benches, lighting, and greenery.

4.Socio-economic disparities: Recognize and address the different socio-economic disparities that might impact walkability. Even though this study failed to identify any firm relationship between the rate of auto ownership and walkability, some other variables like income levels and the availability of transportation resources may be in play. Addressing these disparities could be through better public transportation in areas without them, better access to affordable housing, and more equitable urban planning policies.

Continued Research and Monitoring:

Model-wise, More activation functions can be used for training and add run time as a metric to compare the models better. The density of the models can also be changed to experiment with Deep neural networks.

There is a trade-off between the efficiency and complexity of the model, in addition to the urban elements remains essential in continued research and monitoring on the complex interactions that occur among urban factors in association with walkability. The use of machine learning techniques and predictive models will provide useful insights into pedestrian behavior and urban planning trends. Evidence-based strategies toward an improved walkable environment and sustainable pedestrian-friendly cities require more collaborative research among academic scholars, policy developers, and community stakeholders.

Execution of such recommendations in various urban setups is obviously going to bring about making the cities more walkable, livable, and sustainable, where people can be physically active and transport may be bettered, thus bringing about a better quality of life for all residents.

6. References

Hauer, B. K. (2021). Semi-supervised and unsupervised sense annotation via translations. Proceedings of the Conference Recent Advances in Natural Language Processing - Deep Learning for Natural Language Processing Methods and Applications. https://doi.org/10.26615/978-954-452-072-4_057.

Ray, S. H. (2023). Comparison of Artificial Neural Network (ANN) and response surface methodology (RSM) in predicting the compressive and splitting tensile strength of concrete prepared with glass waste and tin (SN) can fiber. *Journal of King Saud University - Engineering Science*, https://doi.org/10.1016/j.jksues.2021.03.006 .

De Vos, J., Lättman, K., Van der Vlugt, A. L., Welsch, J., & Otsuka, N. (2023). Determinants and effects of perceived walkability: a literature review, conceptual model and research agenda. Transport reviews, 43(2), 303-324.

Emmanuel, T., Maupong, T., Mpoeleng, D., Semong, T., Mphago, B., & Tabona, O. (2021). A survey on missing data in machine learning. Journal of Big data, 8, 1-37.

Fan, C., Zhang, N., Jiang, B., & Liu, W. V. (2024). Using deep neural networks coupled with principal component analysis for ore production forecasting at open-pit mines. Journal of Rock Mechanics and Geotechnical Engineering, 16(3), 727-740.

Hauer, B., Kondrak, G., Luan, Y., Mallik, A., & Mou, L. (2021). Semi-supervised and unsupervised sense annotation via translations. arXiv preprint arXiv:2106.06462.

Hijriyah, L., Alias, A., & Sahabuddin, M. F. M. (2023). Exploring walkability research trends based on systematic literature review (SLR) by applying PRISMA. Open House International, 49(1), 63-121.

Marcu, D. C., & Grava, C. (2021, June). The impact of activation functions on training and performance of a deep neural network. In 2021 16th International Conference on Engineering of Modern Electric Systems (EMES) (pp. 1-4). IEEE.

Zhang, S., Lu, J., & Zhao, H. (2024). Deep network approximation: Beyond relu to diverse activation functions. Journal of Machine Learning Research, 25(35), 1-39.

```
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
import networkx as nx
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, LeakyReLU
from tensorflow.keras.activations import relu, elu, selu
```

Dropping "D4A", "D4C", "D4D", "D4E", "D5BR", "D5BE", "D5DR", "D5DRI", "D5DE", "D5DEI" as they contain too many place holder values "-0.99999"

```
data = pd.read_csv('EPA_SmartLocationDatabase_V3_Jan_2021_Final (2).csv')
columns_to_exclude = ['D2A_Ranked', 'D2B_Ranked', 'D3B_Ranked', 'D4A_Ranked', 'CSA_Name', 'CBSA_Name', "D4A", "D4C", "D4D", "D4E", "D5B
data = data.drop(columns=columns_to_exclude)
```

```
# Loop through each column and replace NaN values with the column's mean
for column in data.columns:
    data[column].fillna(data[column].mean(), inplace=True)

data['CSA'] = data['CSA'].fillna(data['CSA'].mode()[0])
data['CBSA'] = data['CBSA'].fillna(data['CBSA'].mode()[0])
```

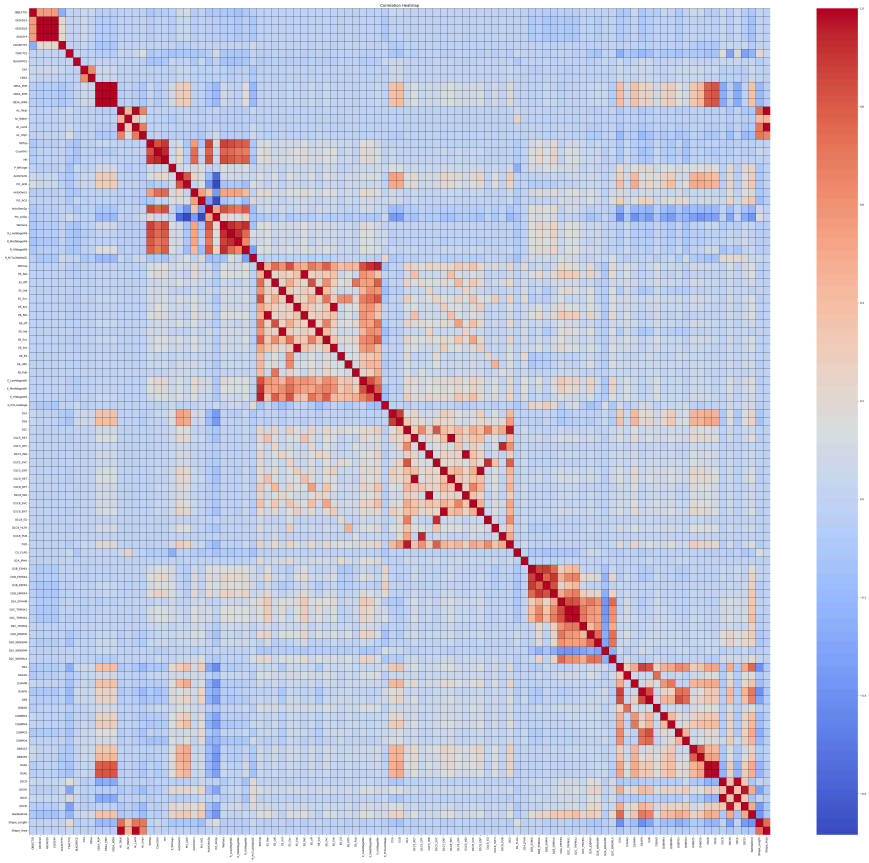
```
data.isnull().sum()
```

```
OBJECTID        0
GEOID10         0
GEOID20         0
STATEFP         0
COUNTYFP        0
               ..
D5CE            0
D5CEI           0
NatWalkInd      0
Shape_Length    0
Shape_Area      0
Length: 101, dtype: int64
```

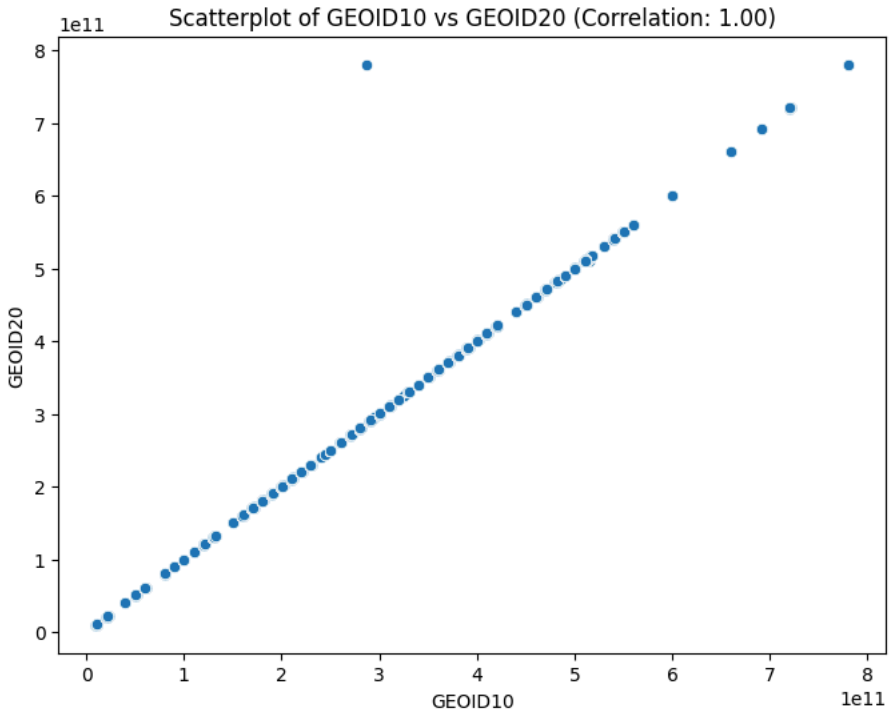
```
# Dataset summary
data_summary = {
    "Number of Entries": data.shape[0],
    "Number of Columns": data.shape[1],
    "Column Names": data.columns.tolist()
}

# Display data summary and missing values info
data_summary, data.info(), data.describe()
```

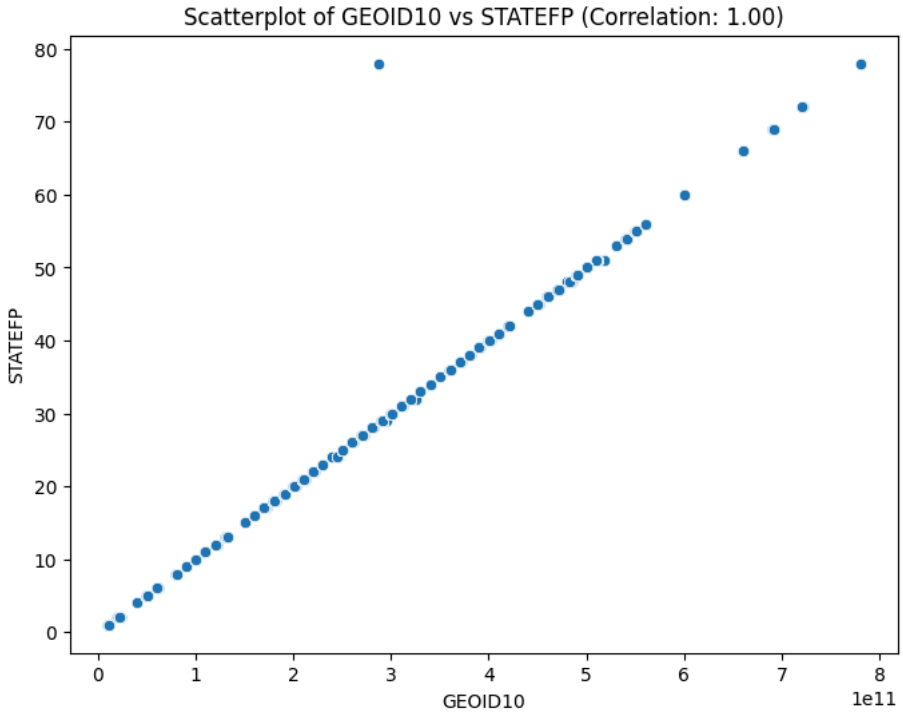
```
            OBJECTID       GEOID10       GEOID20        STATEFP  \
count  220740.000000  2.207400e+05  2.207400e+05  220740.000000
mean   110370.500000  2.870894e+11  2.870915e+11      28.623190
std     63722.293548  1.640742e+11  1.640774e+11      16.386075
min         1.000000  1.001020e+10  1.001020e+10       1.000000
25%     55185.750000  1.312100e+11  1.312100e+11      13.000000
50%    110370.500000  2.901900e+11  2.901900e+11      29.000000
75%    165555.250000  4.200350e+11  4.200350e+11      42.000000
max    220740.000000  7.803100e+11  7.803100e+11      78.000000

            COUNTYFP        TRACTCE       BLKGRPCE            CSA  \
count  220740.000000  220740.000000  220740.000000  220740.000000
mean       85.697449  262342.382110       2.221392     341.742661
std        98.818946  351403.837442       1.195155     109.565259
min         1.000000     100.000000       0.000000     104.000000
25%        29.000000   10303.000000       1.000000     278.000000
50%        61.000000   46298.000000       2.000000     341.742661
75%       109.000000  482503.000000       3.000000     408.000000
max       840.000000  993000.000000       9.000000     566.000000

                CBSA      CBSA_POP  ...         D4B050          D5AR  \
count  220740.000000  2.207400e+05  ...  220740.000000  2.207400e+05
mean    30514.836210  3.607329e+06  ...       0.066813  1.030618e+05
std     10545.755664  5.219925e+06  ...       0.228685  1.531004e+05
min     10100.000000  0.000000e+00  ...       0.000000  0.000000e+00
25%     19820.000000  2.053030e+05  ...       0.000000  9.260000e+03
50%     31080.000000  1.252890e+06  ...       0.000000  4.273400e+04
75%     38660.000000  4.673634e+06  ...       0.000000  1.251635e+05
max     49820.000000  1.931847e+07  ...       1.000000  1.220602e+06

                D5AE           D5CR          D5CRI           D5CE  \
count  220740.000000  220740.000000  220740.000000  220740.000000
mean    88536.521695       0.004195       0.431461       0.004195
std    130263.329763       0.009657       0.281503       0.009364
min         0.000000       0.000000       0.000000       0.000000
25%      9150.000000       0.000145       0.191683       0.000162
50%     38964.000000       0.000652       0.438312       0.000677
75%    105586.000000       0.003253       0.659545       0.003351
max    964355.000000       0.216832       1.000000       0.267951

               D5CEI     NatWalkInd  Shape_Length    Shape_Area
count  220740.000000  220740.000000  2.207400e+05  2.207400e+05
mean        0.494850       9.541628  1.655970e+04  4.466074e+07
std         0.290670       4.373952  3.830373e+04  6.430513e+08
min         0.000000       1.000000  2.685713e+02  4.435890e+03
```

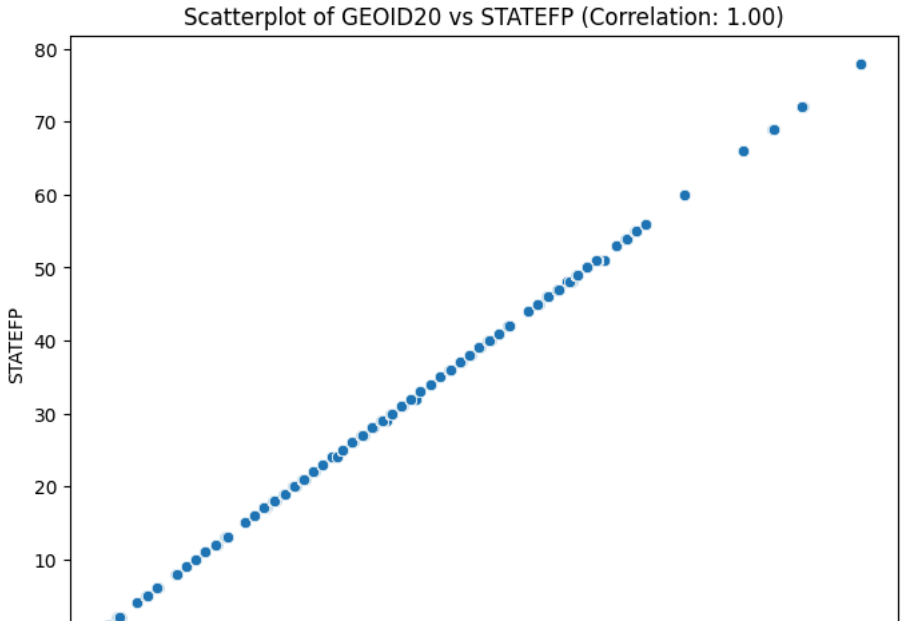
```
# Create the correlation heatmap with line spaces and labels
plt.figure(figsize=(50, 45))
sns.heatmap(data.corr(), annot=False, cmap="coolwarm", fmt=".2f", linewidths=0.5, linecolor="black")
plt.yticks(rotation=0)
plt.xticks(rotation=90)
plt.title('Correlation Heatmap', fontsize=14)
plt.tight_layout()
plt.show()
```

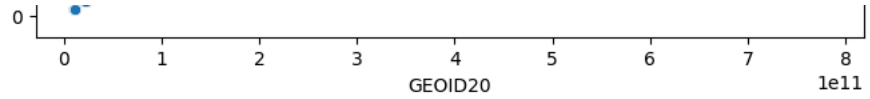


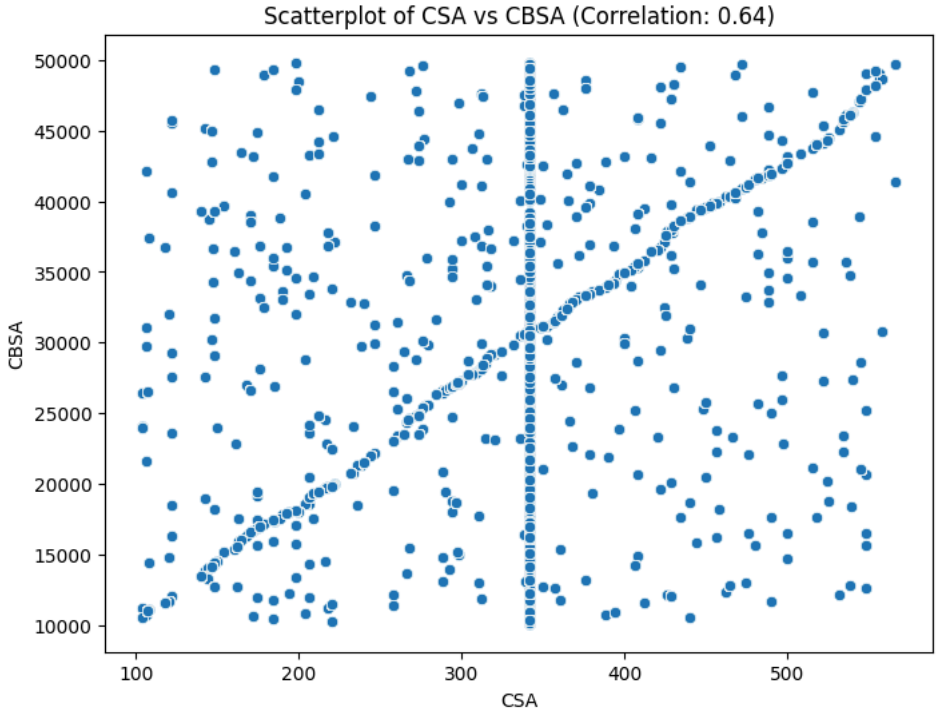
```
import seaborn as sns
import matplotlib.pyplot as plt

# Calculate the updated correlation matrix
updated_correlation_matrix = data.corr()

# Identify highly correlated variables (threshold |r| > 0.8)
high_corr_vars = updated_correlation_matrix.abs().stack().reset_index()
high_corr_vars.columns = ['Variable1', 'Variable2', 'Correlation']
high_corr_vars = high_corr_vars[high_corr_vars['Variable1'] != high_corr_vars['Variable2']]
high_corr_vars = high_corr_vars[high_corr_vars['Correlation'] > 0.6]
high_corr_vars = high_corr_vars[high_corr_vars['Correlation'] < 1]  # Exclude perfect correlation
high_corr_pairs = high_corr_vars.drop_duplicates(subset=['Correlation'])


# Create scatterplots for each pair of highly correlated variables
for _, row in high_corr_pairs.iterrows():
    var1 = row['Variable1']
    var2 = row['Variable2']
    plt.figure(figsize=(8, 6))
    sns.scatterplot(data=data, x=var1, y=var2)
    plt.title(f'Scatterplot of {var1} vs {var2} (Correlation: {row["Correlation"]:.2f})')
    plt.show()
```

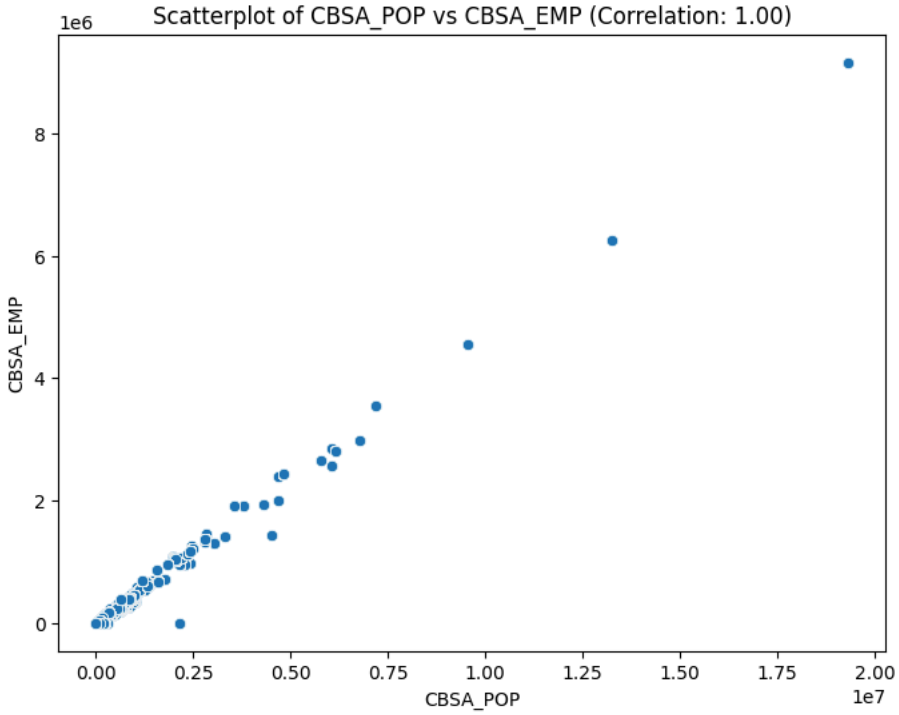

Scatterplot of GEOID10 vs GEOID20 (Correlation: 1.00)
1e11
GEOID20
GEOID10
1e11

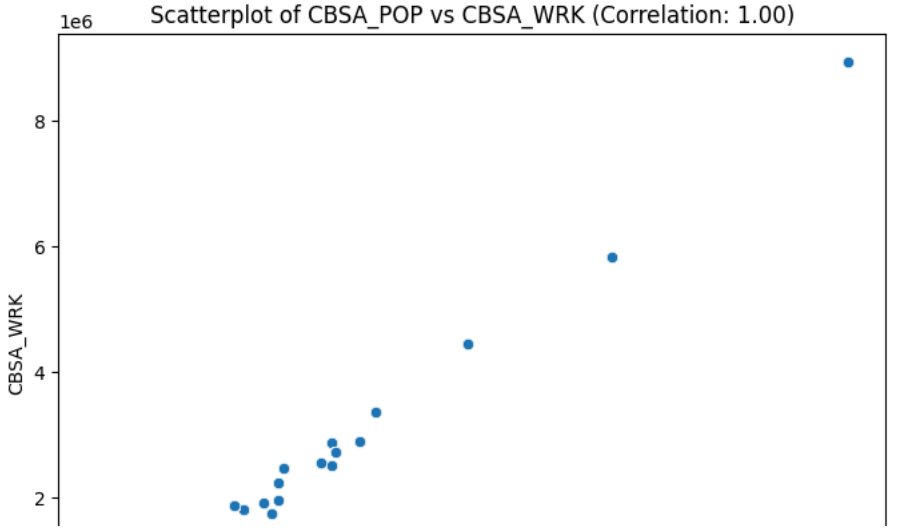

Scatterplot of GEOID10 vs STATEFP (Correlation: 1.00)
STATEFP
GEOID10
1e11

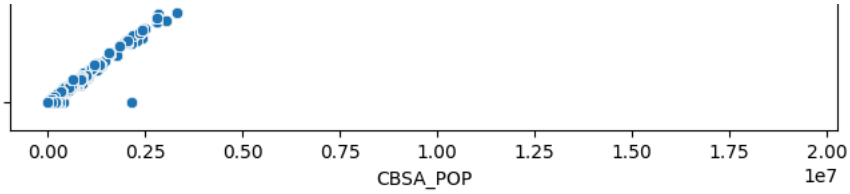

Scatterplot of GEOID20 vs STATEFP (Correlation: 1.00)
STATEFP

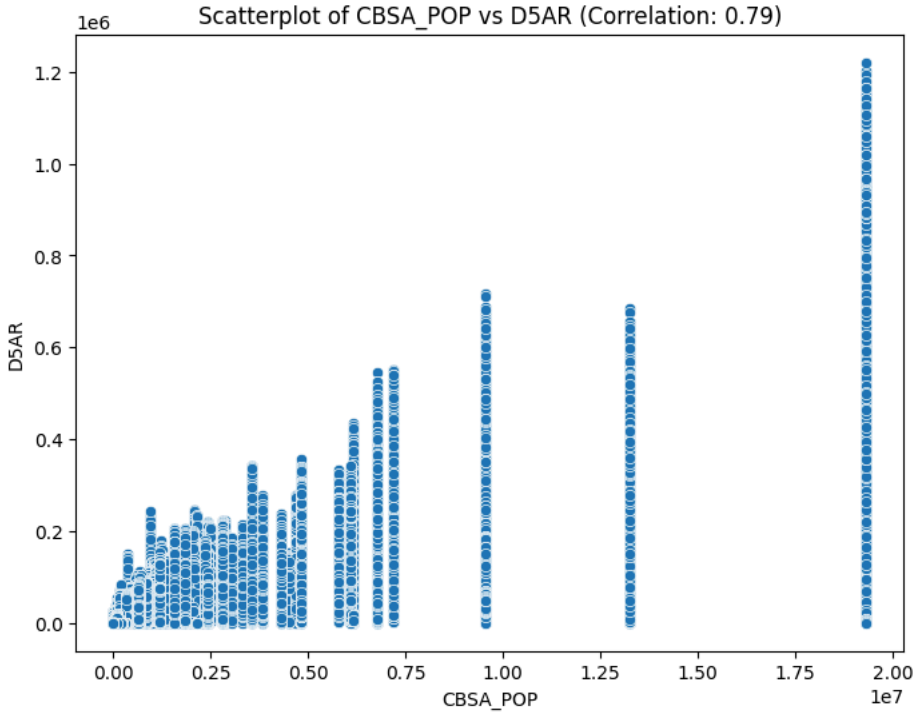

0
0
1
2
3
4
5
6
7
8
GEOID20
1e11

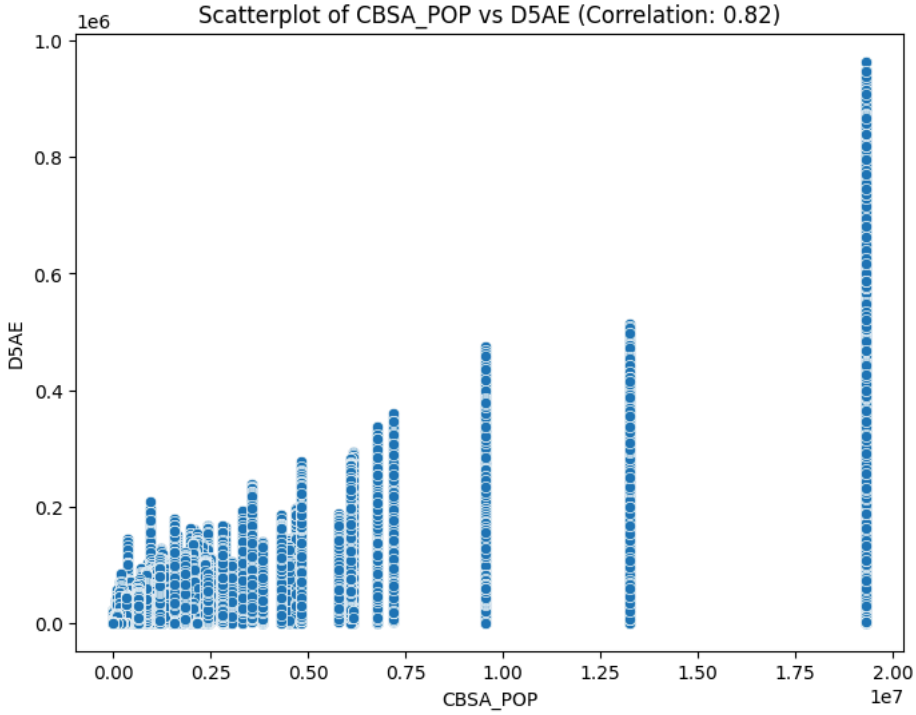

Scatterplot of CSA vs CBSA (Correlation: 0.64)
CBSA
CSA
50000
45000
40000
35000
30000
25000
20000
15000
10000
100
200
300
400
500

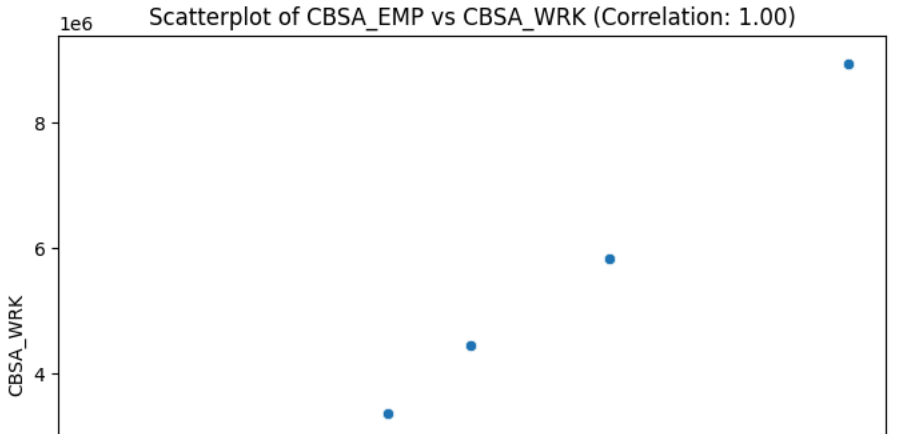

Scatterplot of CBSA_POP vs CBSA_EMP (Correlation: 1.00)
1e6
CBSA_EMP
CBSA_POP
1e7
8
6
4
2
0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00

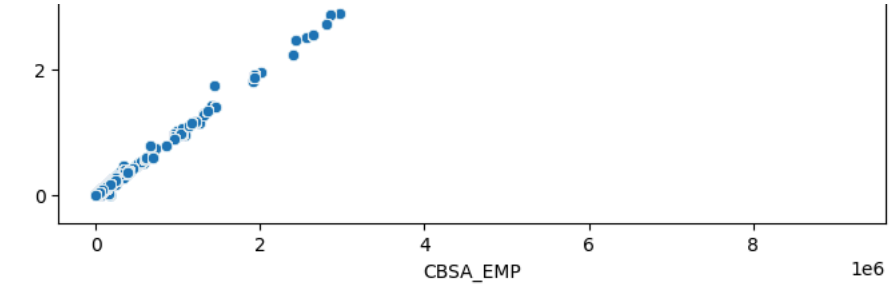

Scatterplot of CBSA_POP vs CBSA_WRK (Correlation: 1.00)
1e6
CBSA_WRK
8
6
4
2

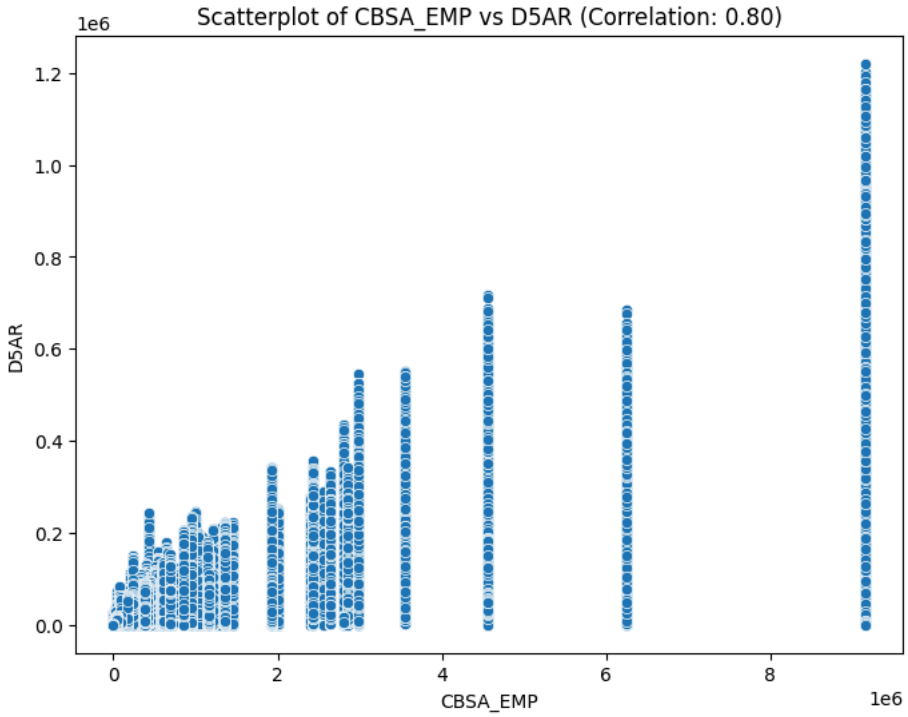

0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
1e7
CBSA_POP

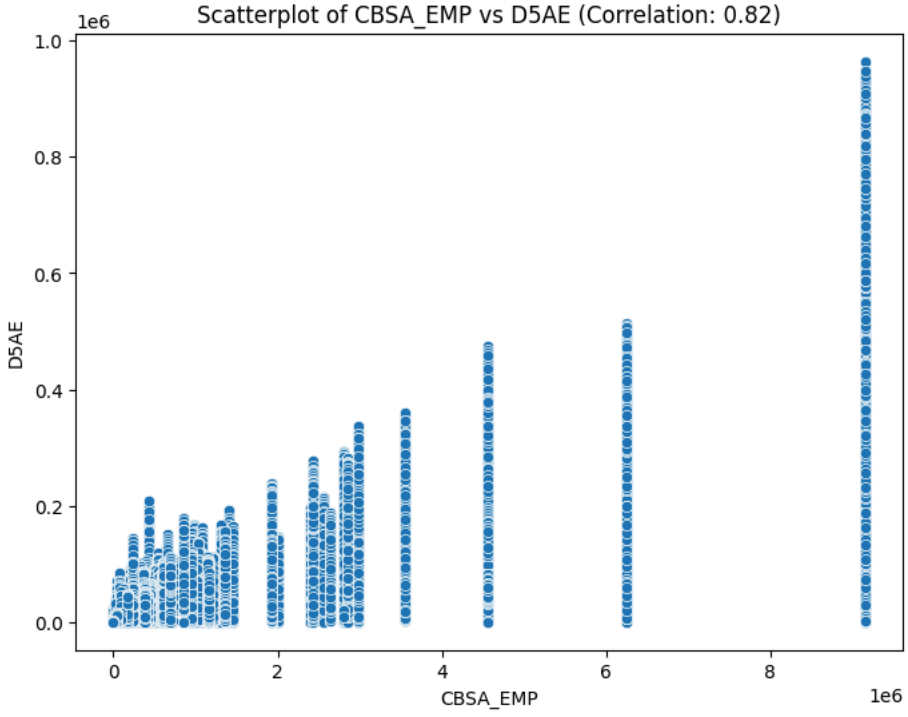

Scatterplot of CBSA_POP vs D5AR (Correlation: 0.79)
1e6
1.2
1.0
0.8
0.6
0.4
0.2
0.0
D5AR
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
1e7
CBSA_POP

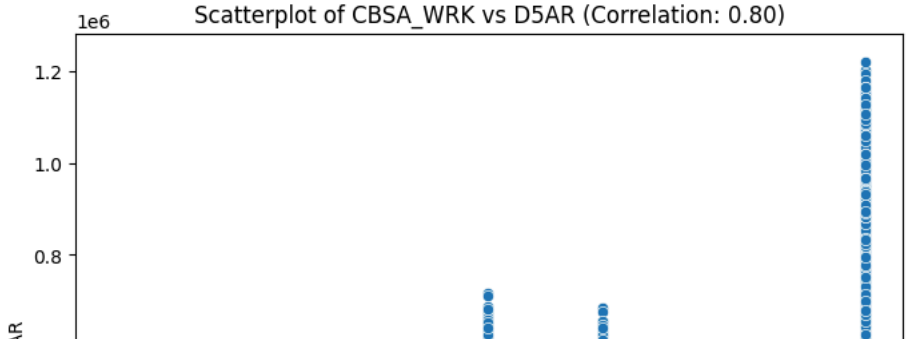

Scatterplot of CBSA_POP vs D5AE (Correlation: 0.82)
1e6
1.0
0.8
0.6
0.4
0.2
0.0
D5AE
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
1e7
CBSA_POP

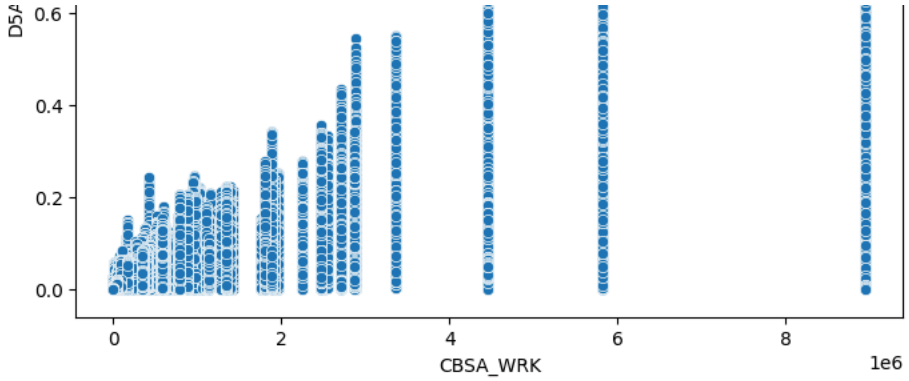

Scatterplot of CBSA_EMP vs CBSA_WRK (Correlation: 1.00)
1e6
8
6
4
CBSA_WRK

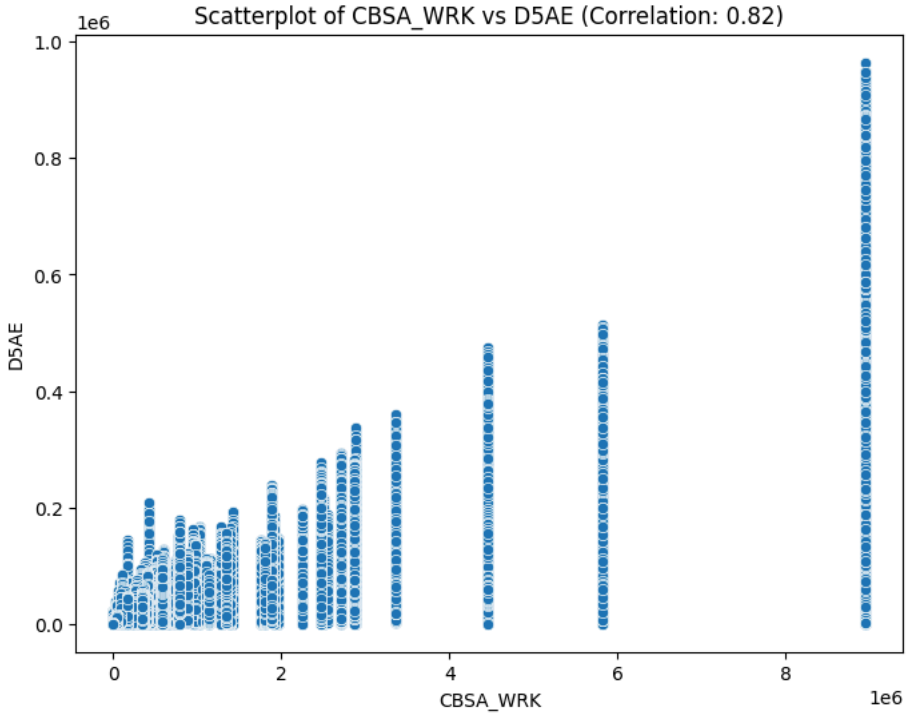

2
0
0
2
4
6
8
CBSA_EMP
1e6

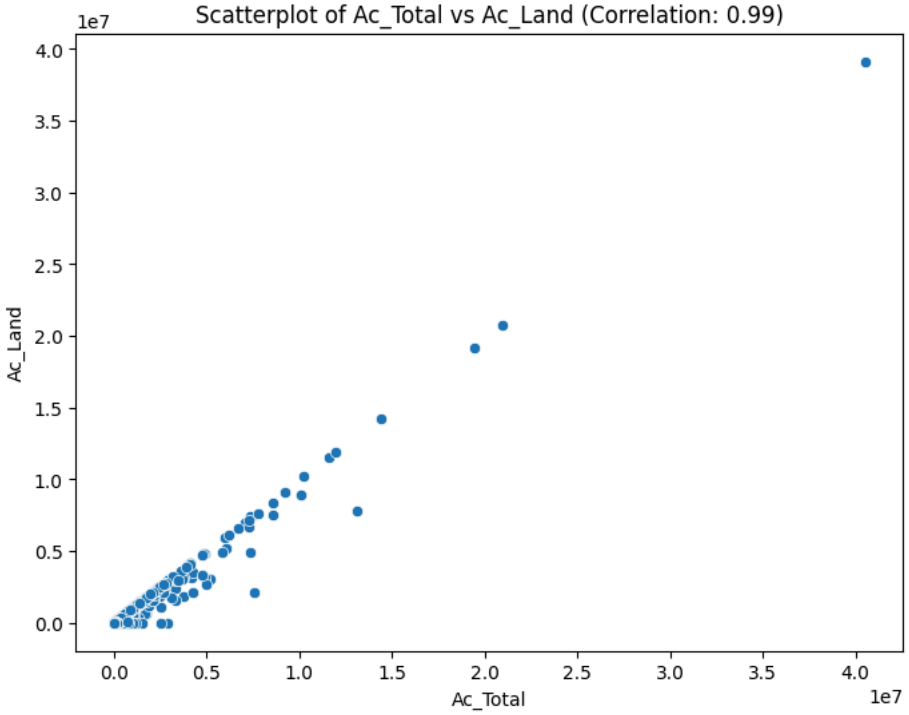

Scatterplot of CBSA_EMP vs D5AR (Correlation: 0.80)
1e6
1.2
1.0
0.8
0.6
0.4
0.2
0.0
D5AR
0
2
4
6
8
CBSA_EMP
1e6

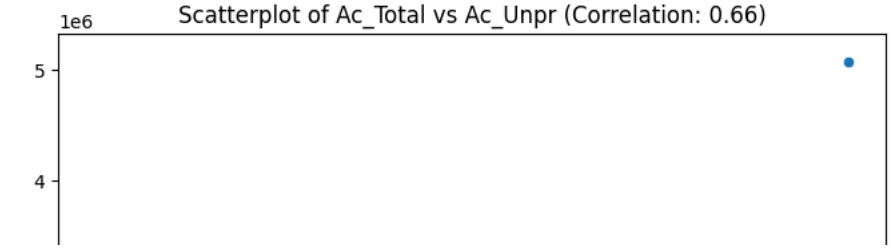

Scatterplot of CBSA_EMP vs D5AE (Correlation: 0.82)
1e6
1.0
0.8
0.6
0.4
0.2
0.0
D5AE
0
2
4
6
8
CBSA_EMP
1e6

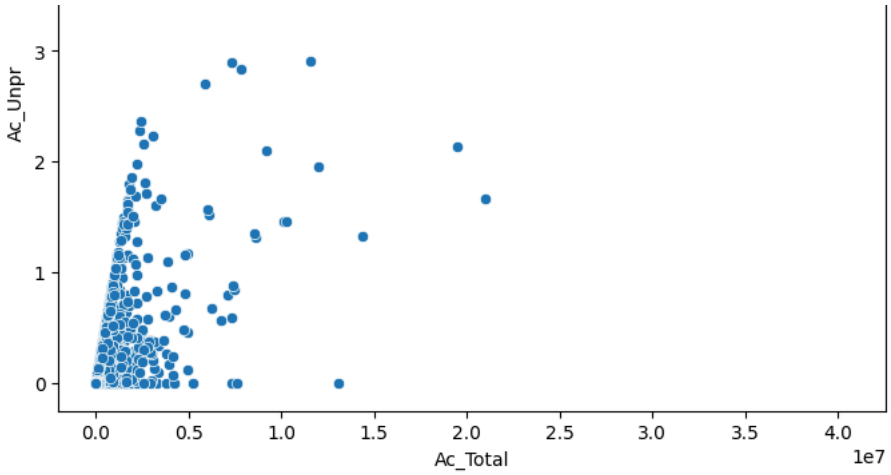

Scatterplot of CBSA_WRK vs D5AR (Correlation: 0.80)
1e6
1.2
1.0
0.8

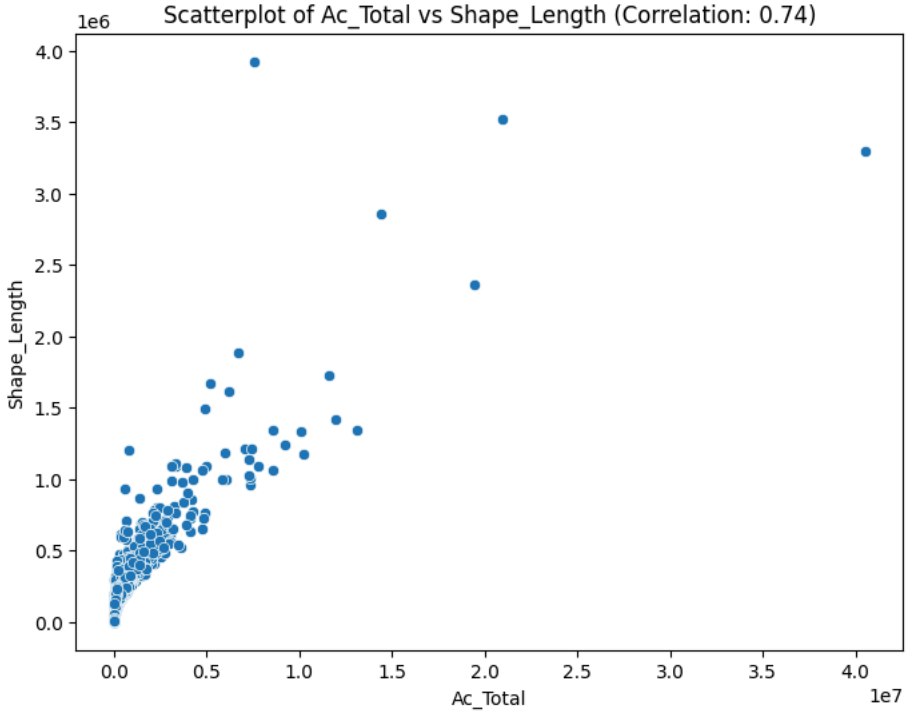

D5A
0.6
0.4
0.2
0.0
0
2
4
6
8
CBSA_WRK
1e6

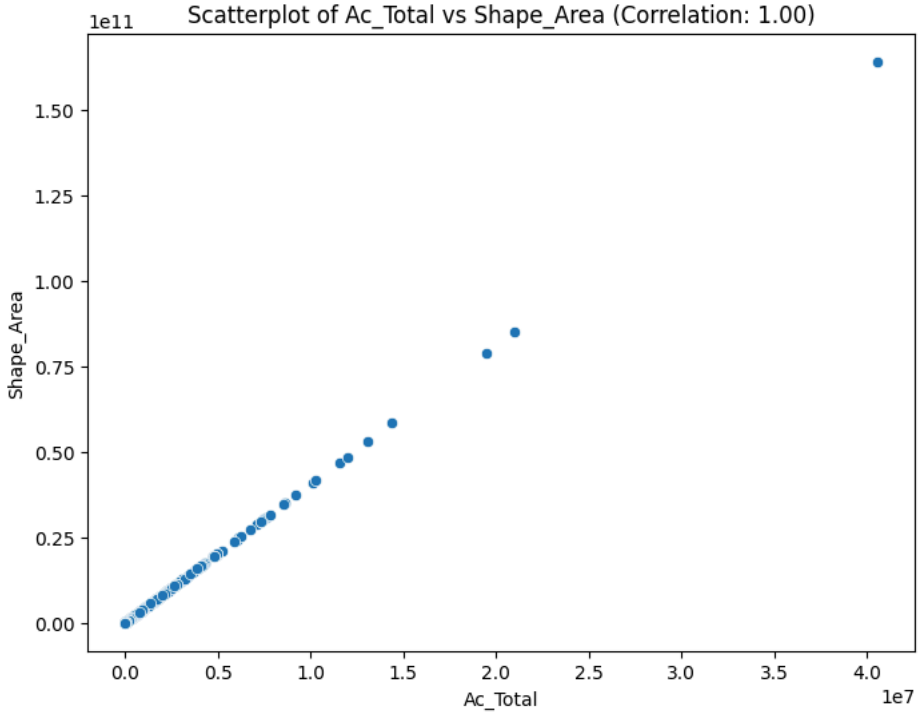

Scatterplot of CBSA_WRK vs D5AE (Correlation: 0.82)
1e6
1.0
0.8
0.6
0.4
0.2
0.0
D5AE
0
2
4
6
8
CBSA_WRK
1e6

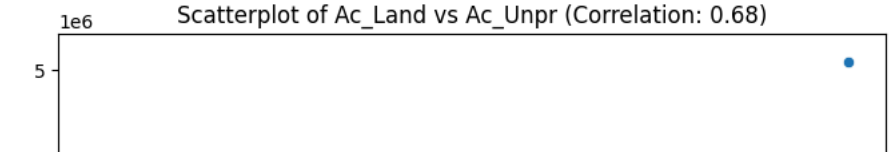

Scatterplot of Ac_Total vs Ac_Land (Correlation: 0.99)
1e7
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Ac_Land
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Ac_Total
1e7

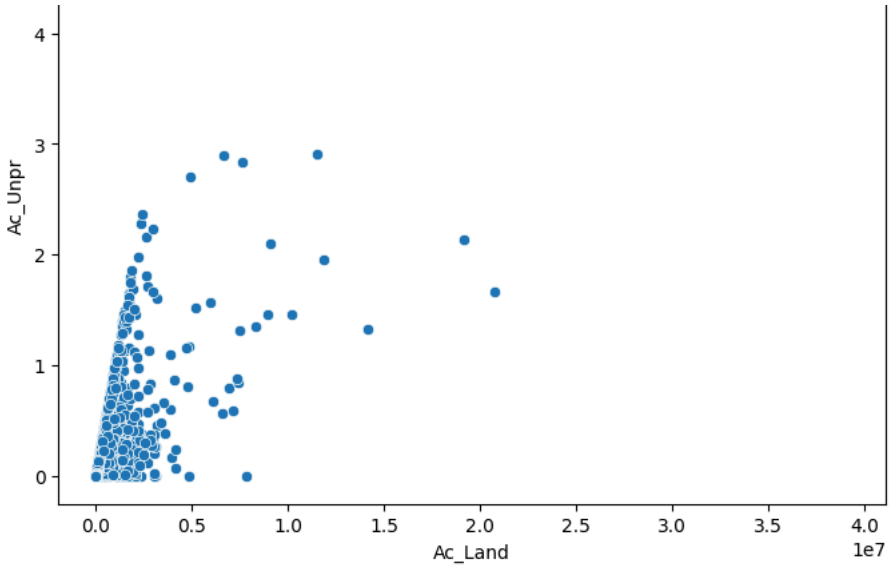

Scatterplot of Ac_Total vs Ac_Unpr (Correlation: 0.66)
1e6
5
4

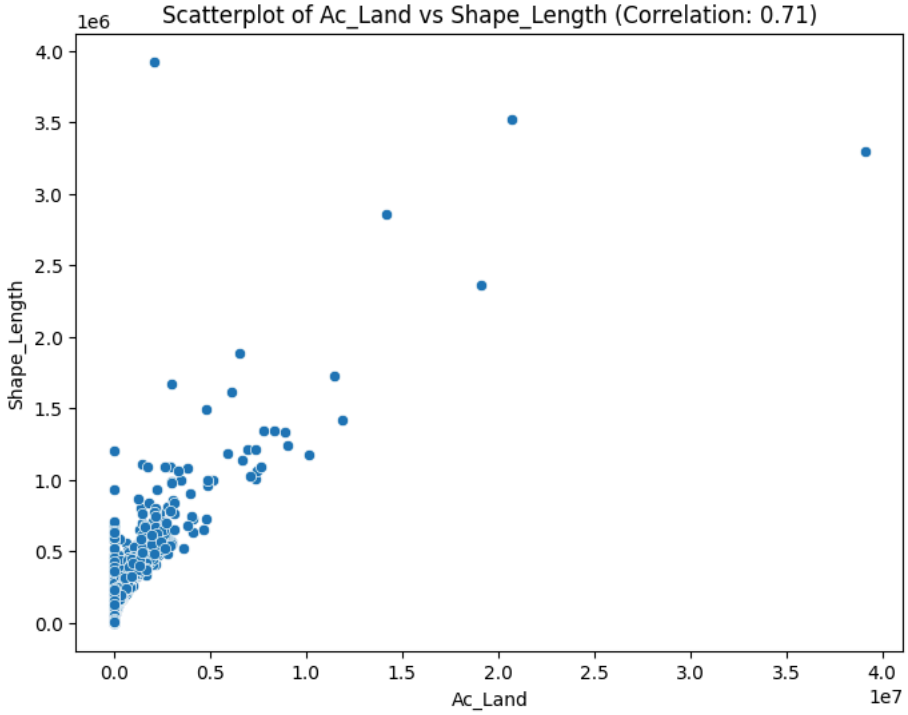

Ac_Unpr
Ac_Total
1e7

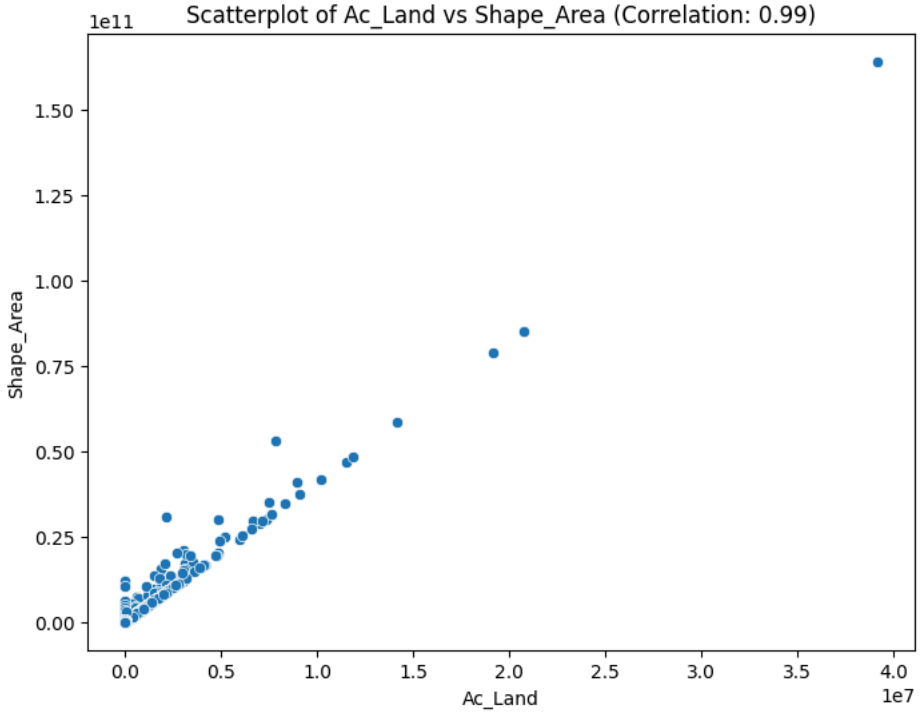

Scatterplot of Ac_Total vs Shape_Length (Correlation: 0.74)
1e6
Shape_Length
Ac_Total
1e7

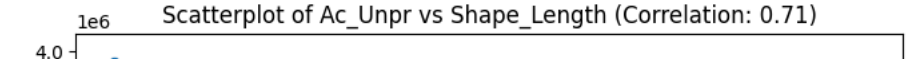

Scatterplot of Ac_Total vs Shape_Area (Correlation: 1.00)
1e11
Shape_Area
Ac_Total
1e7

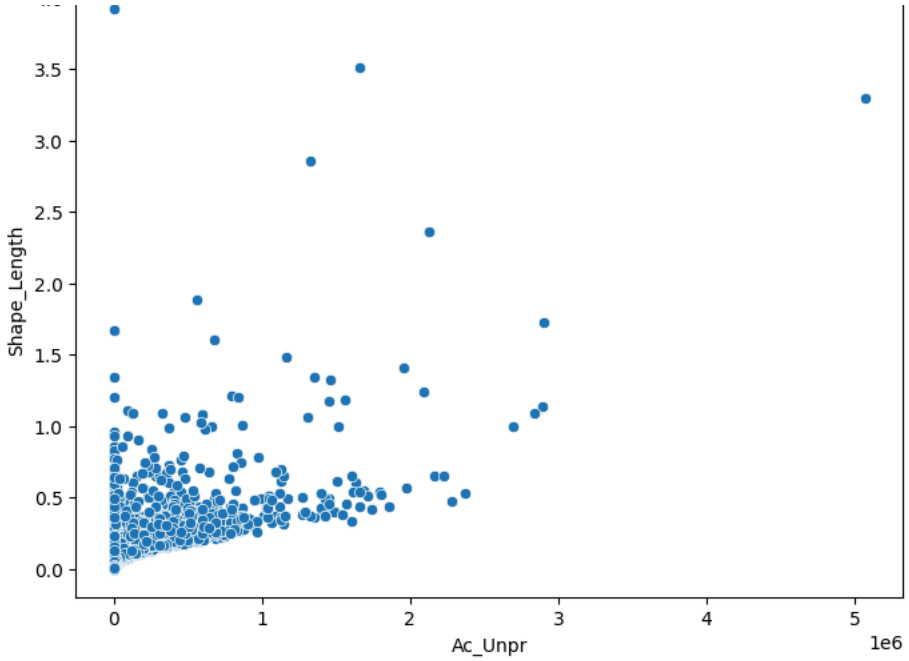

Scatterplot of Ac_Land vs Ac_Unpr (Correlation: 0.68)
1e6

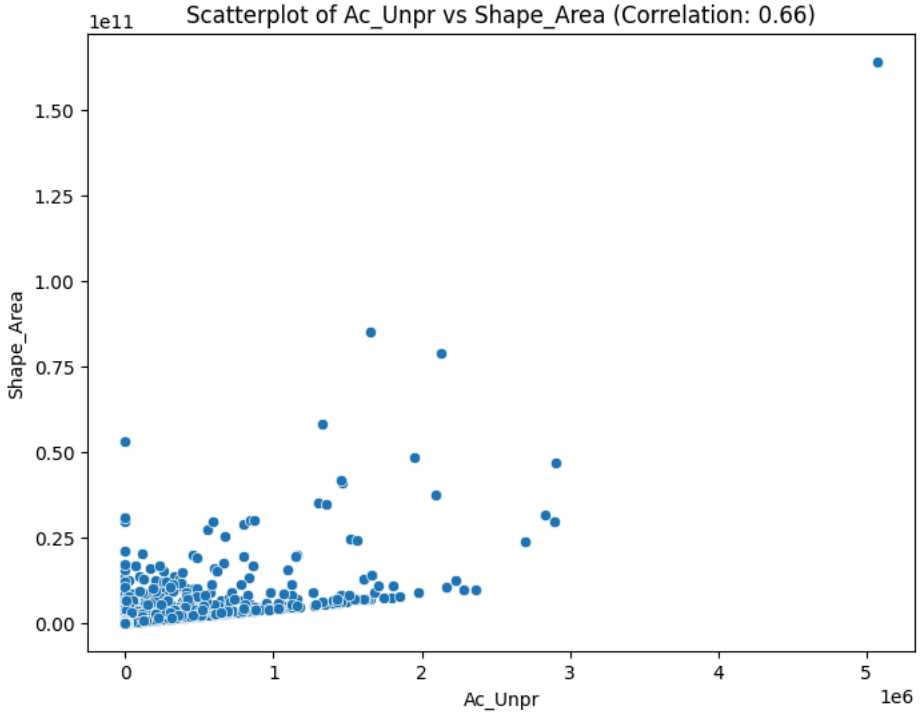

Ac_Unpr
Ac_Land
1e7

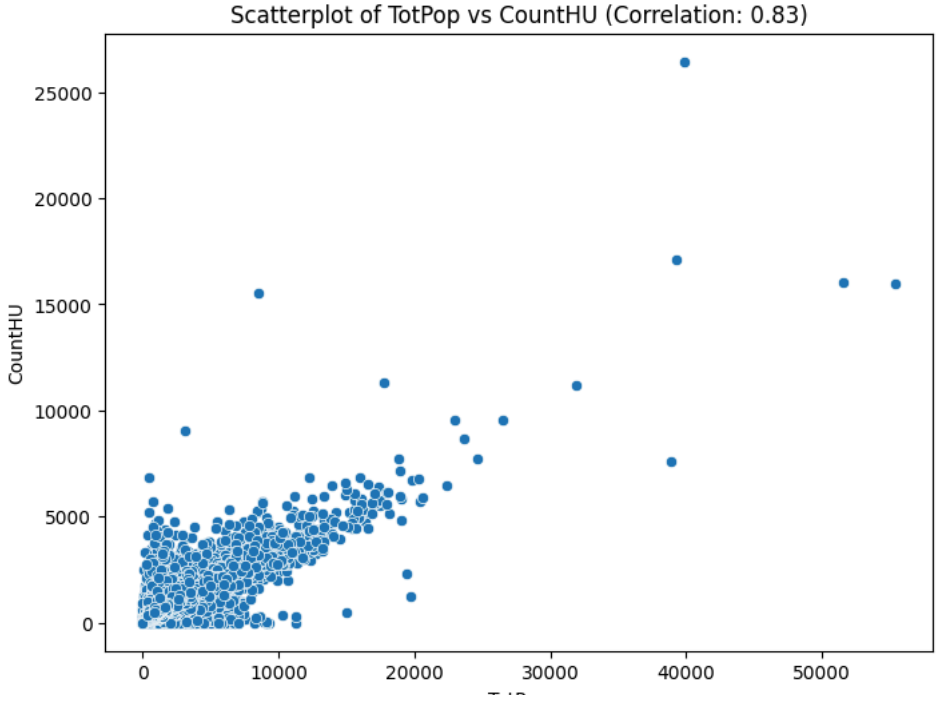

Scatterplot of Ac_Land vs Shape_Length (Correlation: 0.71)
1e6
Shape_Length
Ac_Land
1e7

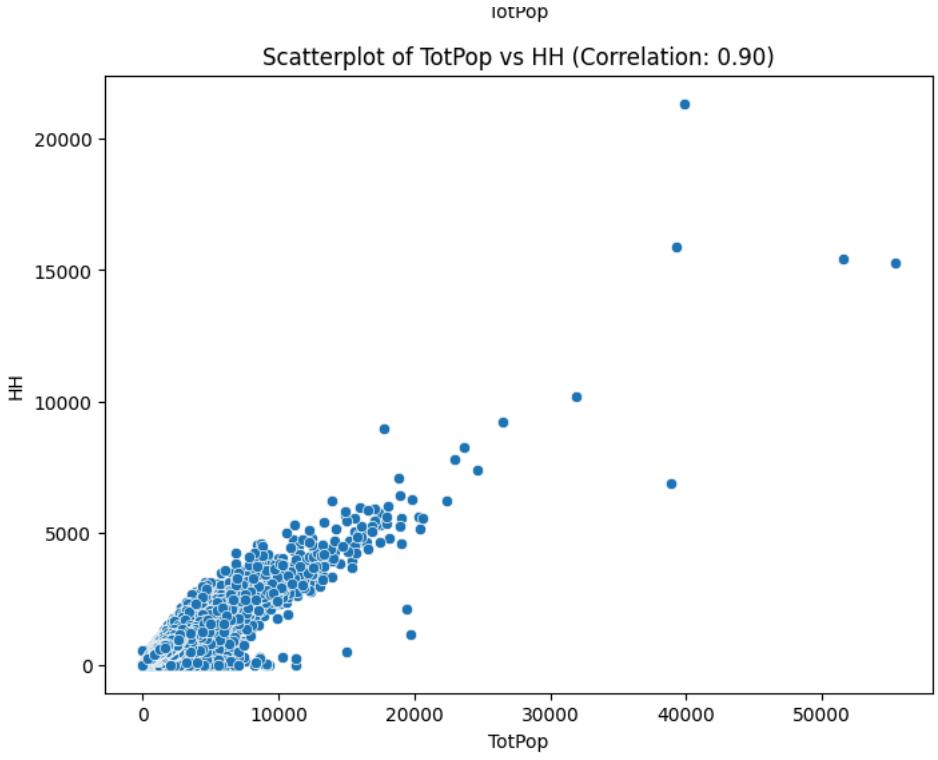

Scatterplot of Ac_Land vs Shape_Area (Correlation: 0.99)
1e11
Shape_Area
Ac_Land
1e7

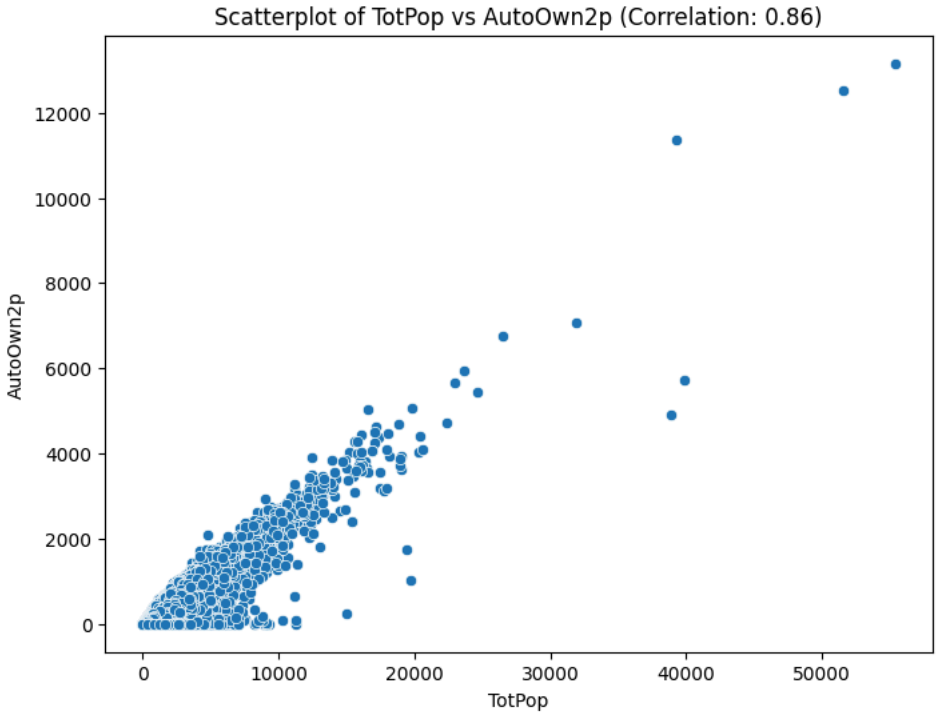

Scatterplot of Ac_Unpr vs Shape_Length (Correlation: 0.71)
1e6

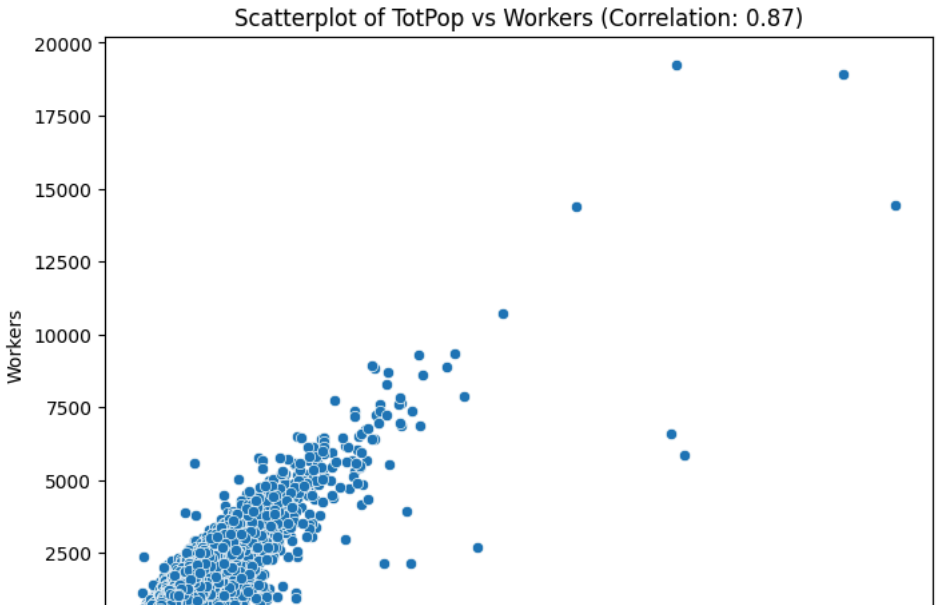

Shape_Length
Ac_Unpr
1e6

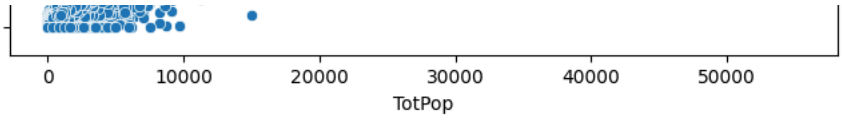

Scatterplot of Ac_Unpr vs Shape_Area (Correlation: 0.66)
1e11
Shape_Area
Ac_Unpr
1e6

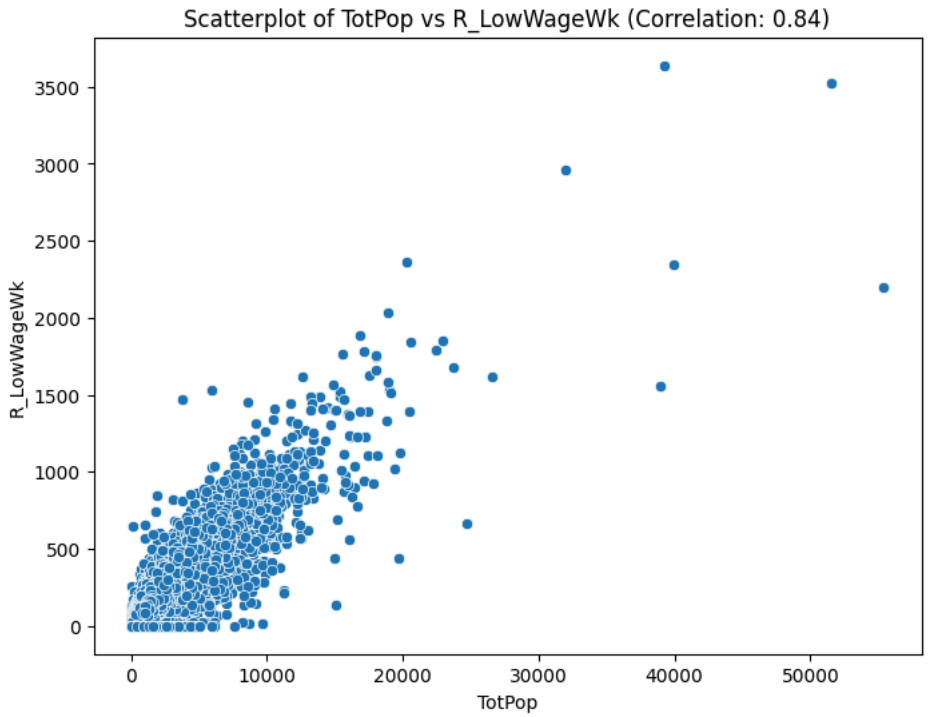

Scatterplot of TotPop vs CountHU (Correlation: 0.83)
CountHU

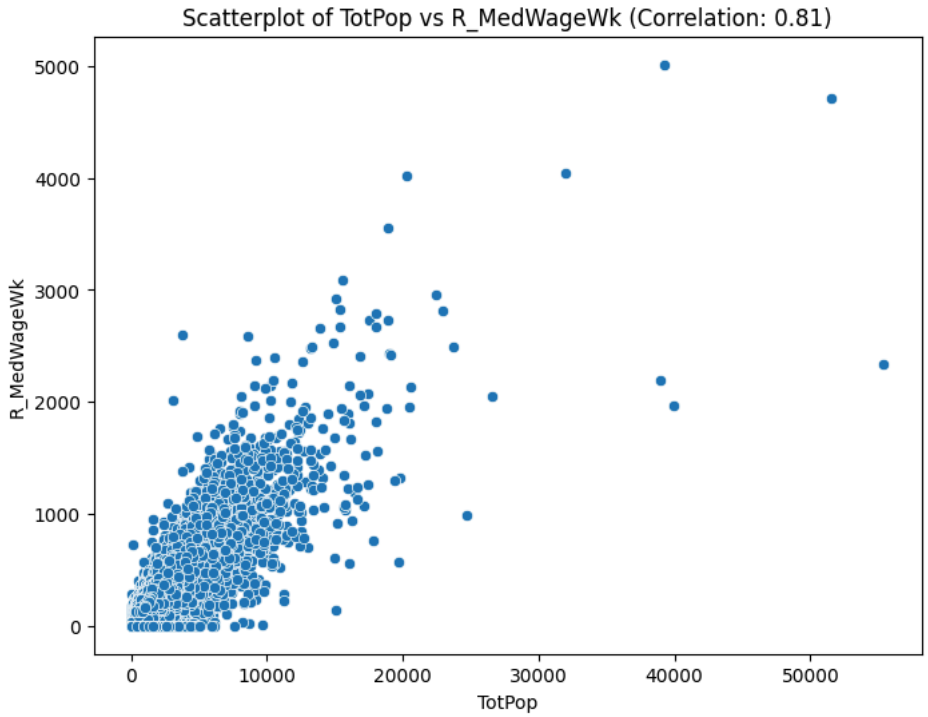

TotPop
Scatterplot of TotPop vs HH (Correlation: 0.90)
HH
TotPop
0
5000
10000
15000
20000
0
10000
20000
30000
40000
50000

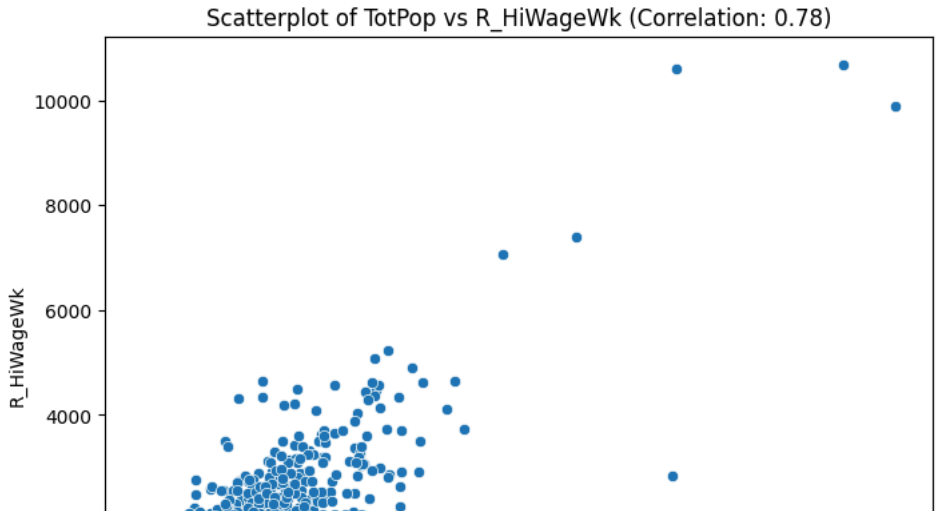

Scatterplot of TotPop vs AutoOwn2p (Correlation: 0.86)
AutoOwn2p
TotPop
0
2000
4000
6000
8000
10000
12000
0
10000
20000
30000
40000
50000

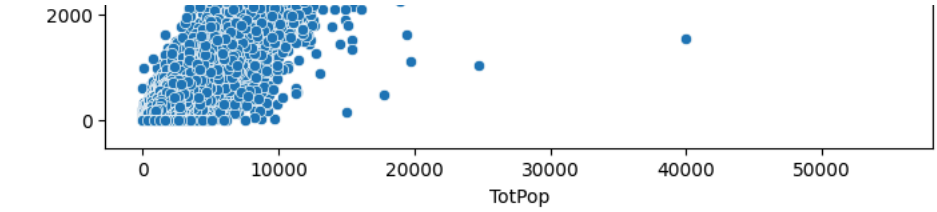

Scatterplot of TotPop vs Workers (Correlation: 0.87)
Workers
2500
5000
7500
10000
12500
15000
17500
20000

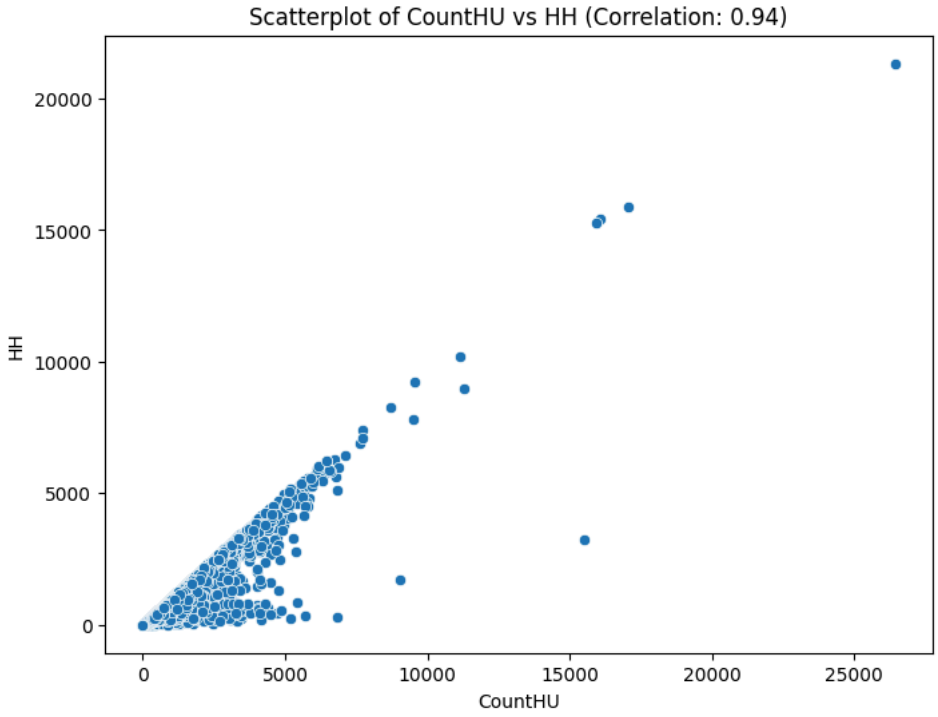

0
0
10000
20000
30000
40000
50000
TotPop

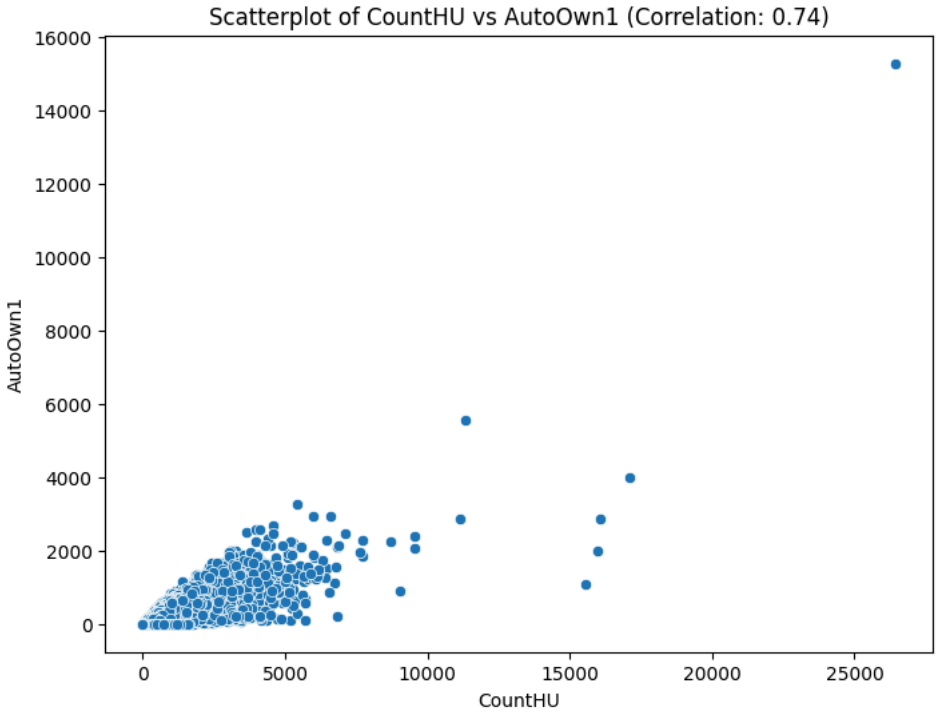

Scatterplot of TotPop vs R_LowWageWk (Correlation: 0.84)
R_LowWageWk
0
500
1000
1500
2000
2500
3000
3500
0
10000
20000
30000
40000
50000
TotPop

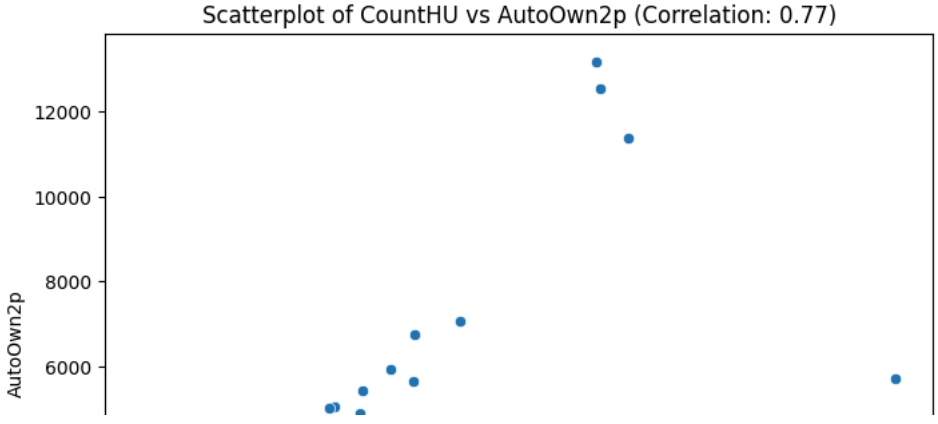

Scatterplot of TotPop vs R_MedWageWk (Correlation: 0.81)
R_MedWageWk
0
1000
2000
3000
4000
5000
0
10000
20000
30000
40000
50000
TotPop

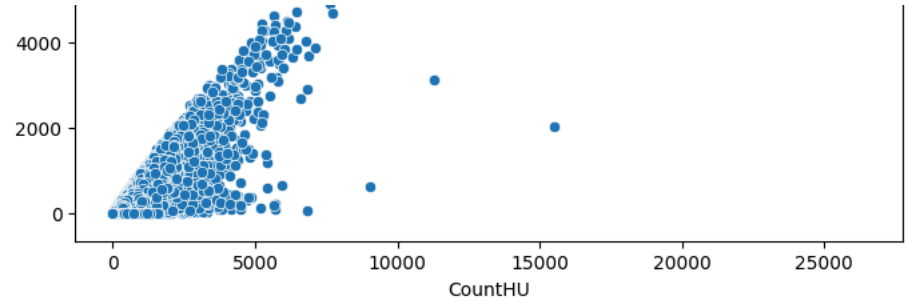

Scatterplot of TotPop vs R_HiWageWk (Correlation: 0.78)
R_HiWageWk
4000
6000
8000
10000

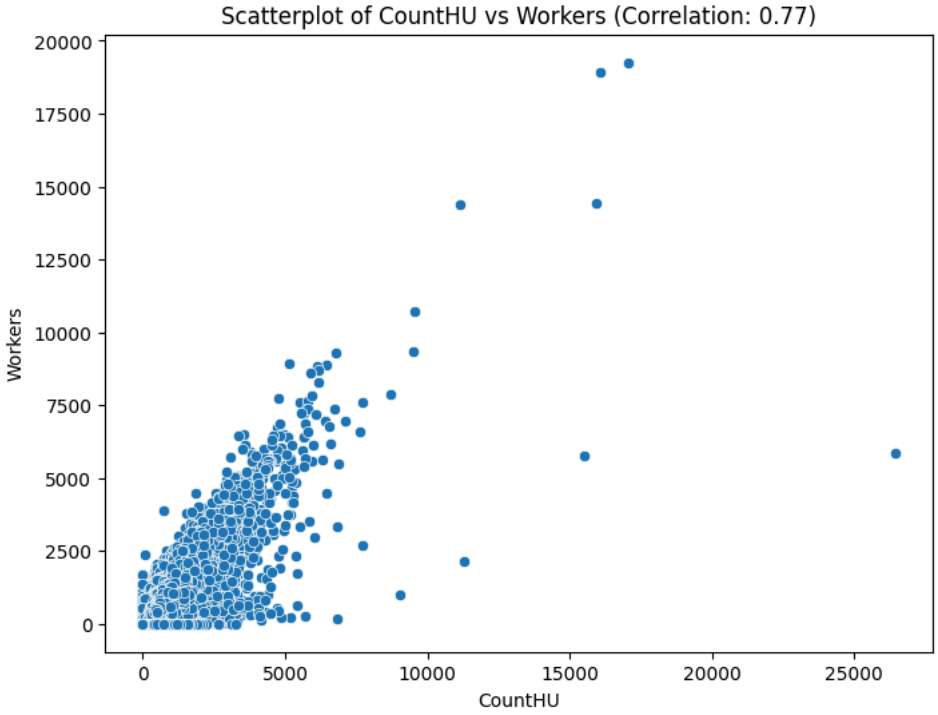

2000
0
0
10000
20000
30000
40000
50000
TotPop

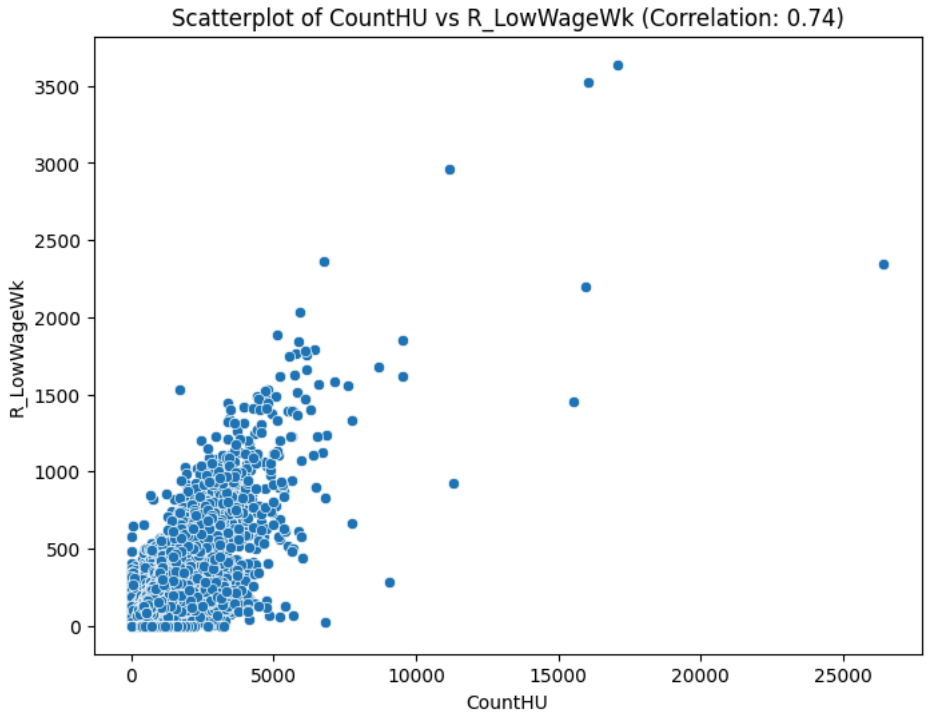

Scatterplot of CountHU vs HH (Correlation: 0.94)
20000
15000
10000
5000
0
HH
0
5000
10000
15000
20000
25000
CountHU

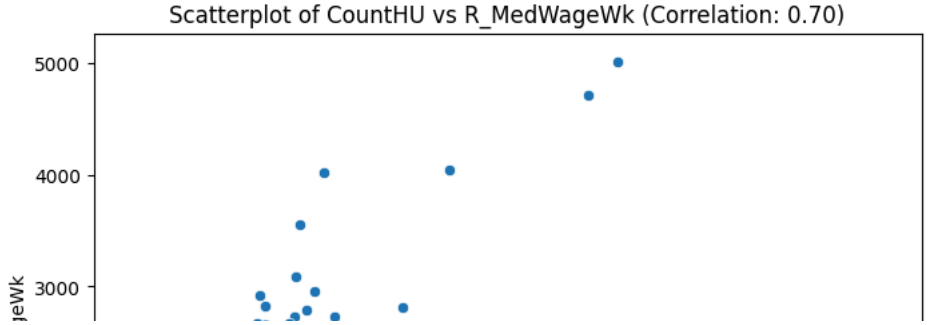

Scatterplot of CountHU vs AutoOwn1 (Correlation: 0.74)
16000
14000
12000
10000
8000
6000
4000
2000
0
AutoOwn1
0
5000
10000
15000
20000
25000
CountHU

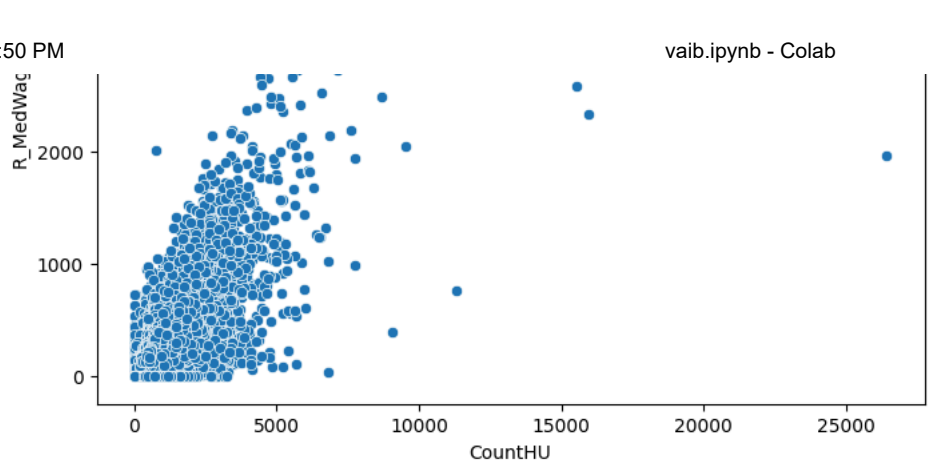

Scatterplot of CountHU vs AutoOwn2p (Correlation: 0.77)
12000
10000
8000
6000
AutoOwn2p

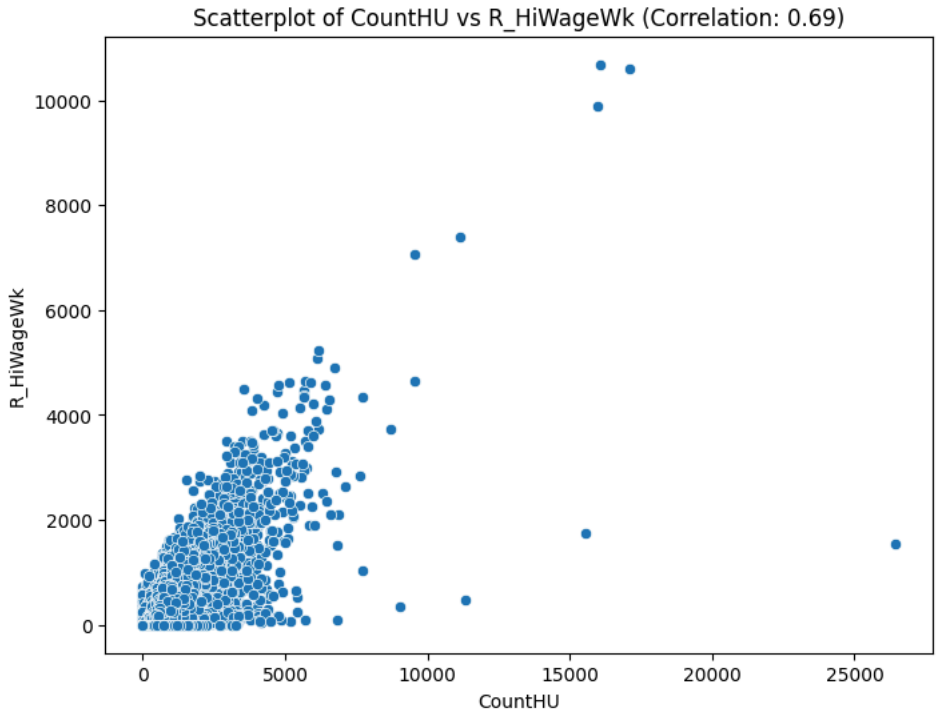

4000
2000
0
0
5000
10000
15000
20000
25000
CountHU

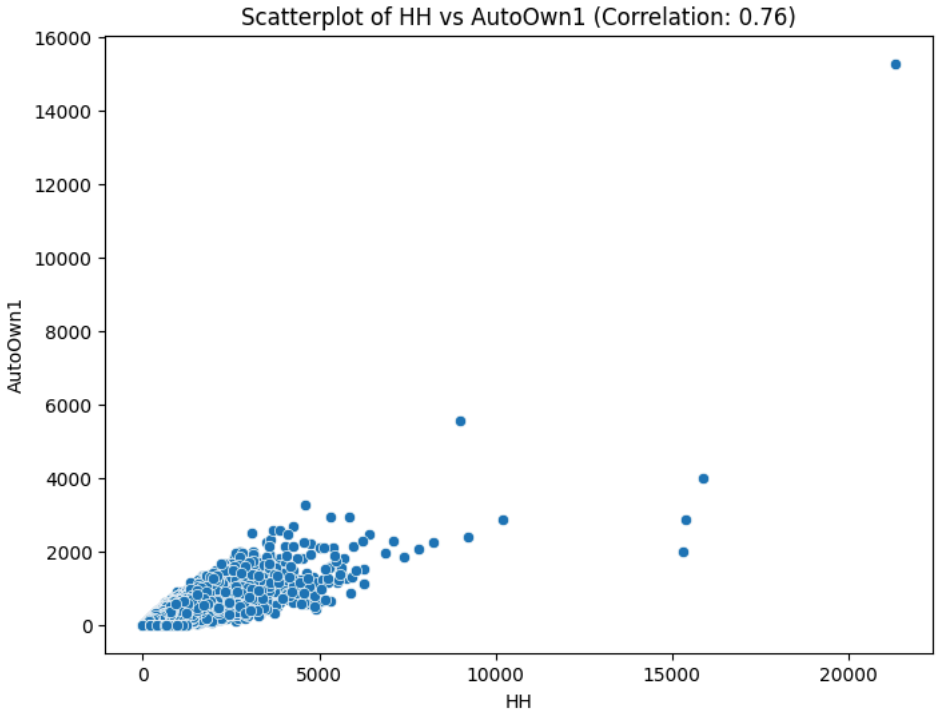

Scatterplot of CountHU vs Workers (Correlation: 0.77)
20000
17500
15000
12500
10000
7500
5000
2500
0
Workers
0
5000
10000
15000
20000
25000
CountHU

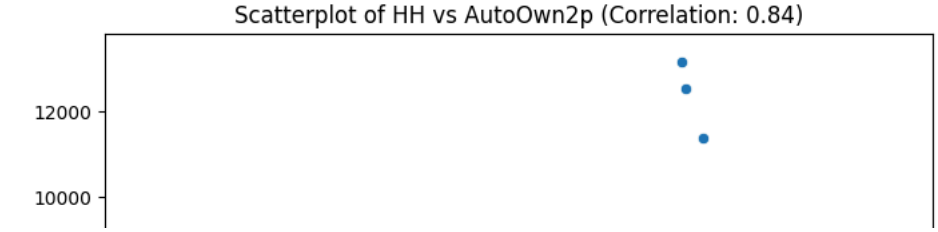

Scatterplot of CountHU vs R_LowWageWk (Correlation: 0.74)
3500
3000
2500
2000
1500
1000
500
0
R_LowWageWk
0
5000
10000
15000
20000
25000
CountHU

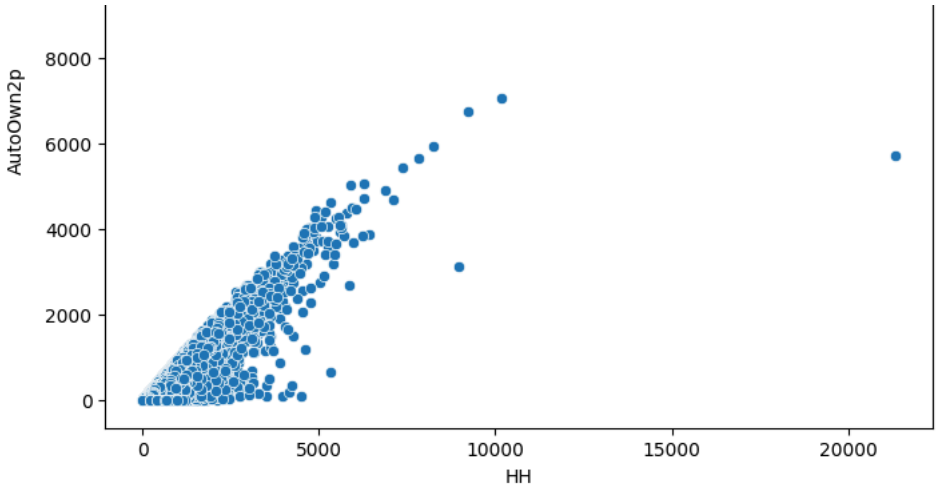

Scatterplot of CountHU vs R_MedWageWk (Correlation: 0.70)
5000
4000
3000

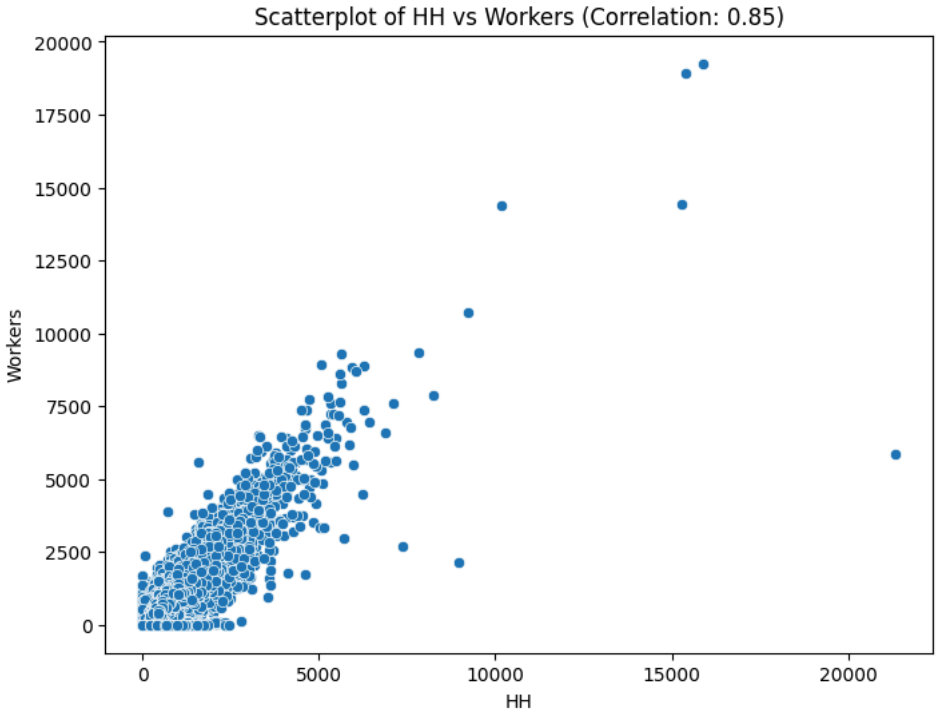

R_MedWag
2000
1000
0
0
5000
10000
15000
20000
25000
CountHU

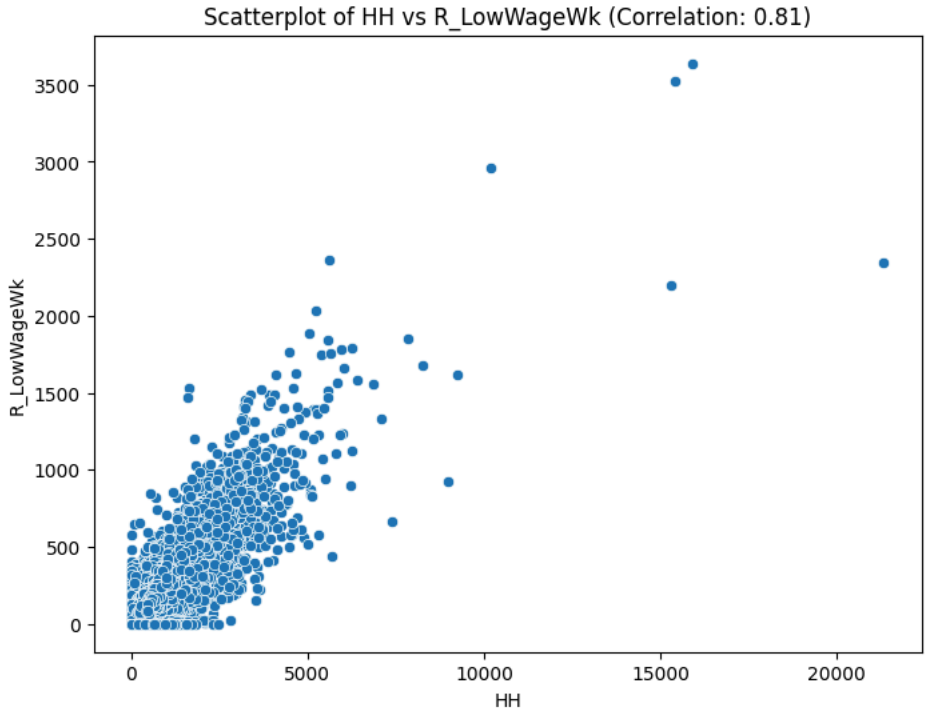

Scatterplot of CountHU vs R_HiWageWk (Correlation: 0.69)
R_HiWageWk
10000
8000
6000
4000
2000
0
0
5000
10000
15000
20000
25000
CountHU

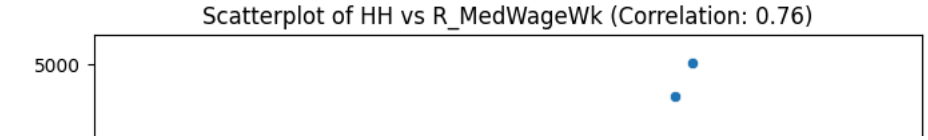

Scatterplot of HH vs AutoOwn1 (Correlation: 0.76)
AutoOwn1
16000
14000
12000
10000
8000
6000
4000
2000
0
0
5000
10000
15000
20000
HH

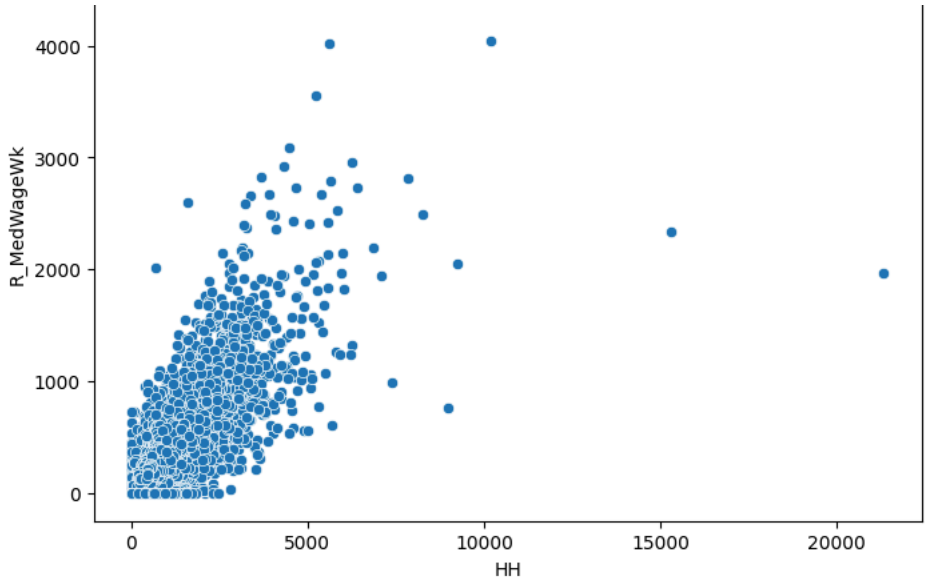

Scatterplot of HH vs AutoOwn2p (Correlation: 0.84)
12000
10000

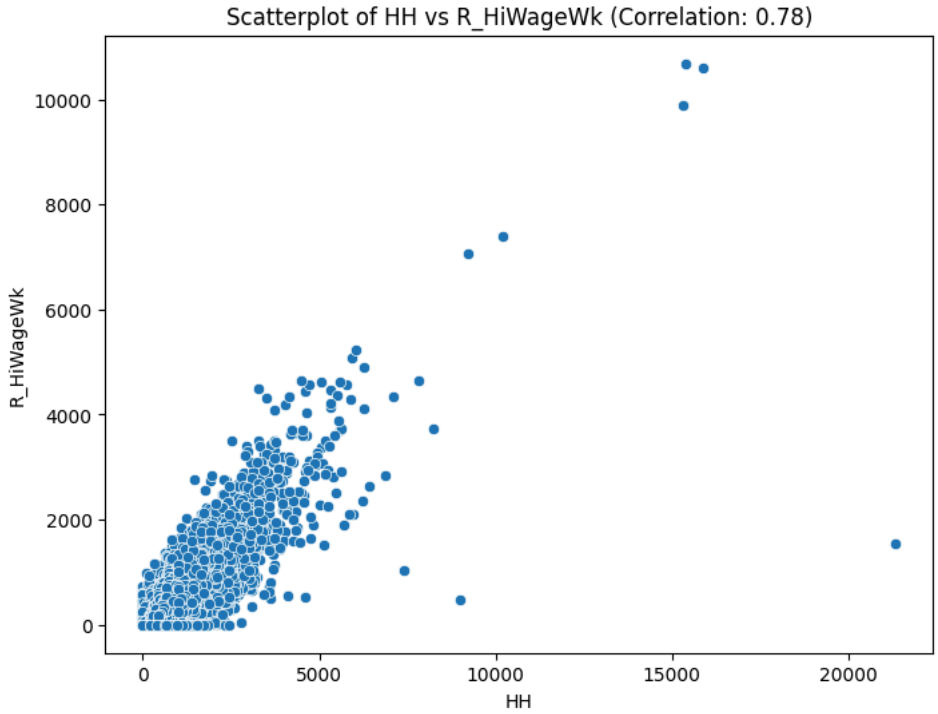

AutoOwn2p
HH
0
2000
4000
6000
8000
0
5000
10000
15000
20000

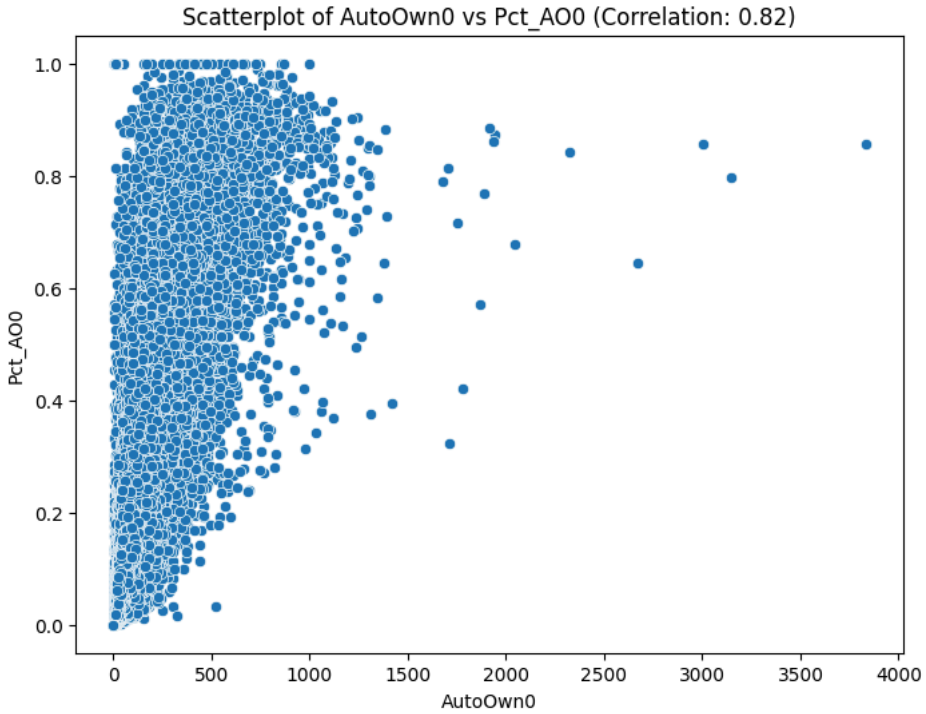

Scatterplot of HH vs Workers (Correlation: 0.85)
Workers
HH
0
2500
5000
7500
10000
12500
15000
17500
20000
0
5000
10000
15000
20000

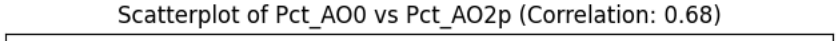

Scatterplot of HH vs R_LowWageWk (Correlation: 0.81)
R_LowWageWk
HH
0
500
1000
1500
2000
2500
3000
3500
0
5000
10000
15000
20000

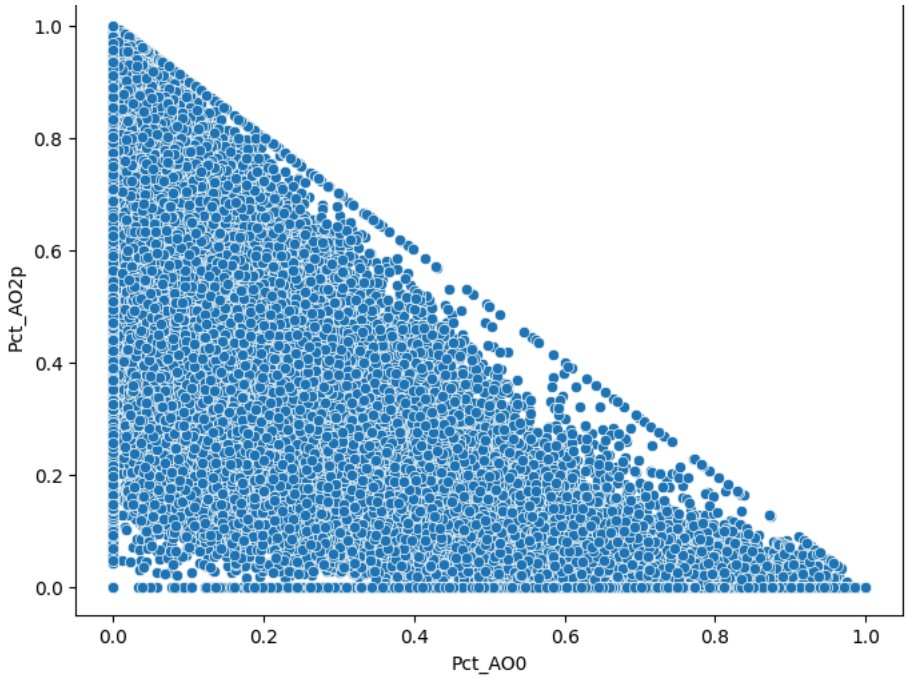

Scatterplot of HH vs R_MedWageWk (Correlation: 0.76)
5000


R_MedWageWk
HH
0
1000
2000
3000
4000
0
5000
10000
15000
20000

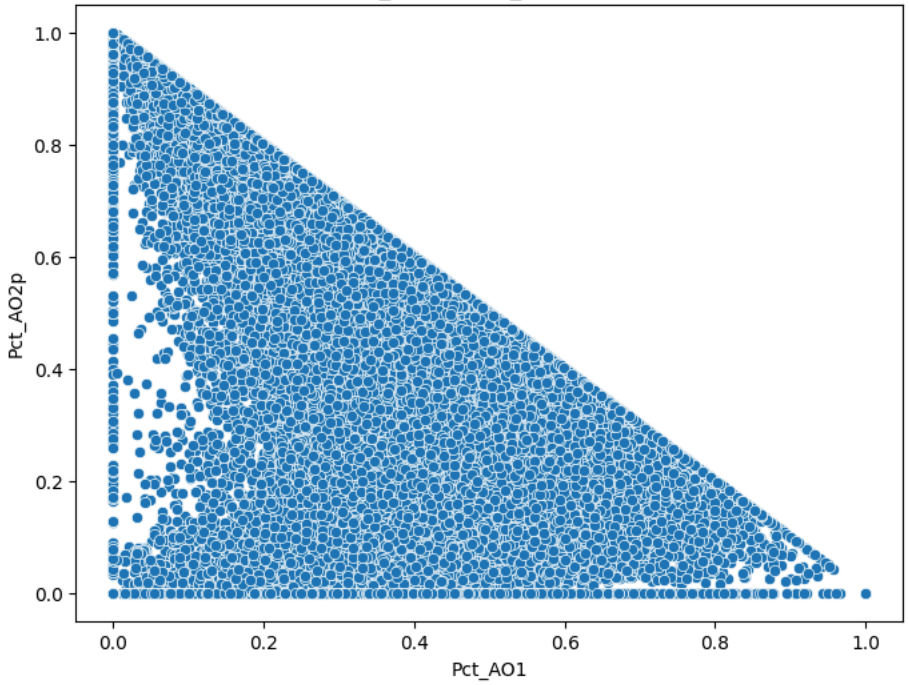

Scatterplot of HH vs R_HiWageWk (Correlation: 0.78)
R_HiWageWk
HH
0
2000
4000
6000
8000
10000
0
5000
10000
15000
20000


Scatterplot of AutoOwn0 vs Pct_AO0 (Correlation: 0.82)
Pct_AO0
AutoOwn0
0.0
0.2
0.4
0.6
0.8
1.0
0
500
1000
1500
2000
2500
3000
3500
4000

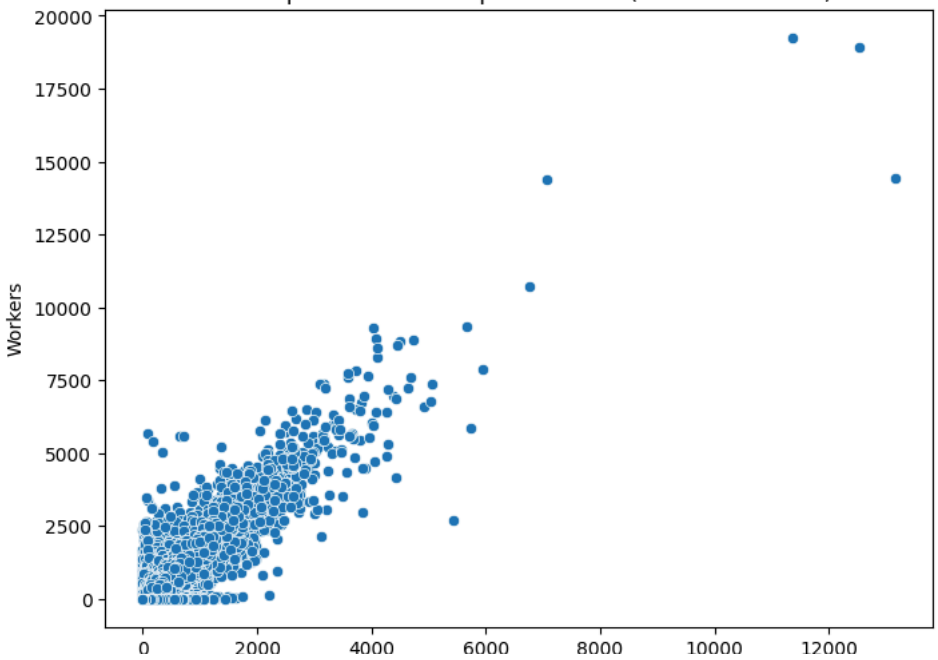

Scatterplot of Pct_AO0 vs Pct_AO2p (Correlation: 0.68)

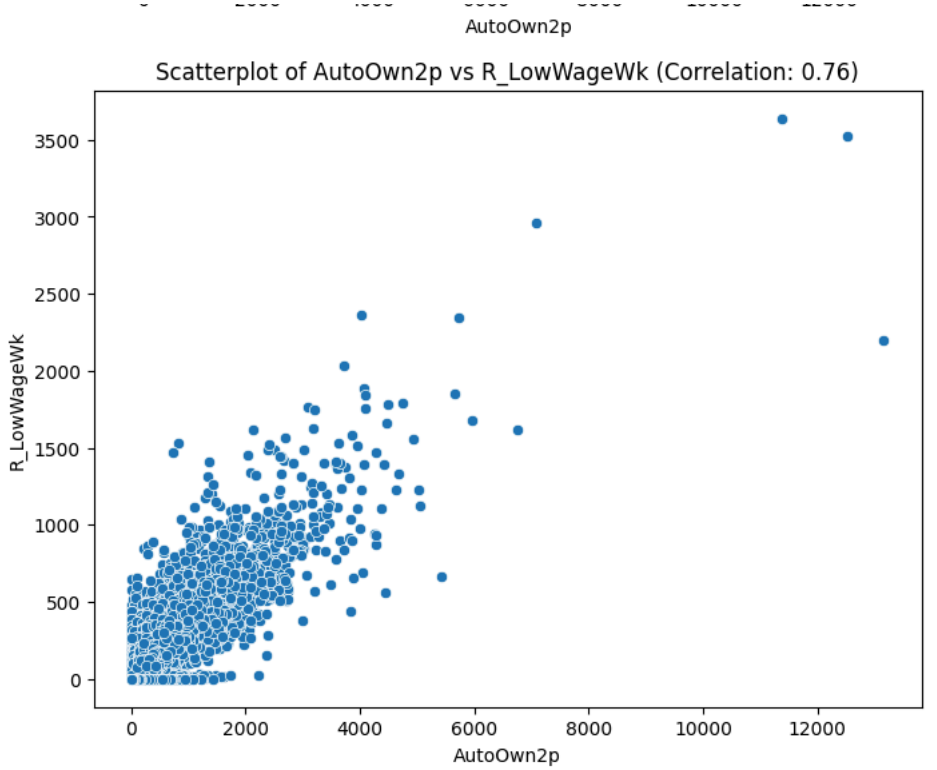


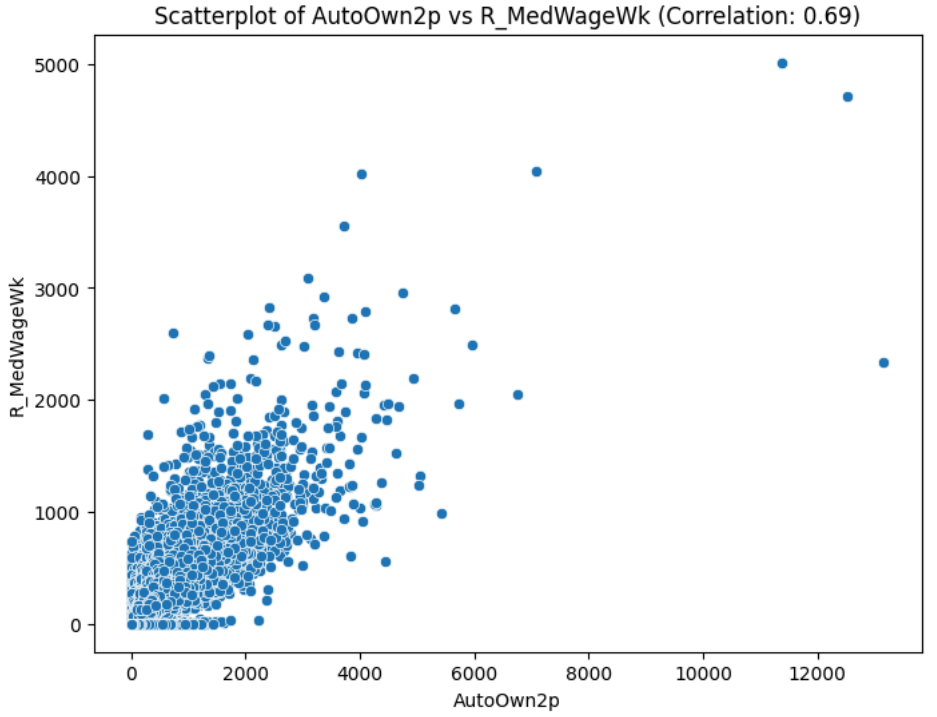
Scatterplot of Pct_AO1 vs Pct_AO2p (Correlation: 0.68)

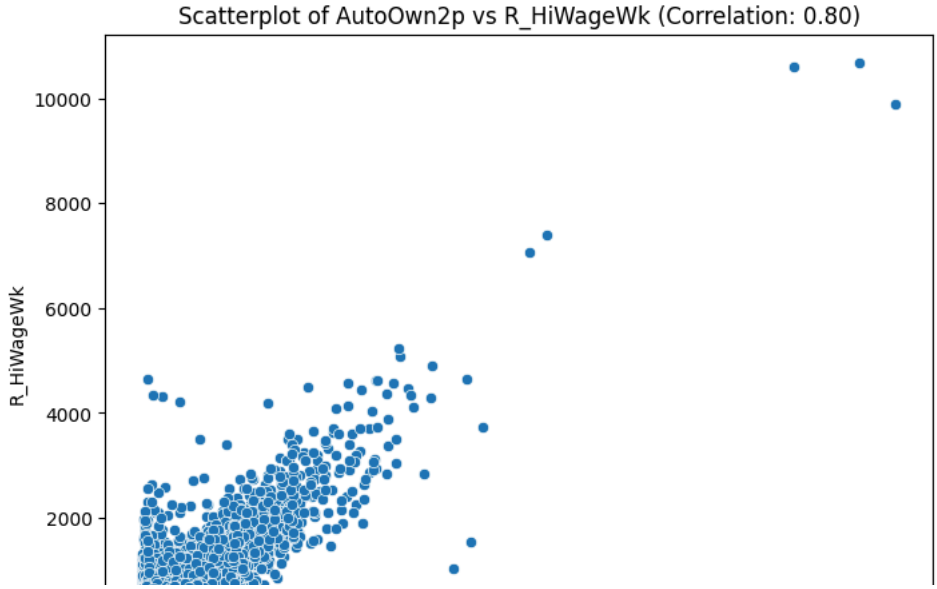


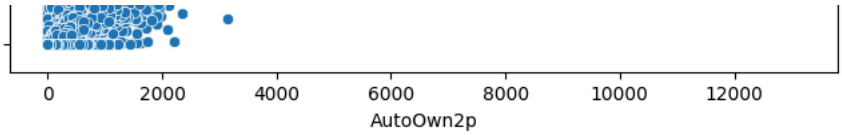
Scatterplot of AutoOwn2p vs Workers (Correlation: 0.83)

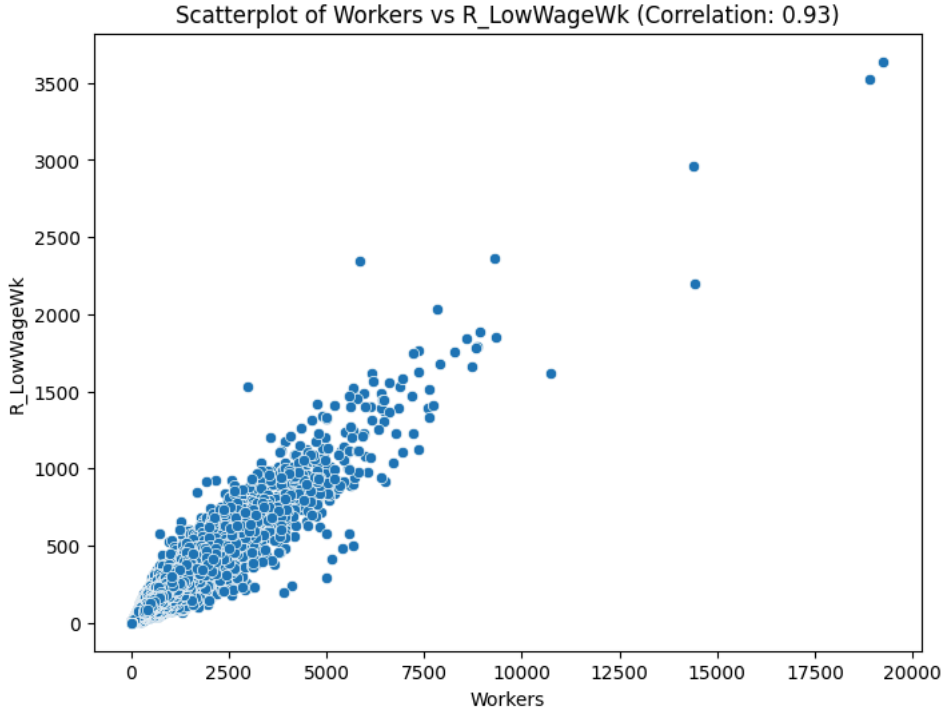


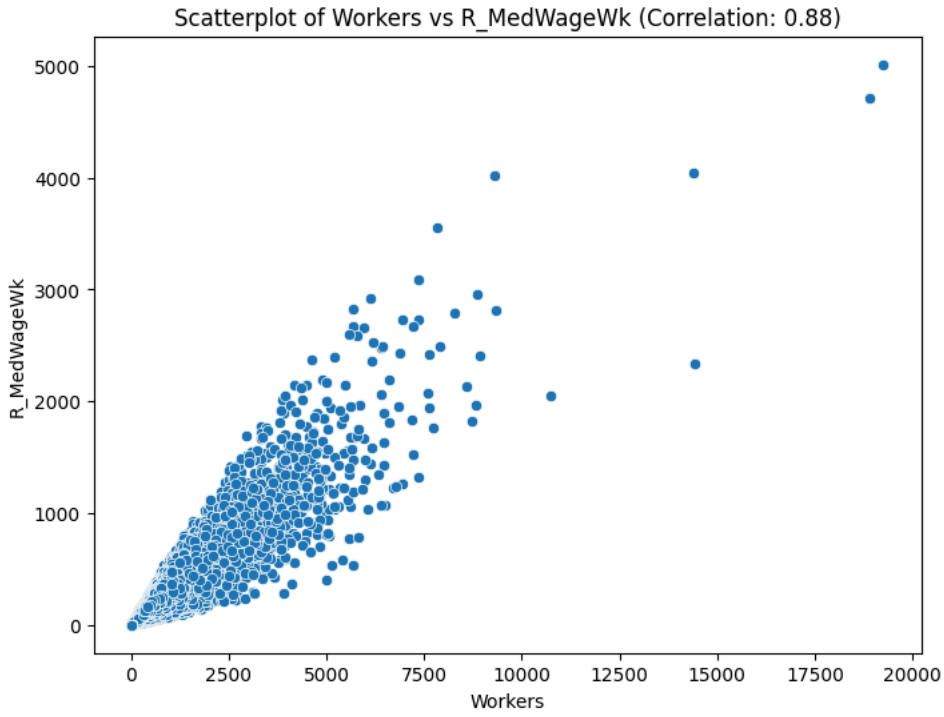

AutoOwn2p
Scatterplot of AutoOwn2p vs R_LowWageWk (Correlation: 0.76)
R_LowWageWk
0
500
1000
1500
2000
2500
3000
3500
0
2000
4000
6000
8000
10000
12000
AutoOwn2p

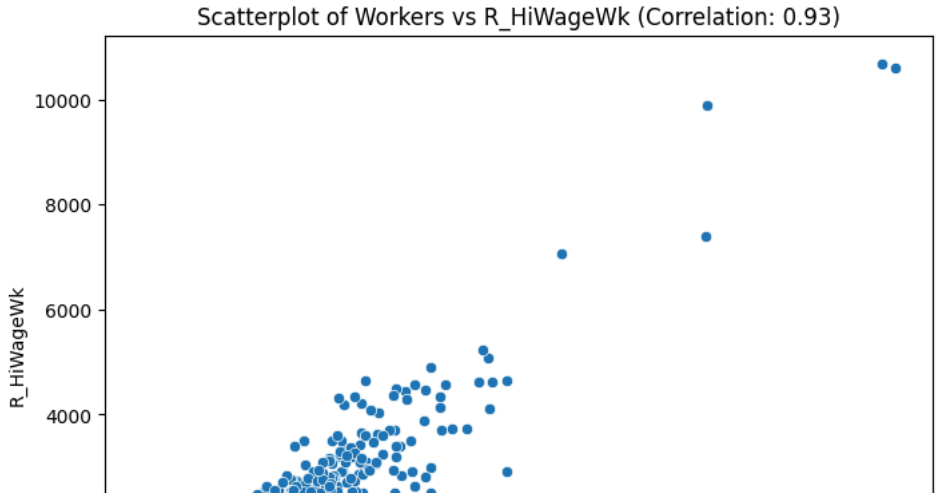

Scatterplot of AutoOwn2p vs R_MedWageWk (Correlation: 0.69)
R_MedWageWk
0
1000
2000
3000
4000
5000
0
2000
4000
6000
8000
10000
12000
AutoOwn2p

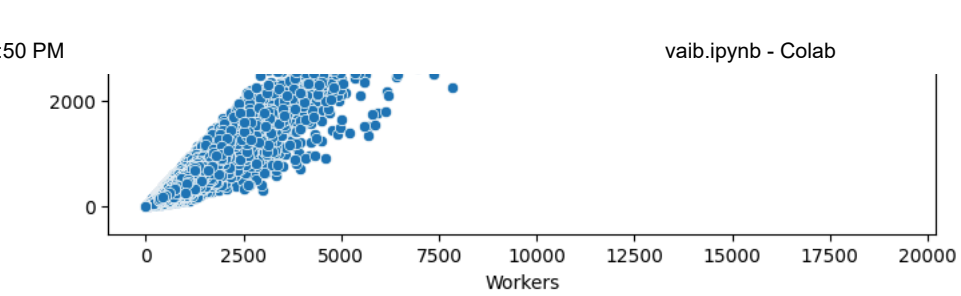

Scatterplot of AutoOwn2p vs R_HiWageWk (Correlation: 0.80)
R_HiWageWk
2000
4000
6000
8000
10000

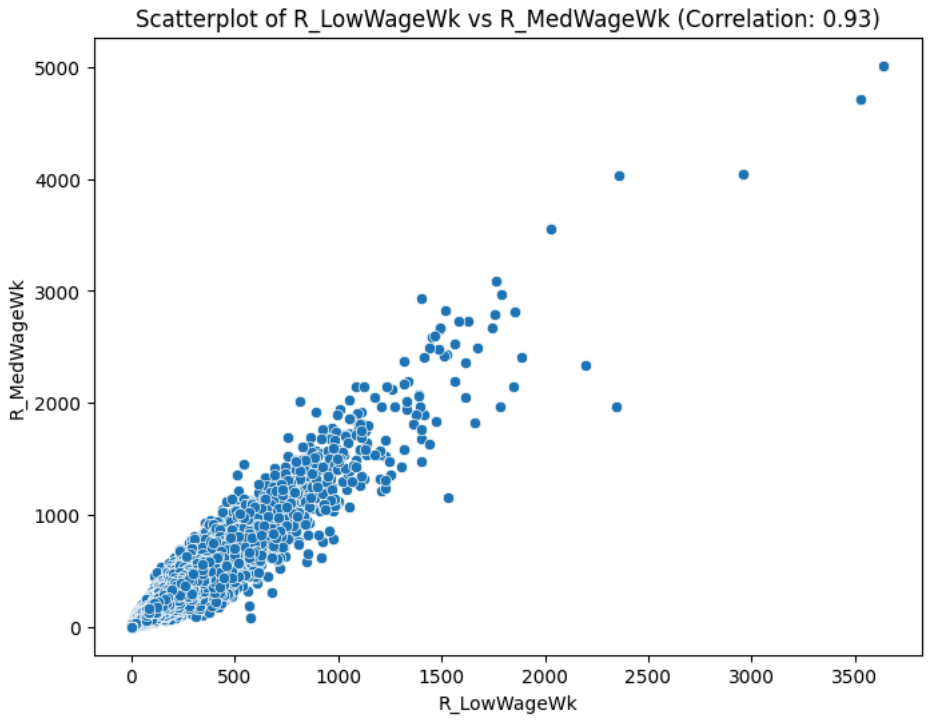

0
0
2000
4000
6000
8000
10000
12000
AutoOwn2p

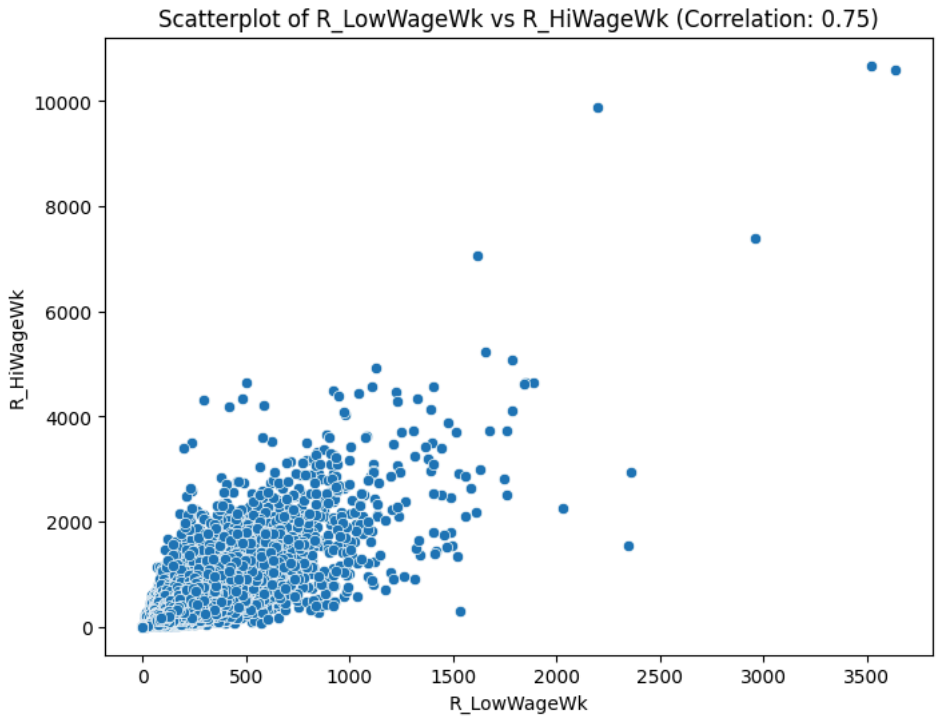

Scatterplot of Workers vs R_LowWageWk (Correlation: 0.93)
R_LowWageWk
0
500
1000
1500
2000
2500
3000
3500
0
2500
5000
7500
10000
12500
15000
17500
20000
Workers

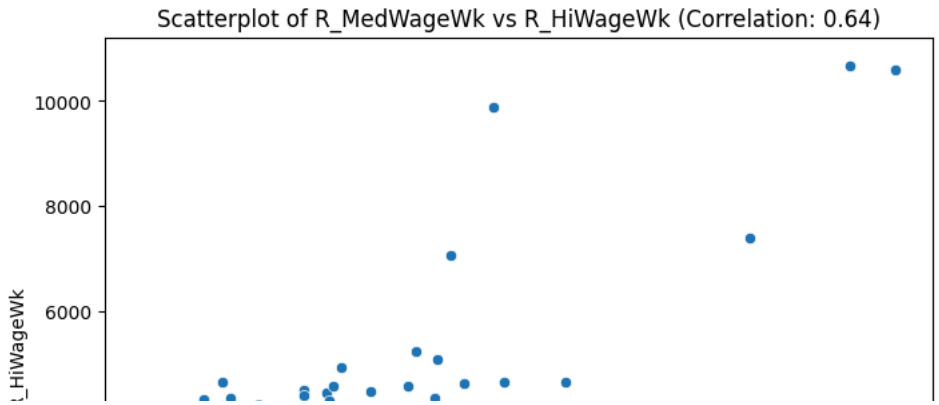

Scatterplot of Workers vs R_MedWageWk (Correlation: 0.88)
R_MedWageWk
0
1000
2000
3000
4000
5000
0
2500
5000
7500
10000
12500
15000
17500
20000
Workers

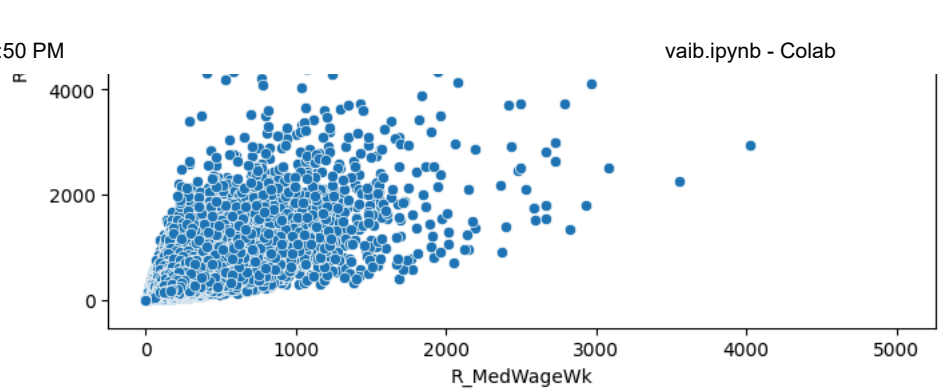

Scatterplot of Workers vs R_HiWageWk (Correlation: 0.93)
R_HiWageWk
4000
6000
8000
10000

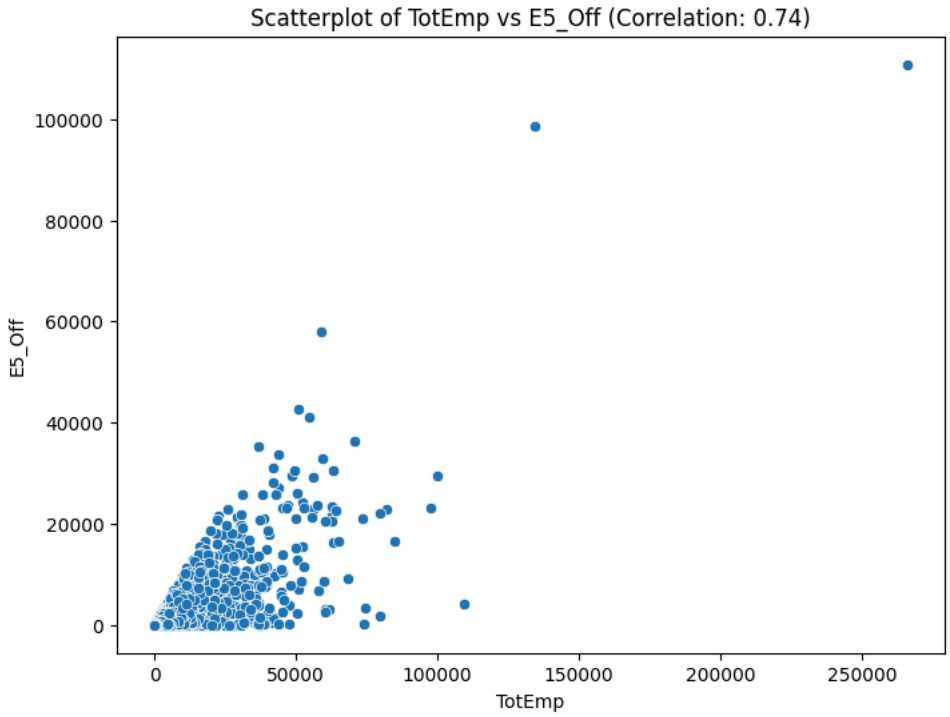

2000
0
0
2500
5000
7500
10000
12500
15000
17500
20000
Workers

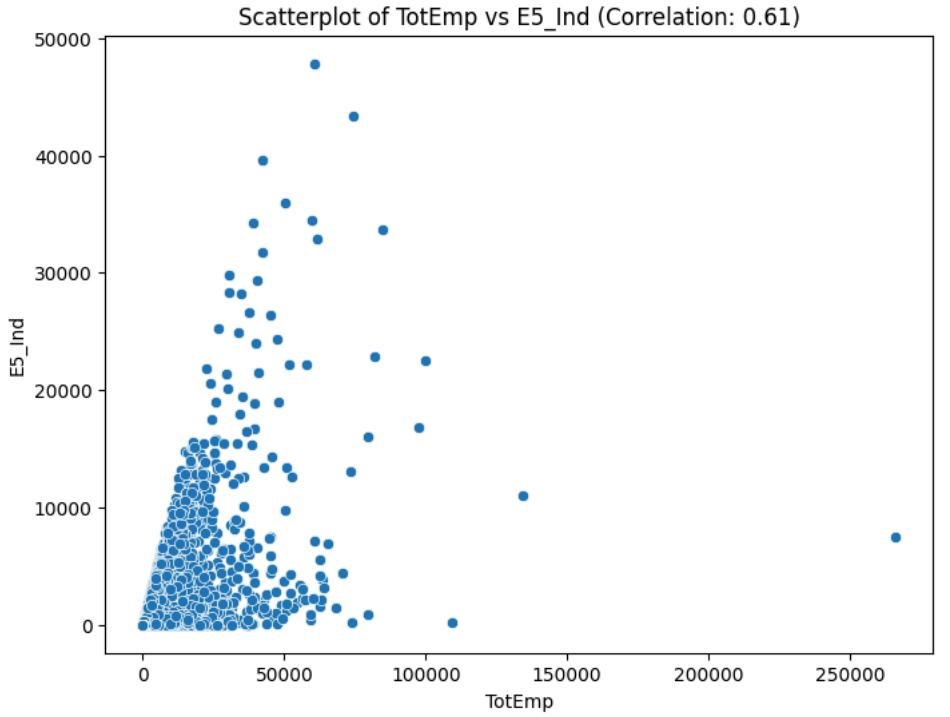

Scatterplot of R_LowWageWk vs R_MedWageWk (Correlation: 0.93)
5000
4000
3000
2000
1000
0
R_MedWageWk
0
500
1000
1500
2000
2500
3000
3500
R_LowWageWk

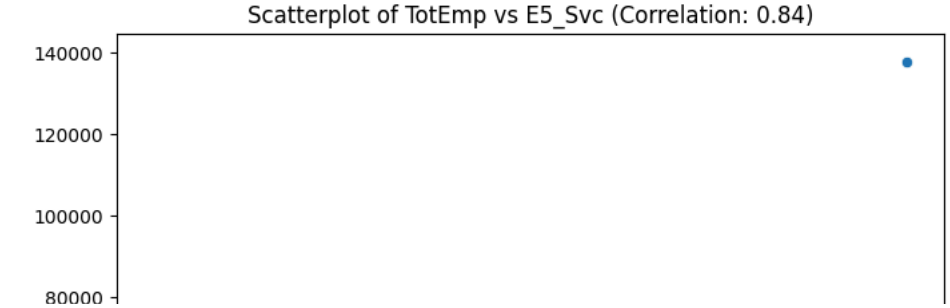

Scatterplot of R_LowWageWk vs R_HiWageWk (Correlation: 0.75)
10000
8000
6000
4000
2000
0
R_HiWageWk
0
500
1000
1500
2000
2500
3000
3500
R_LowWageWk

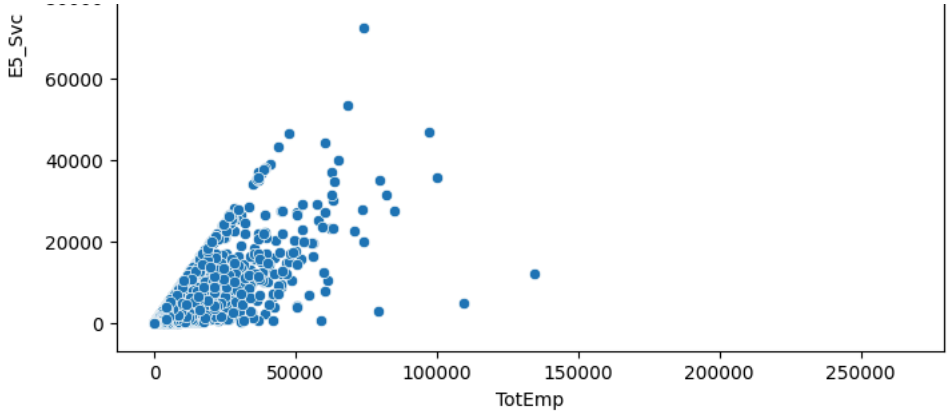

Scatterplot of R_MedWageWk vs R_HiWageWk (Correlation: 0.64)
10000
8000
6000

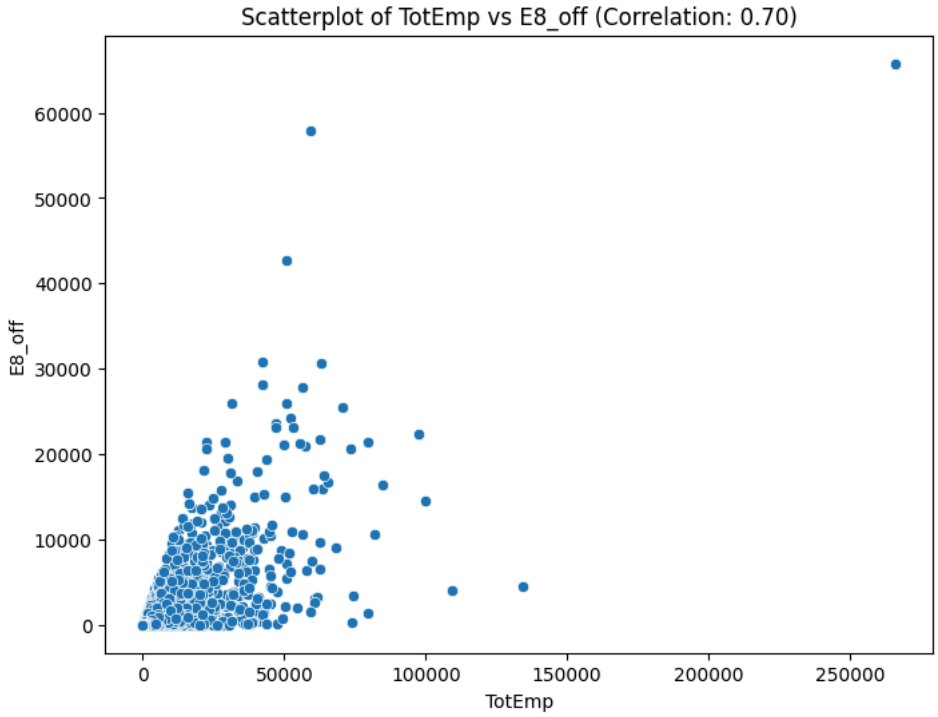

4000
2000
0
0
1000
2000
3000
4000
5000
R_MedWageWk

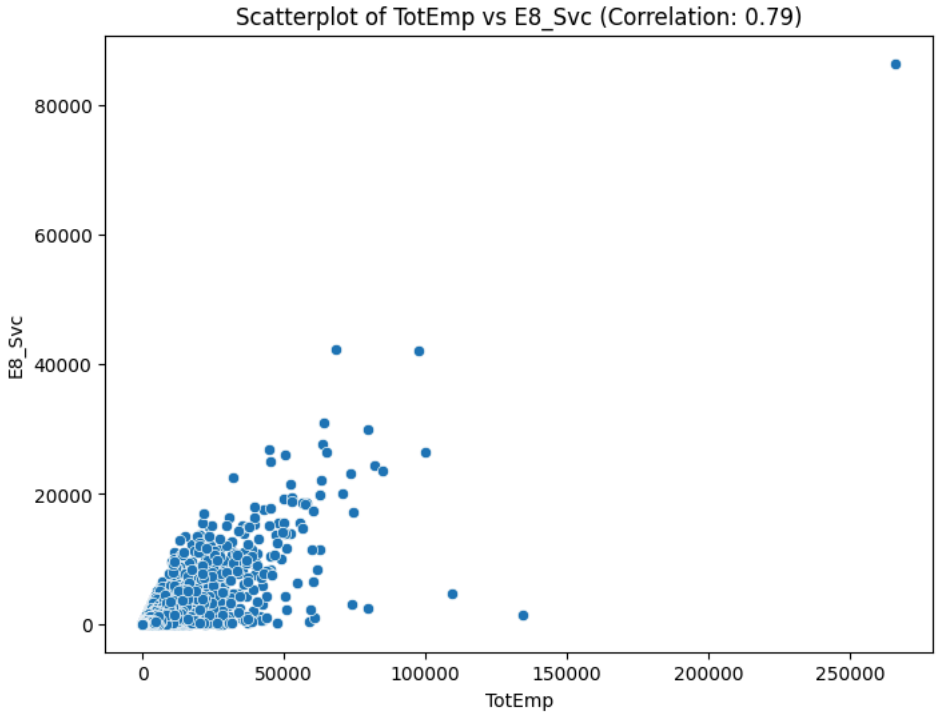

Scatterplot of TotEmp vs E5_Off (Correlation: 0.74)
100000
80000
60000
40000
20000
0
E5_Off
0
50000
100000
150000
200000
250000
TotEmp

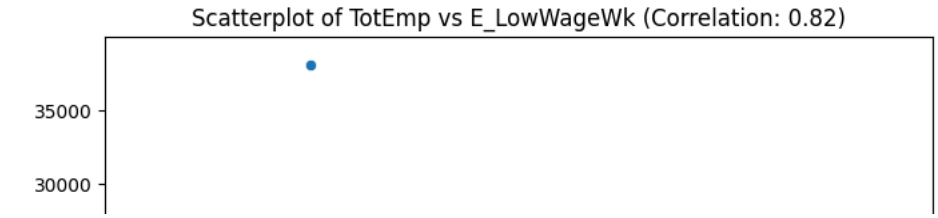

Scatterplot of TotEmp vs E5_Ind (Correlation: 0.61)
50000
40000
30000
20000
10000
0
E5_Ind
0
50000
100000
150000
200000
250000
TotEmp

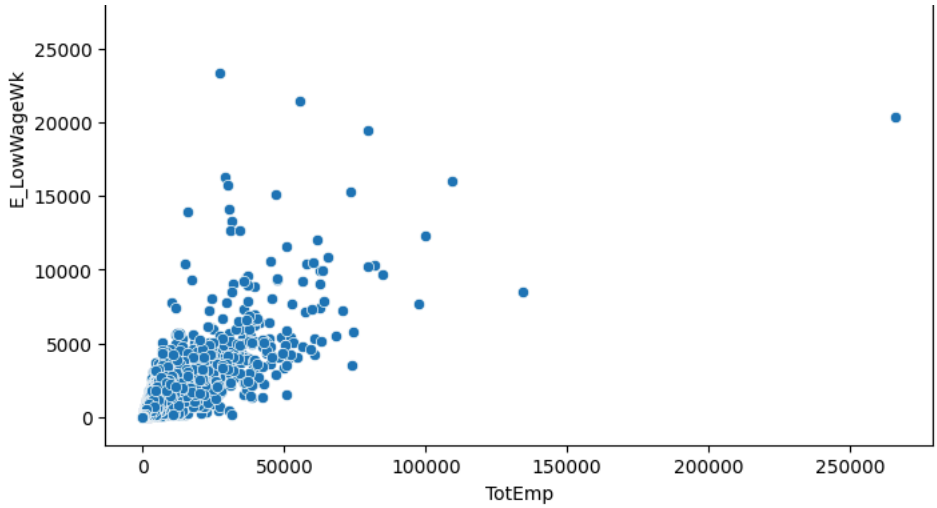

Scatterplot of TotEmp vs E5_Svc (Correlation: 0.84)
140000
120000
100000
80000


E5_Svc
60000
40000
20000
0
0
50000
100000
150000
200000
250000
TotEmp

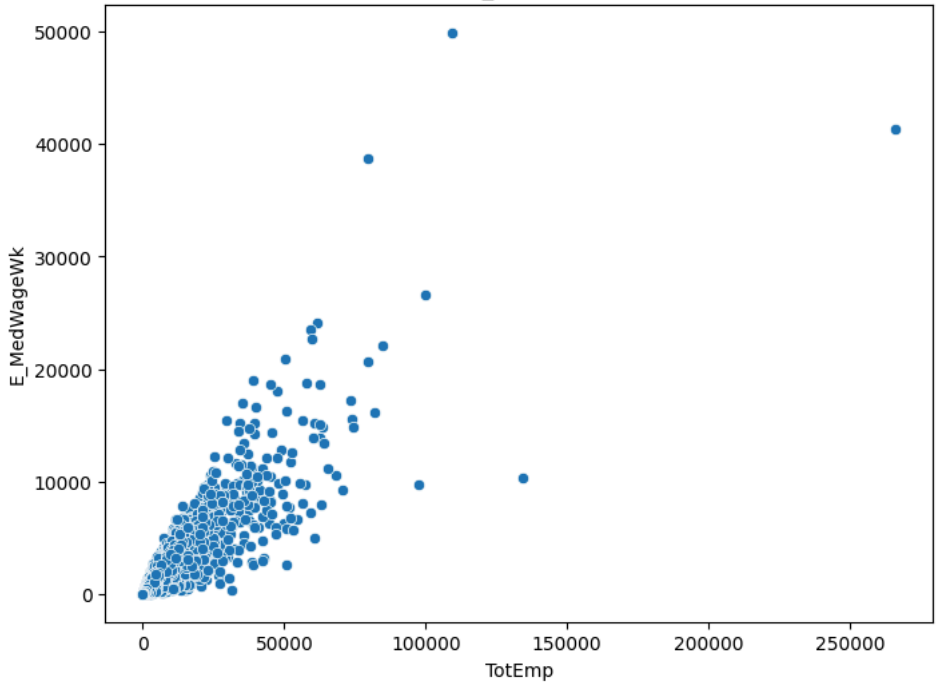

Scatterplot of TotEmp vs E8_off (Correlation: 0.70)
E8_off
60000
50000
40000
30000
20000
10000
0
0
50000
100000
150000
200000
250000
TotEmp


Scatterplot of TotEmp vs E8_Svc (Correlation: 0.79)
E8_Svc
80000
60000
40000
20000
0
0
50000
100000
150000
200000
250000
TotEmp

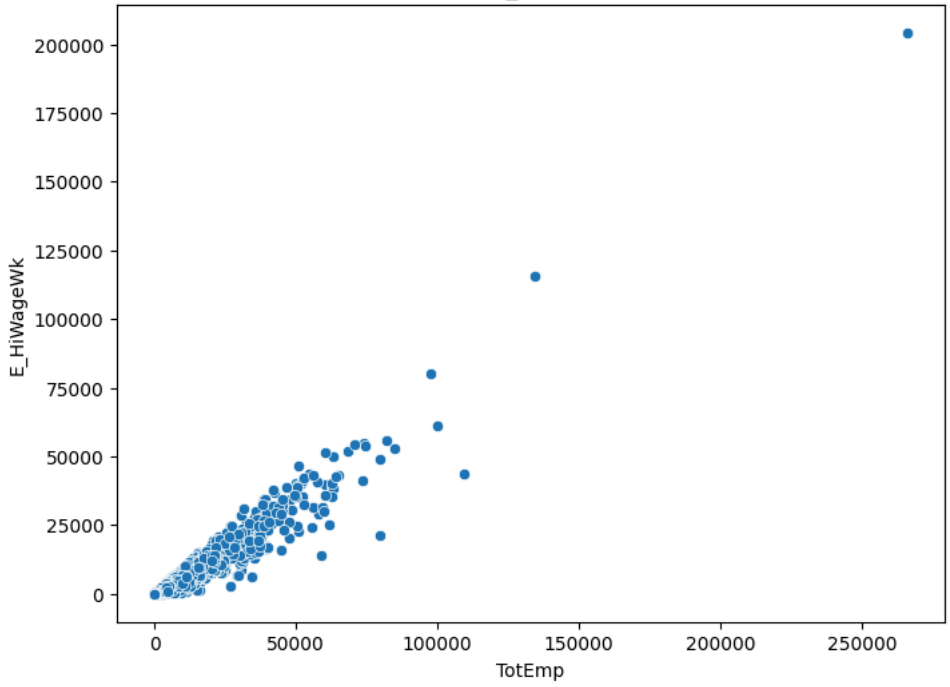

Scatterplot of TotEmp vs E_LowWageWk (Correlation: 0.82)
35000
30000



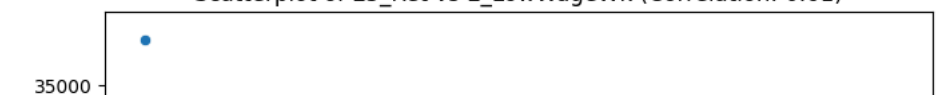
Scatterplot of TotEmp vs E_MedWageWk (Correlation: 0.91)

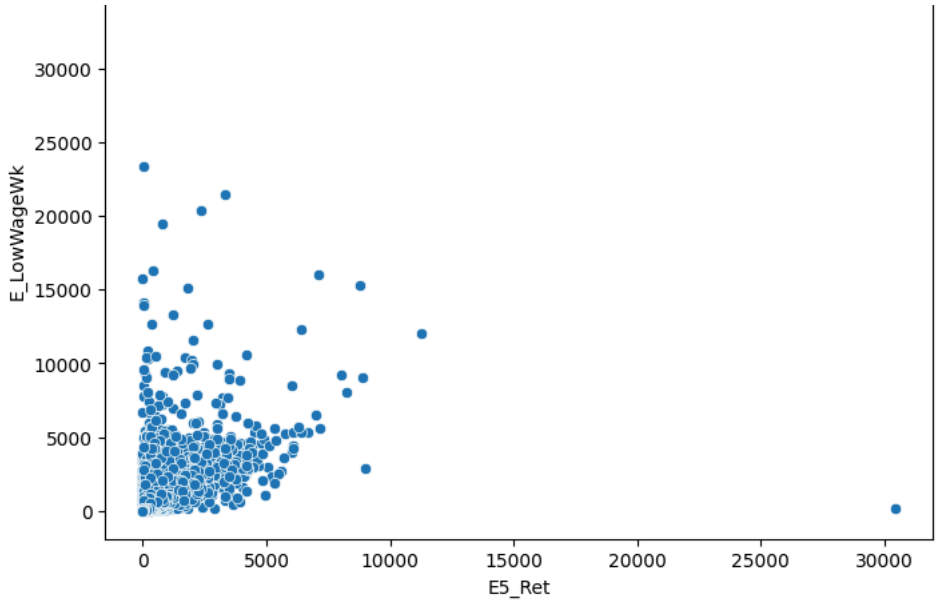


Scatterplot of TotEmp vs E_HiWageWk (Correlation: 0.96)

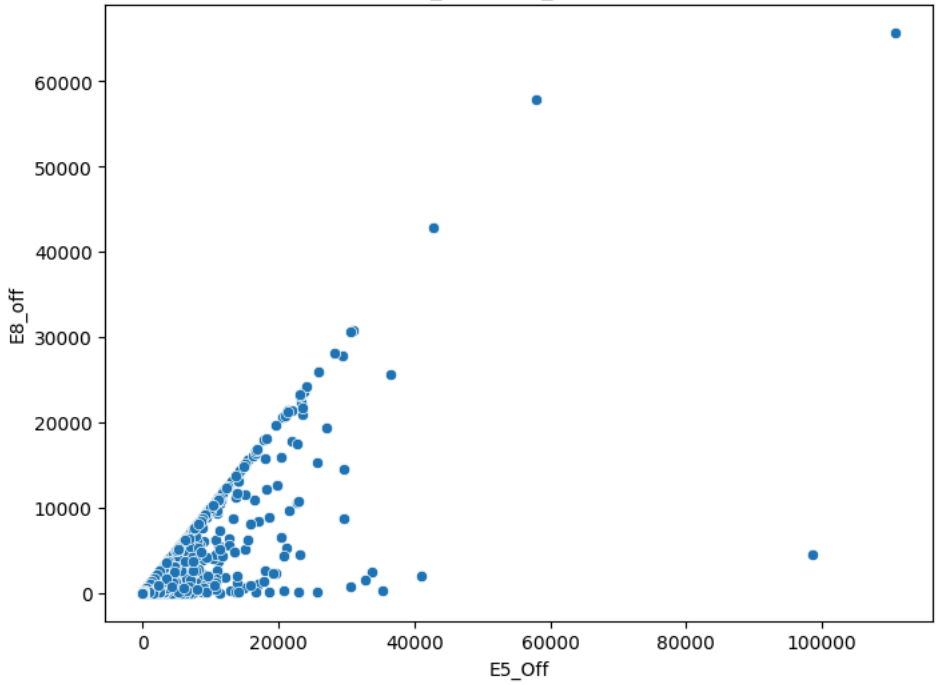


Scatterplot of E5_Ret vs E_LowWageWk (Correlation: 0.61)

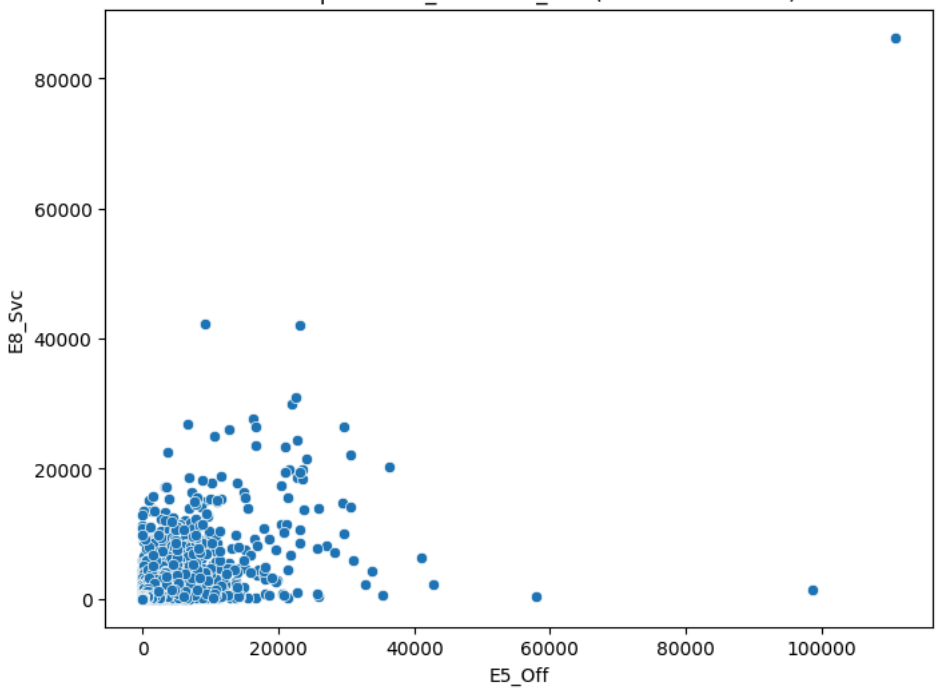




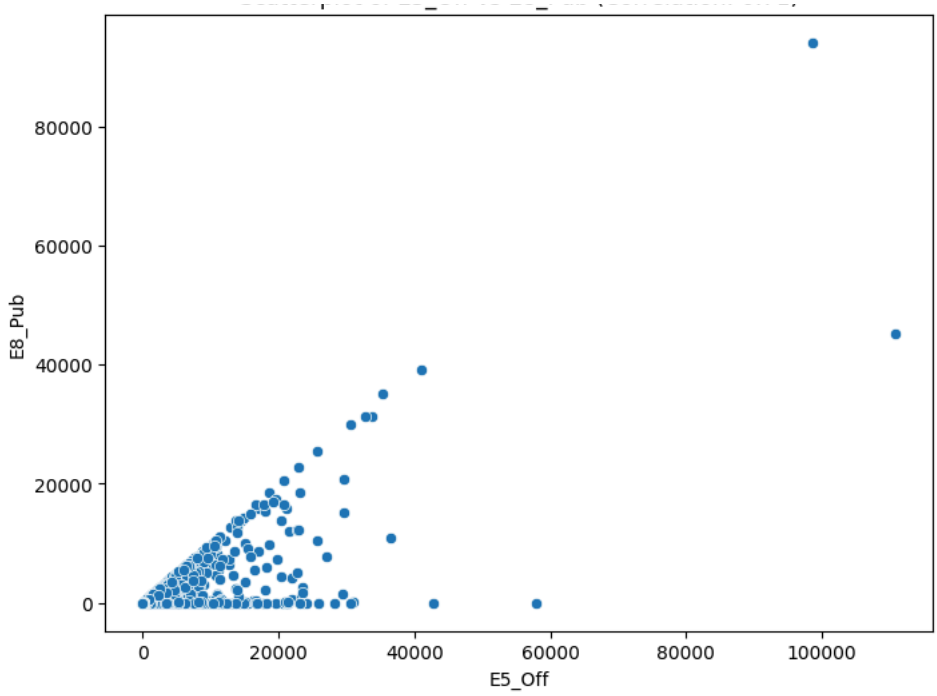
Scatterplot of E5_Off vs E8_off (Correlation: 0.83)

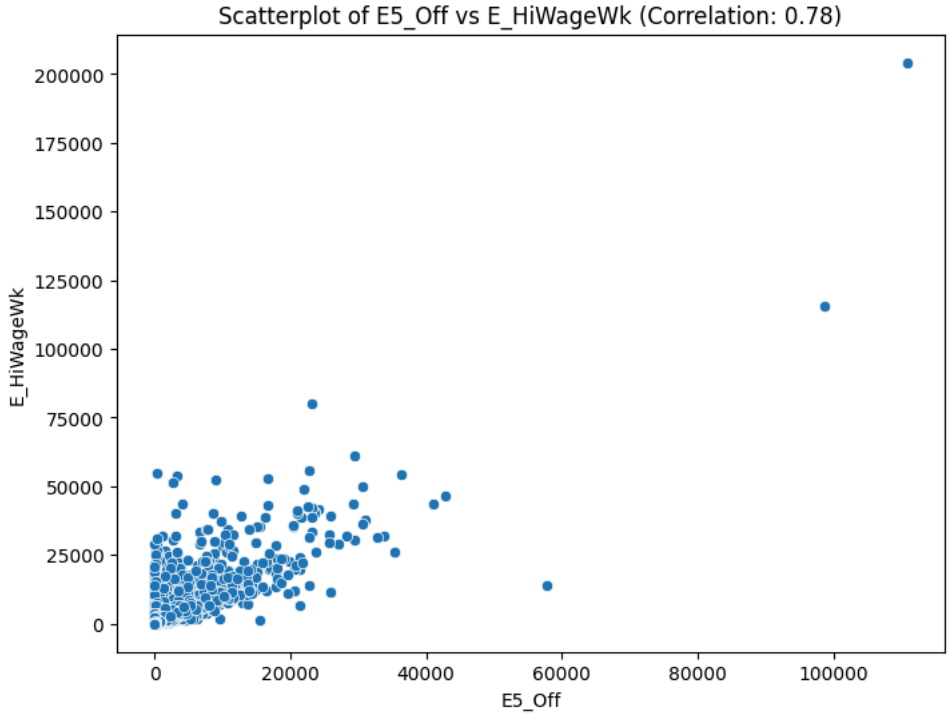


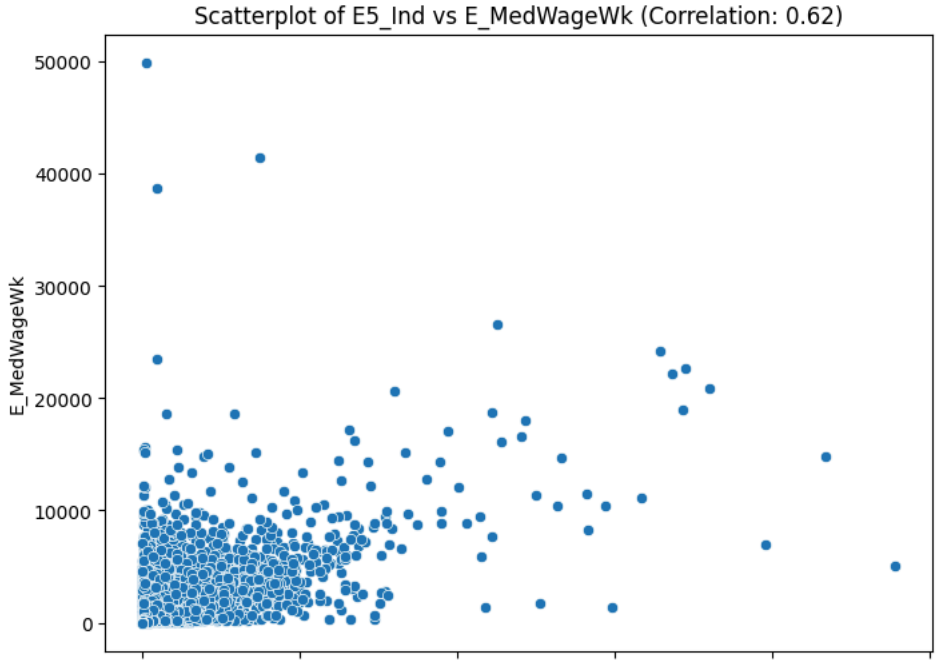
Scatterplot of E5_Off vs E8_Svc (Correlation: 0.62)

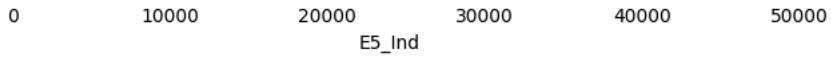


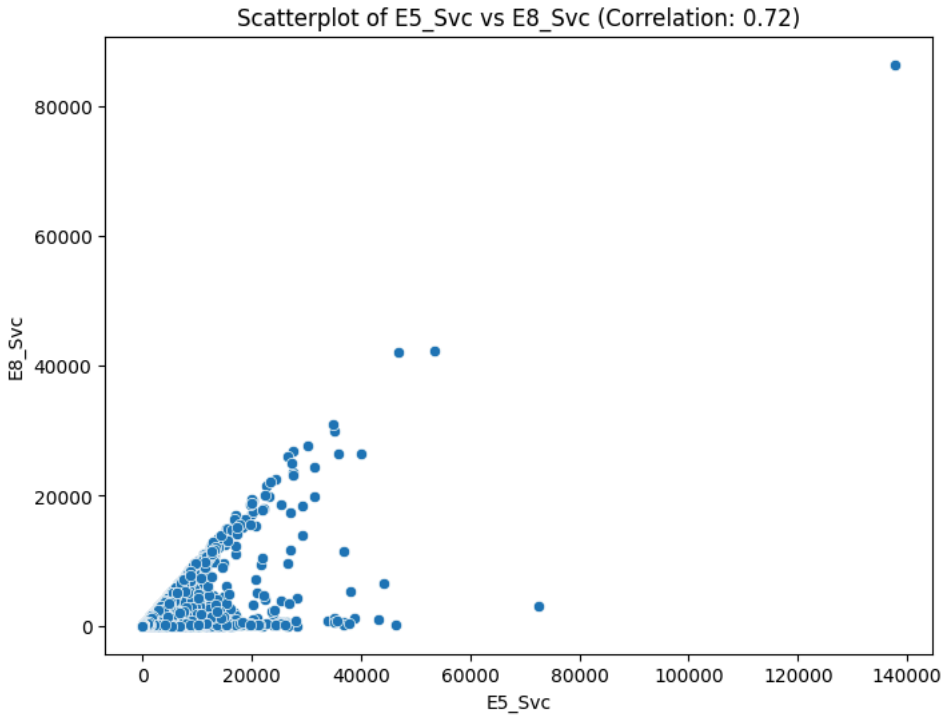
Scatterplot of E5_Off vs E8_Pub (Correlation: 0.71)

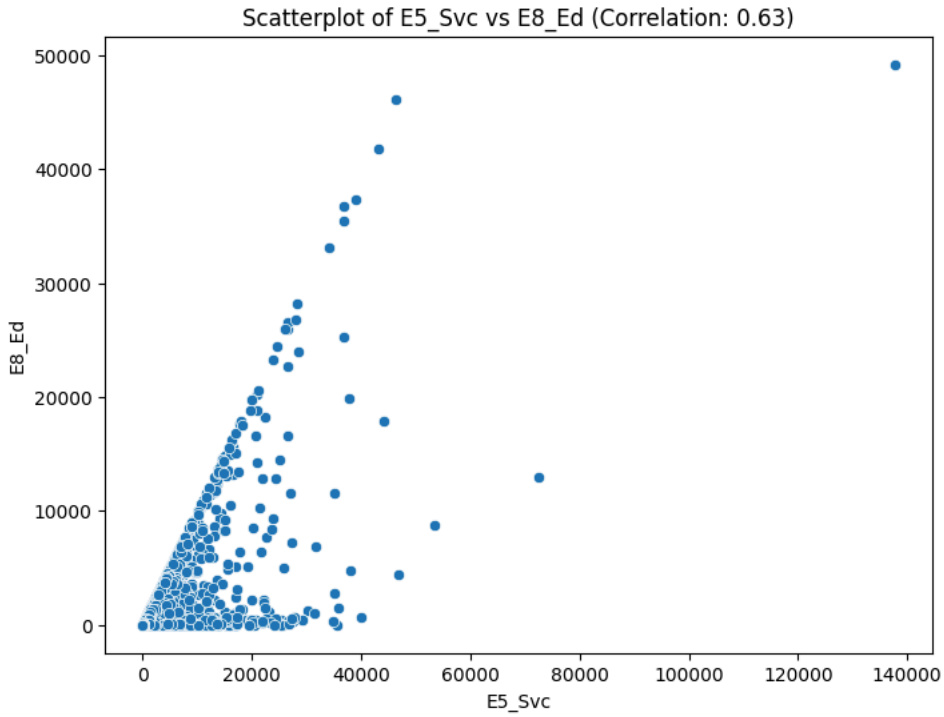

E8_Pub
E5_Off
0
20000
40000
60000
80000
100000

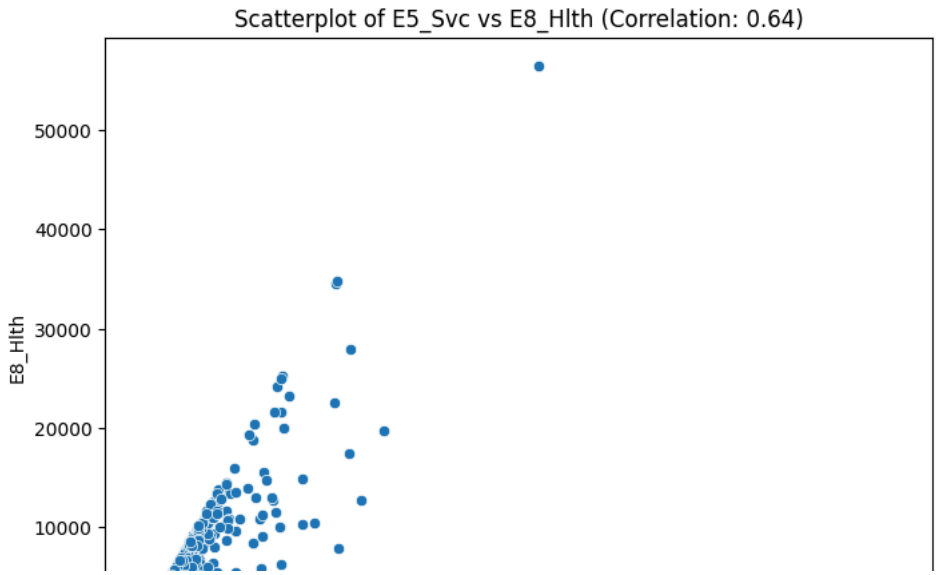

Scatterplot of E5_Off vs E_HiWageWk (Correlation: 0.78)
E_HiWageWk
E5_Off
0
25000
50000
75000
100000
125000
150000
175000
200000
0
20000
40000
60000
80000
100000

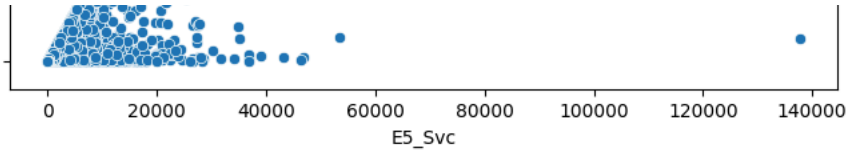

Scatterplot of E5_Ind vs E_MedWageWk (Correlation: 0.62)
E_MedWageWk
0
10000
20000
30000
40000
50000

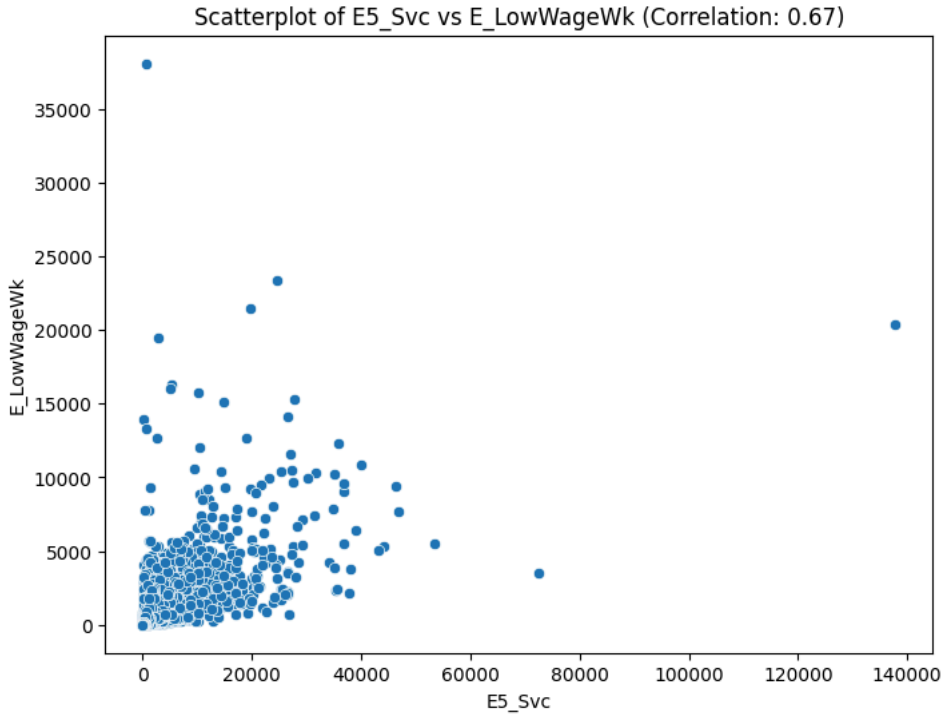

0
10000
20000
30000
40000
50000
E5_Ind

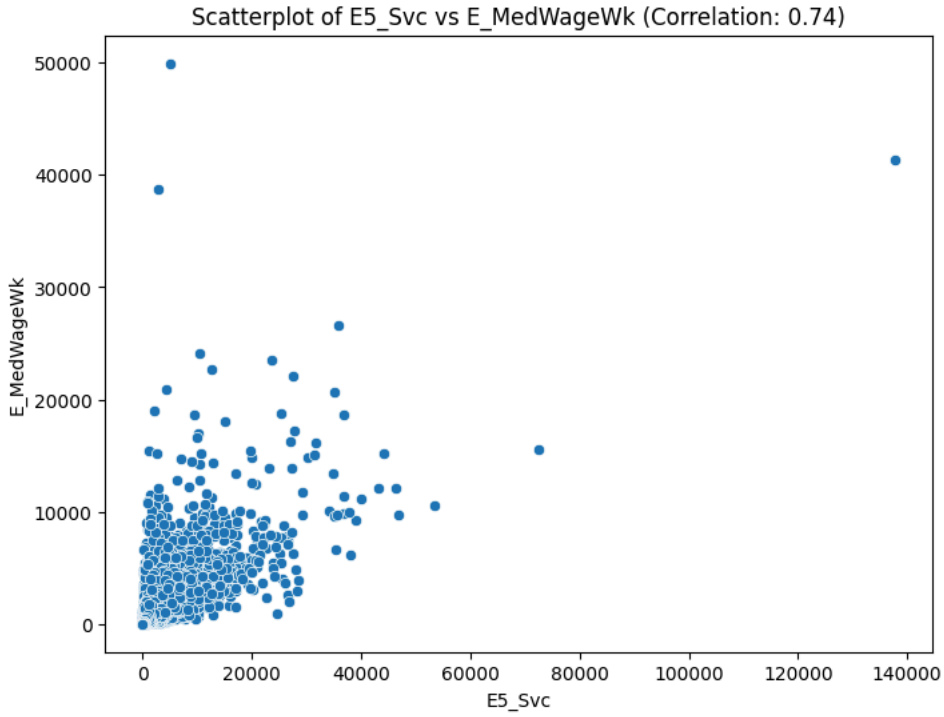

Scatterplot of E5_Svc vs E8_Svc (Correlation: 0.72)
E8_Svc
E5_Svc
0
20000
40000
60000
80000
100000
120000
140000

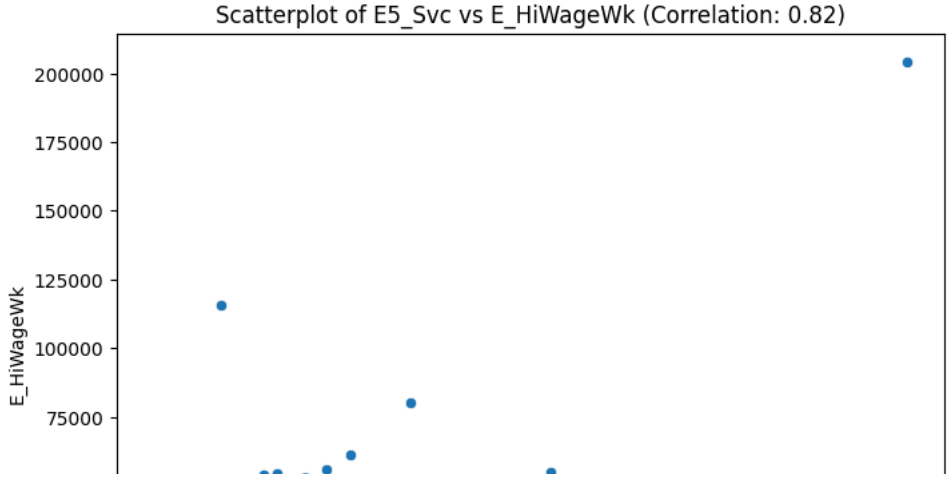

Scatterplot of E5_Svc vs E8_Ed (Correlation: 0.63)
E8_Ed
E5_Svc
0
10000
20000
30000
40000
50000
60000
80000
100000
120000
140000

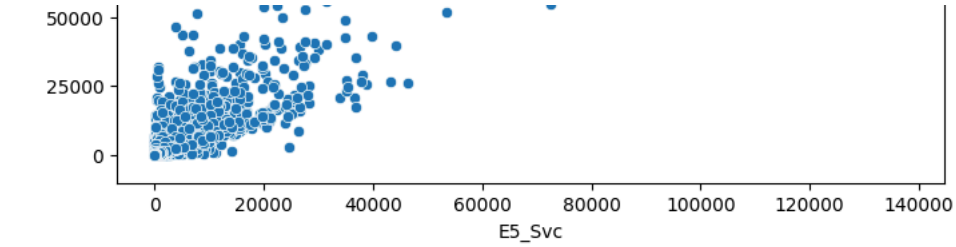

Scatterplot of E5_Svc vs E8_Hlth (Correlation: 0.64)
E8_Hlth
10000
20000
30000
40000
50000

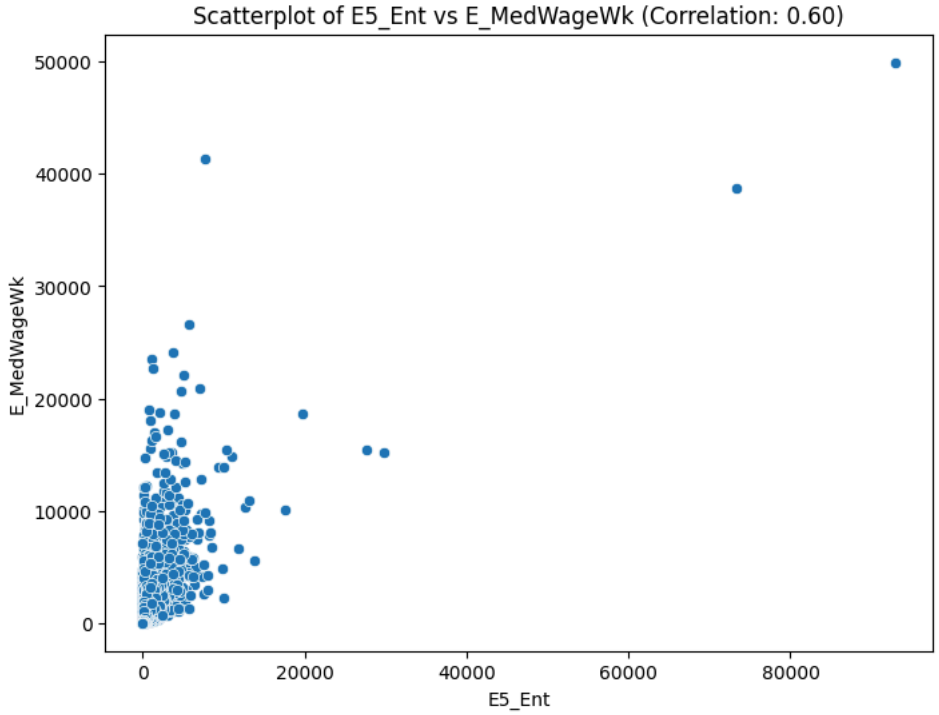

0
0
20000
40000
60000
80000
100000
120000
140000
E5_Svc

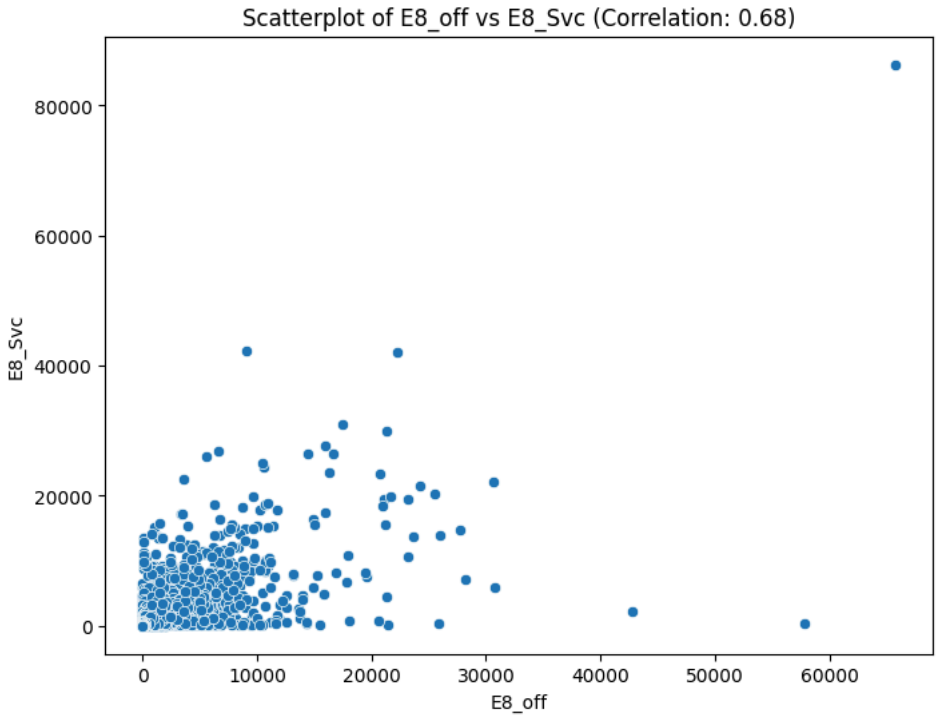

Scatterplot of E5_Svc vs E_LowWageWk (Correlation: 0.67)
E_LowWageWk
35000
30000
25000
20000
15000
10000
5000
0
0
20000
40000
60000
80000
100000
120000
140000
E5_Svc

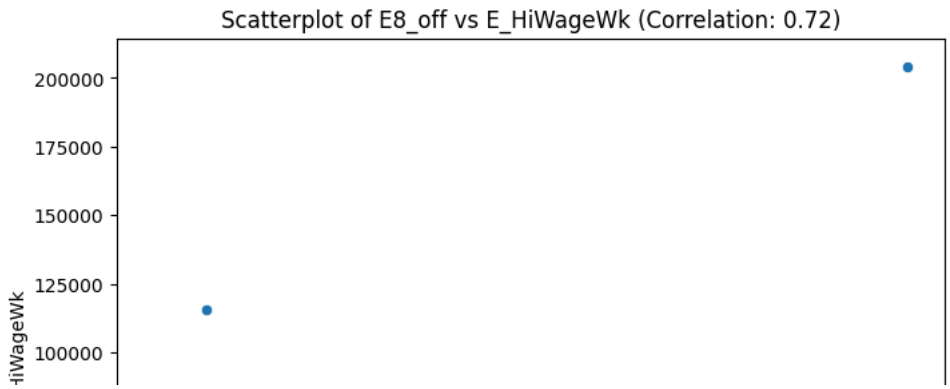

Scatterplot of E5_Svc vs E_MedWageWk (Correlation: 0.74)
E_MedWageWk
50000
40000
30000
20000
10000
0
0
20000
40000
60000
80000
100000
120000
140000
E5_Svc

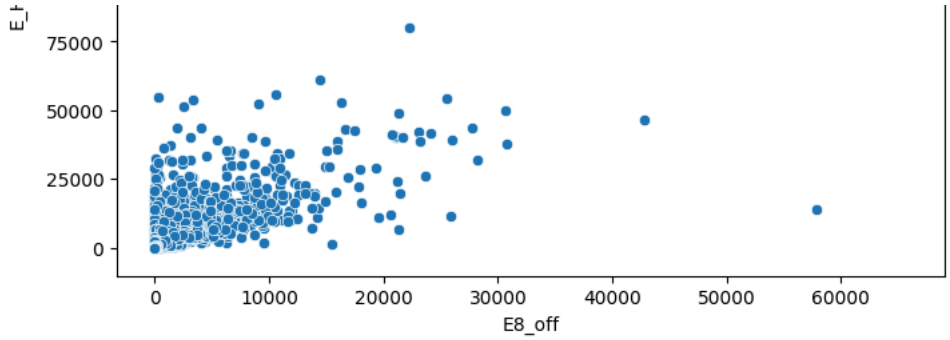

Scatterplot of E5_Svc vs E_HiWageWk (Correlation: 0.82)
E_HiWageWk
200000
175000
150000
125000
100000
75000

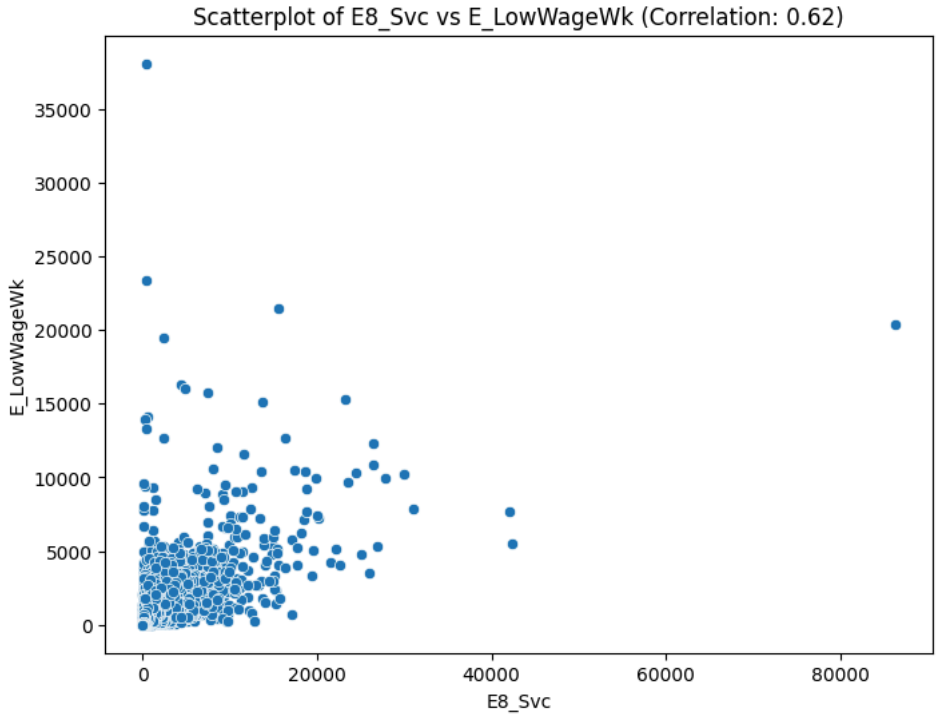

50000
25000
0
0
20000
40000
60000
80000
100000
120000
140000
E5_Svc

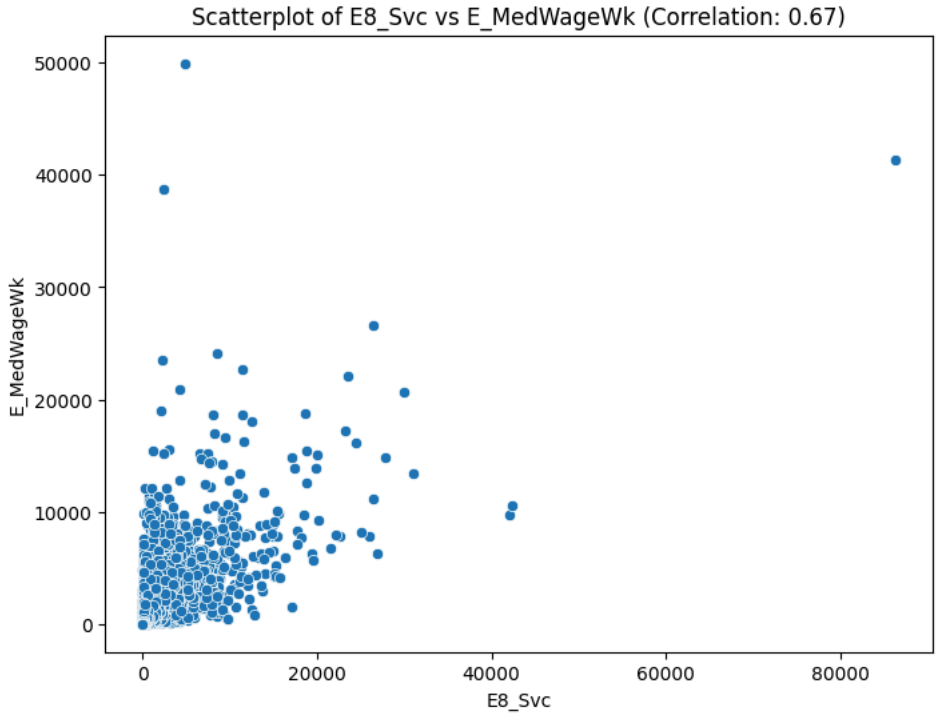

Scatterplot of E5_Ent vs E_MedWageWk (Correlation: 0.60)
50000
40000
30000
20000
10000
0
E_MedWageWk
0
20000
40000
60000
80000
E5_Ent

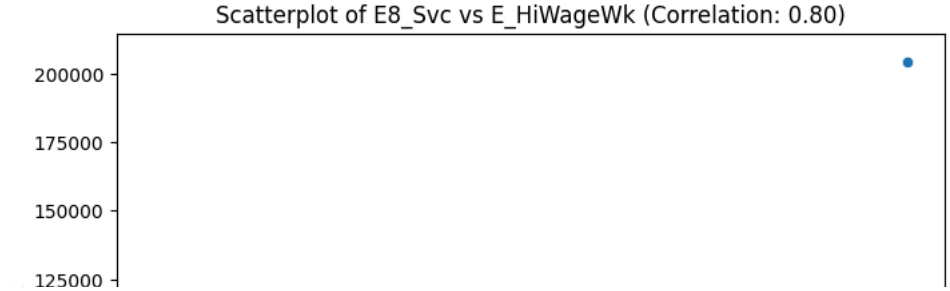

Scatterplot of E8_off vs E8_Svc (Correlation: 0.68)
80000
60000
40000
20000
0
E8_Svc
0
10000
20000
30000
40000
50000
60000
E8_off

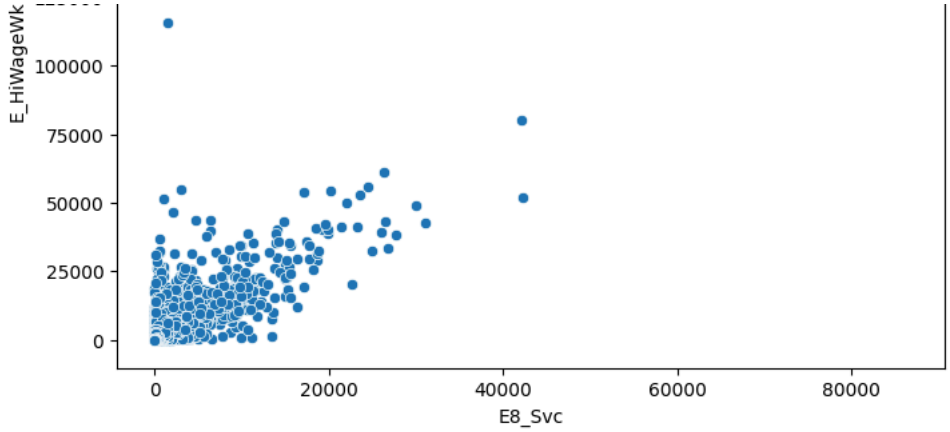

Scatterplot of E8_off vs E_HiWageWk (Correlation: 0.72)
200000
175000
150000
125000
100000
iWageWk

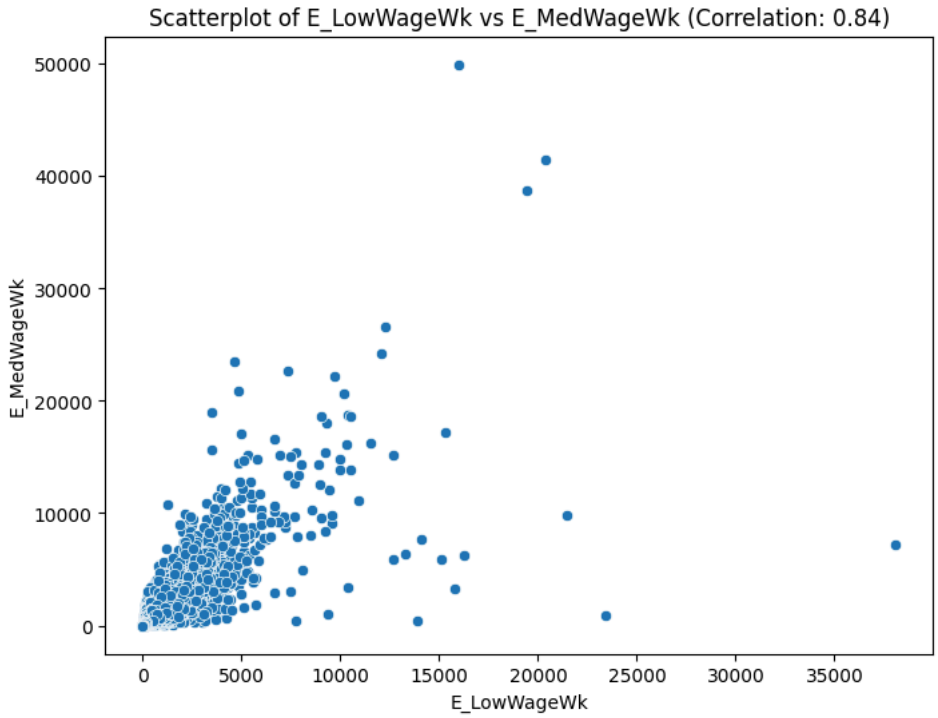

E_I
75000
50000
25000
0
0
10000
20000
30000
40000
50000
60000
E8_off

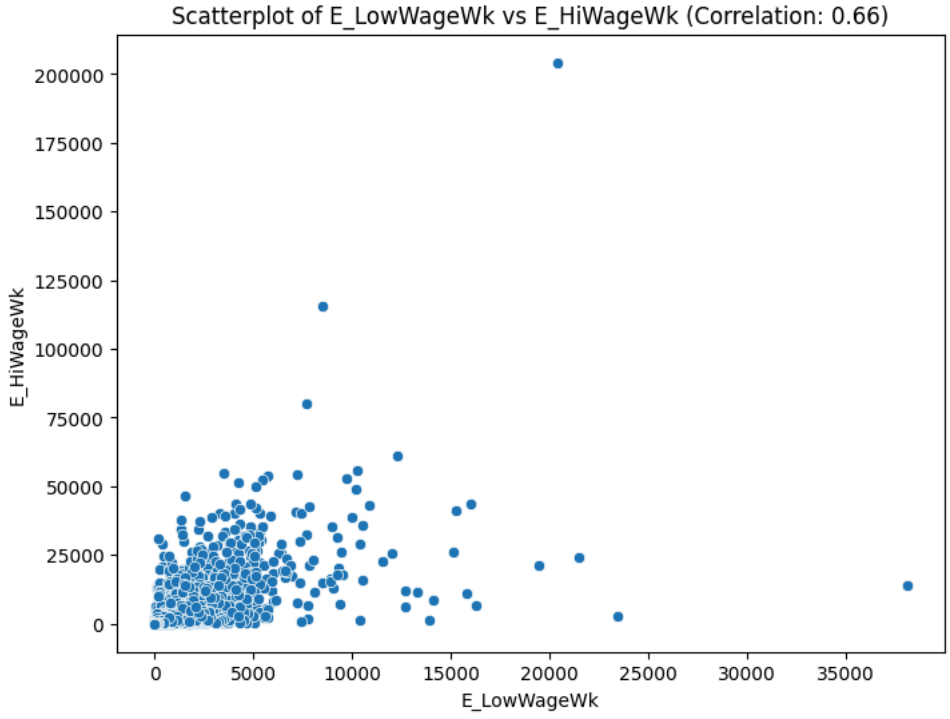

Scatterplot of E8_Svc vs E_LowWageWk (Correlation: 0.62)
E_LowWageWk
35000
30000
25000
20000
15000
10000
5000
0
0
20000
40000
60000
80000
E8_Svc

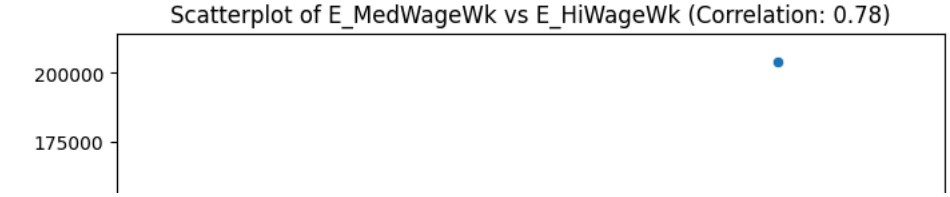

Scatterplot of E8_Svc vs E_MedWageWk (Correlation: 0.67)
E_MedWageWk
50000
40000
30000
20000
10000
0
0
20000
40000
60000
80000
E8_Svc

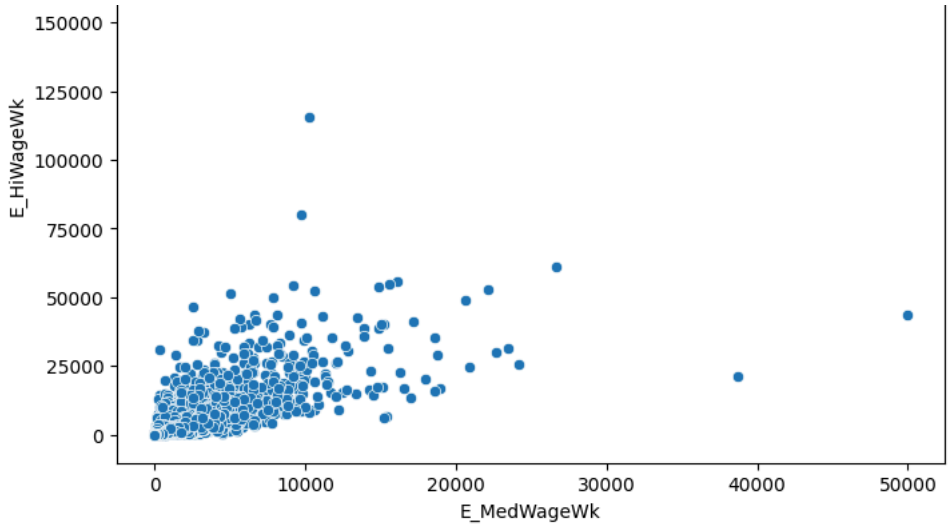

Scatterplot of E8_Svc vs E_HiWageWk (Correlation: 0.80)
200000
175000
150000
125000

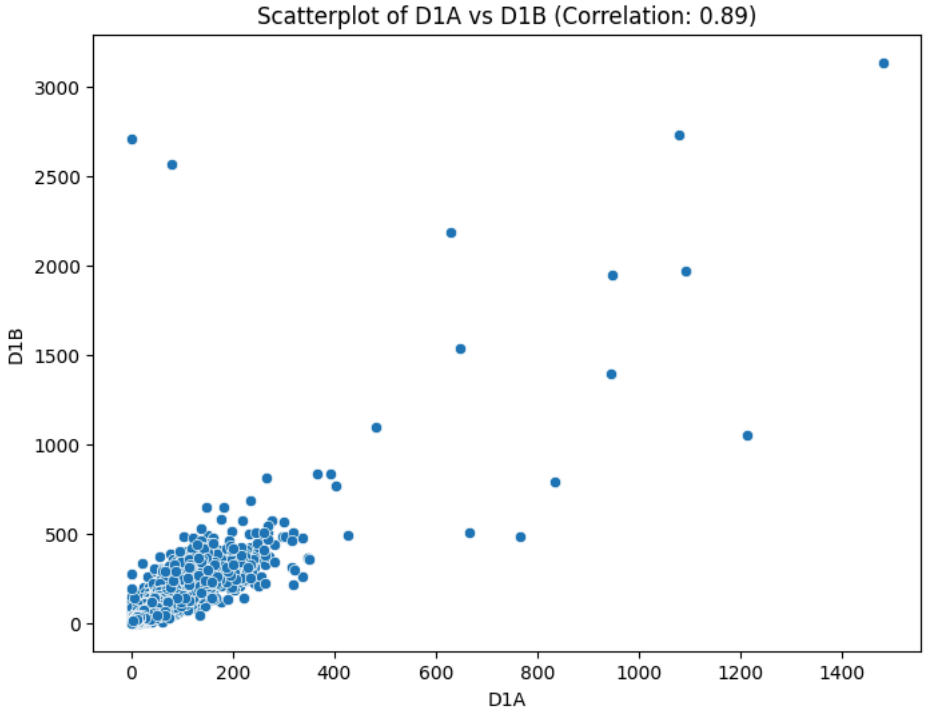

E_HiWageWk
100000
75000
50000
25000
0
0
20000
40000
60000
80000
E8_Svc

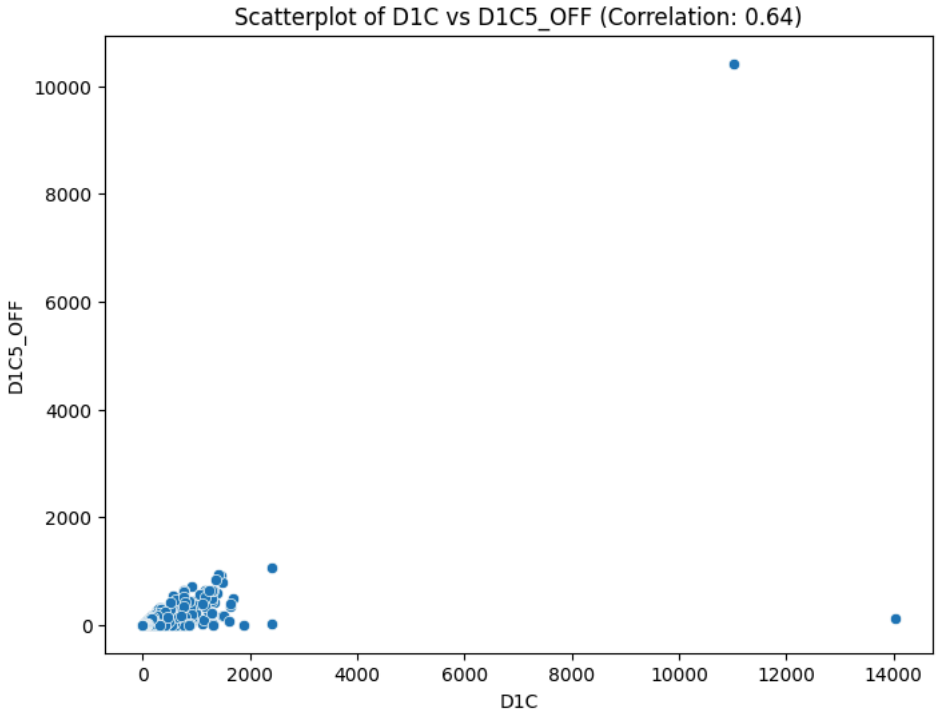

Scatterplot of E_LowWageWk vs E_MedWageWk (Correlation: 0.84)
E_MedWageWk
50000
40000
30000
20000
10000
0
0
5000
10000
15000
20000
25000
30000
35000
E_LowWageWk

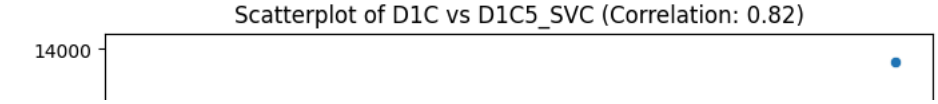

Scatterplot of E_LowWageWk vs E_HiWageWk (Correlation: 0.66)
E_HiWageWk
200000
175000
150000
125000
100000
75000
50000
25000
0
0
5000
10000
15000
20000
25000
30000
35000
E_LowWageWk

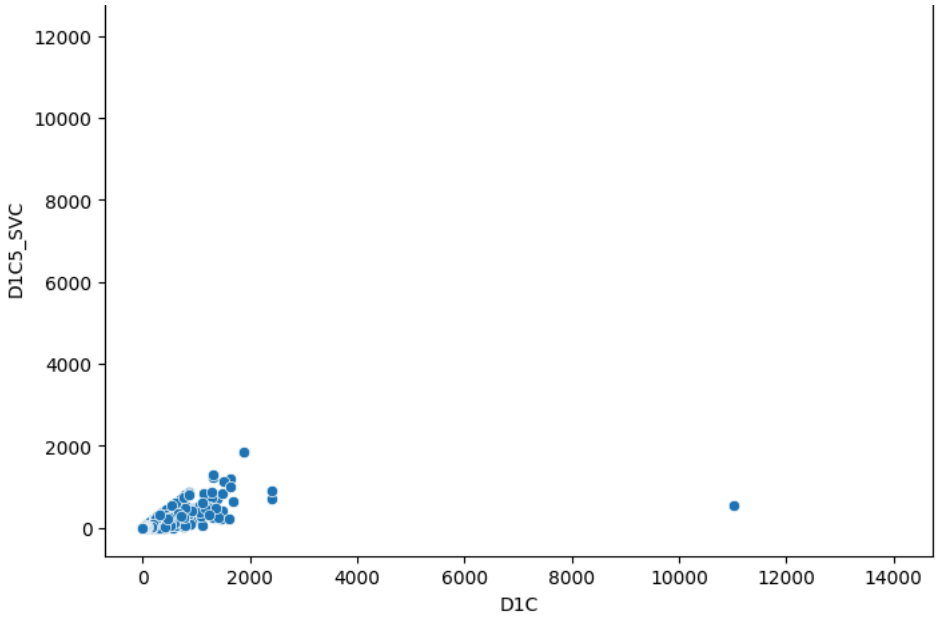

Scatterplot of E_MedWageWk vs E_HiWageWk (Correlation: 0.78)
200000
175000

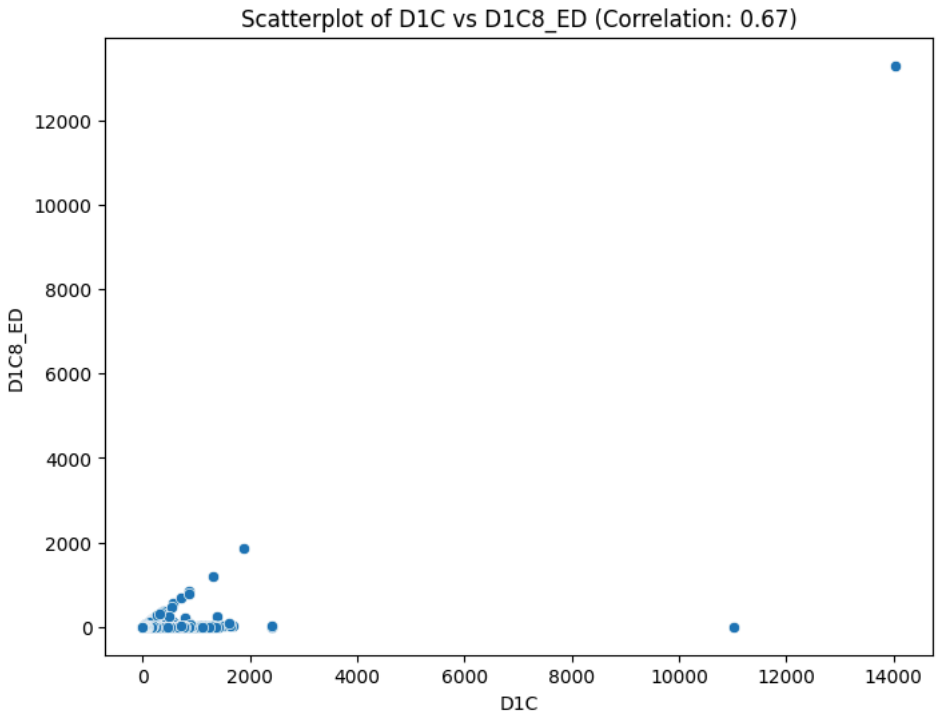

E_HiWageWk
E_MedWageWk
0
25000
50000
75000
100000
125000
150000
0
10000
20000
30000
40000
50000

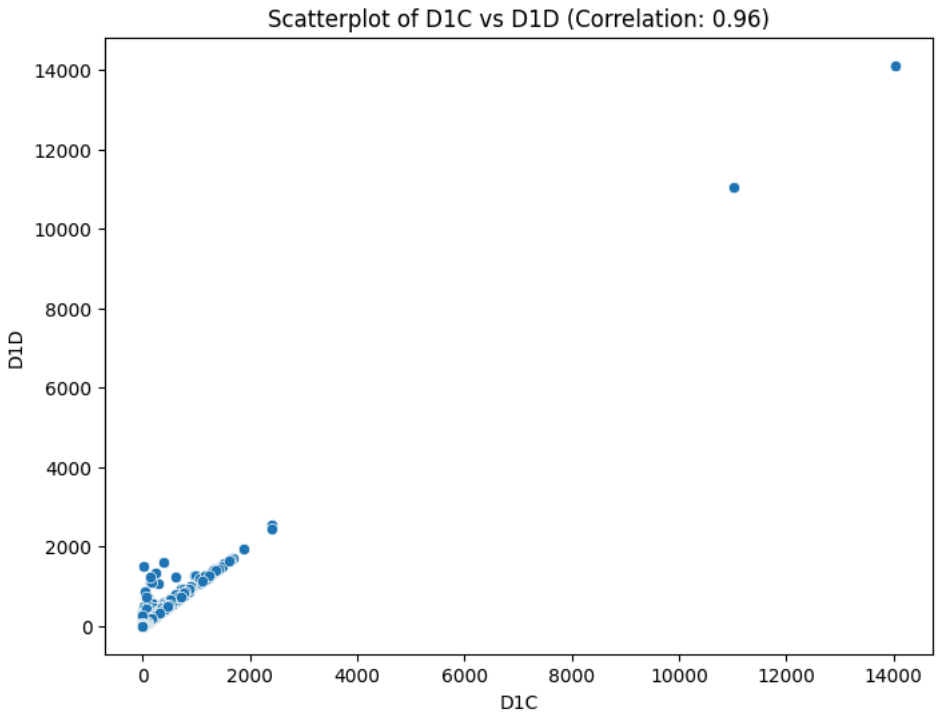

Scatterplot of D1A vs D1B (Correlation: 0.89)
D1B
D1A
0
500
1000
1500
2000
2500
3000
0
200
400
600
800
1000
1200
1400

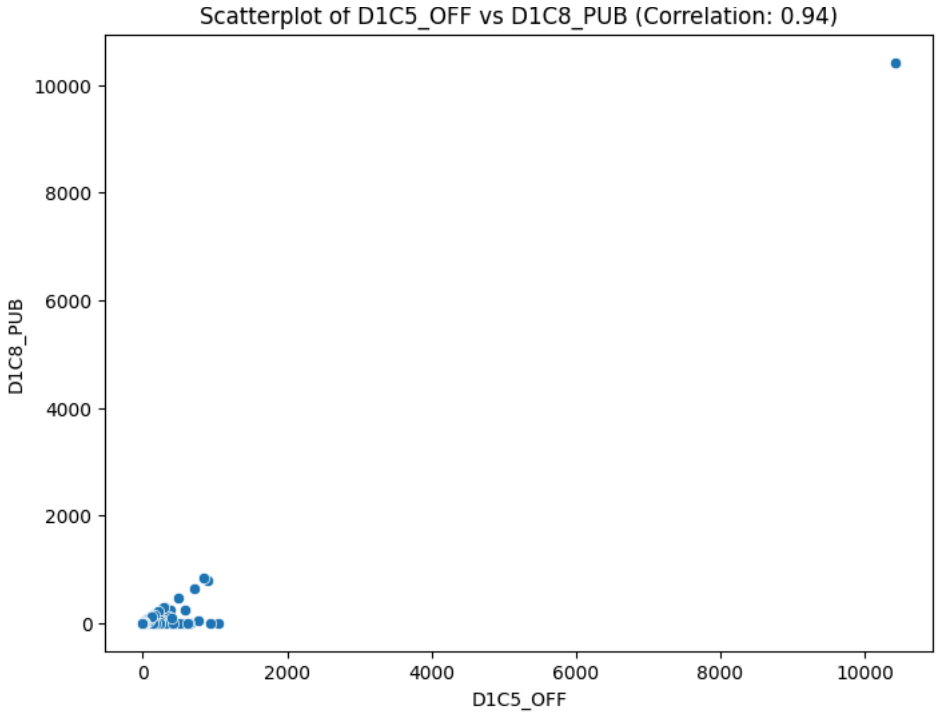

Scatterplot of D1C vs D1C5_OFF (Correlation: 0.64)
D1C5_OFF
D1C
0
2000
4000
6000
8000
10000
0
2000
4000
6000
8000
10000
12000
14000

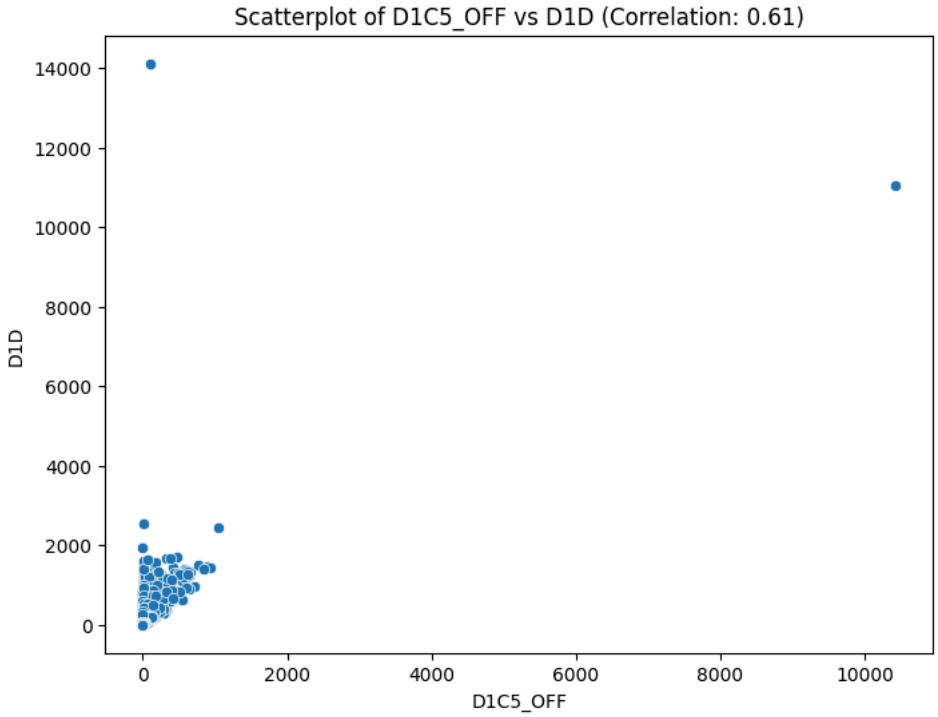

Scatterplot of D1C vs D1C5_SVC (Correlation: 0.82)
14000

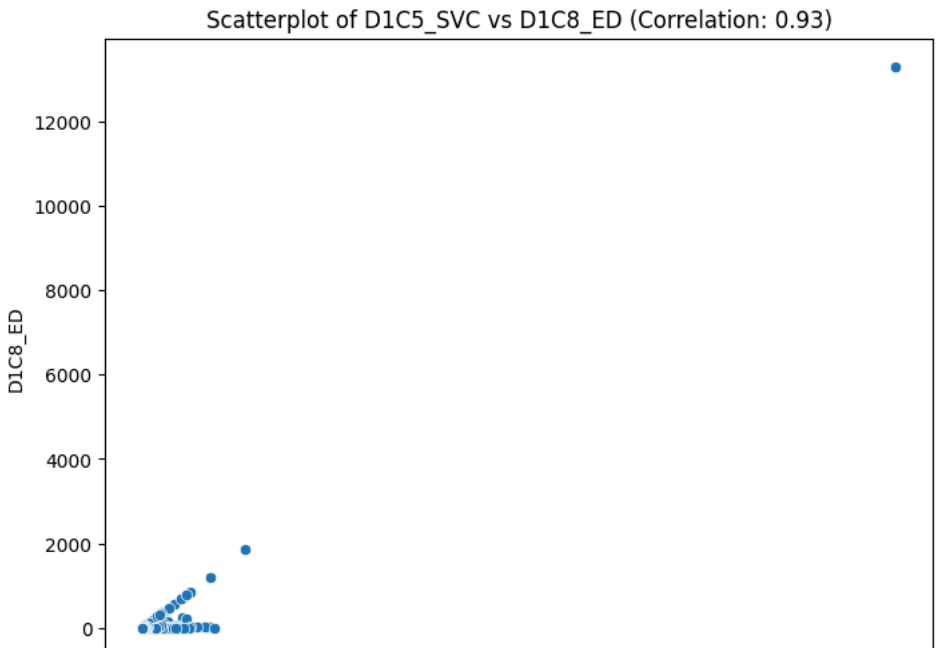

D1C5_SVC
D1C
0
2000
4000
6000
8000
10000
12000
14000

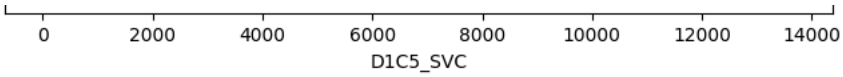

Scatterplot of D1C vs D1C8_ED (Correlation: 0.67)
D1C8_ED
D1C
0
2000
4000
6000
8000
10000
12000
14000

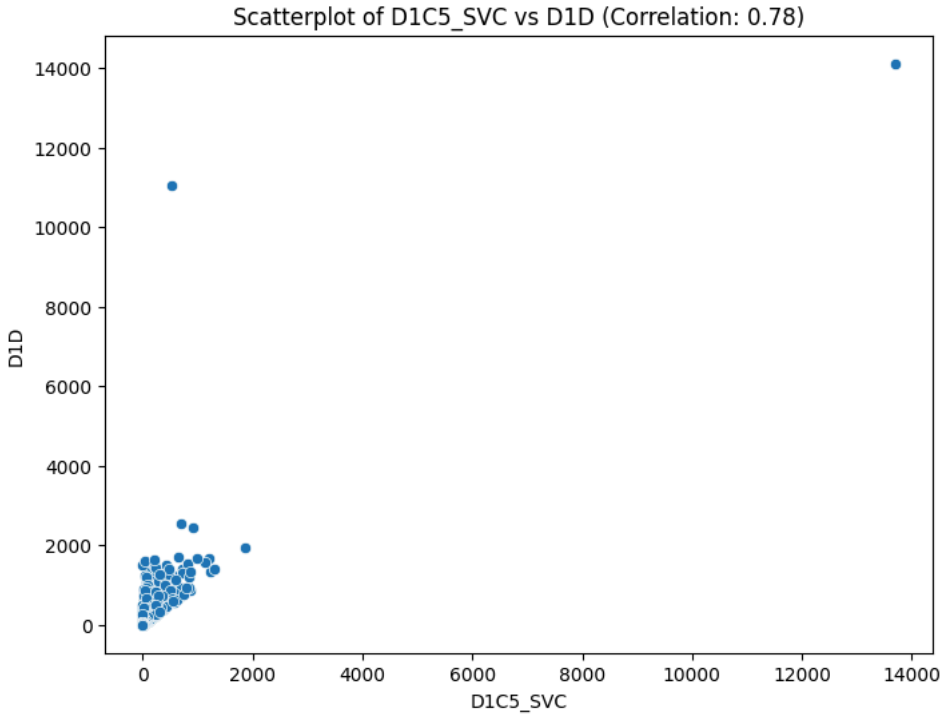

Scatterplot of D1C vs D1D (Correlation: 0.96)
D1D
D1C
0
2000
4000
6000
8000
10000
12000
14000

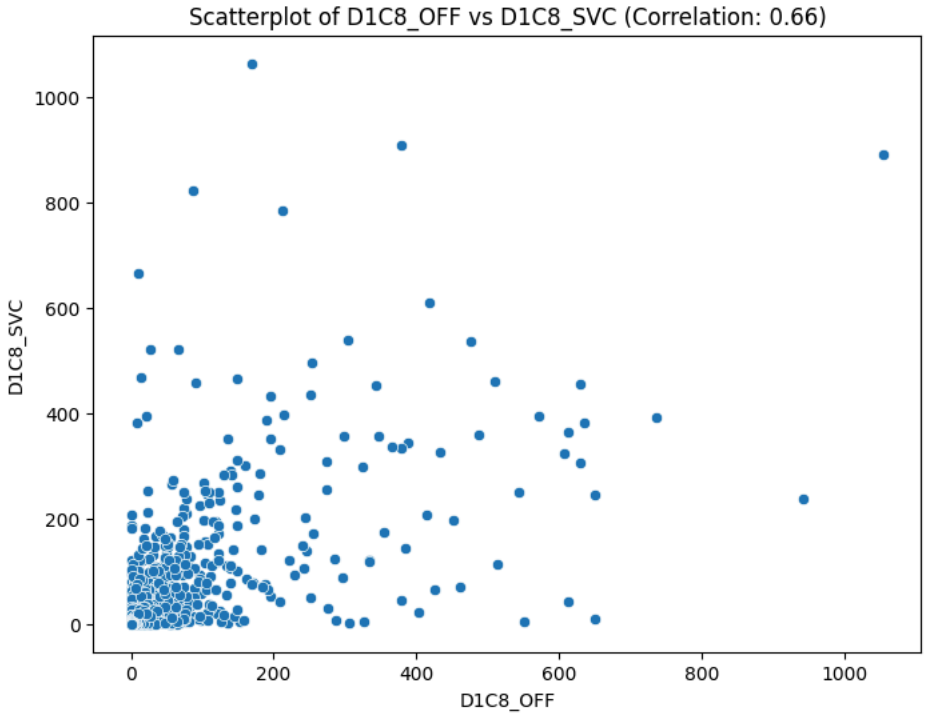

Scatterplot of D1C5_OFF vs D1C8_PUB (Correlation: 0.94)
D1C8_PUB
D1C5_OFF
0
2000
4000
6000
8000
10000

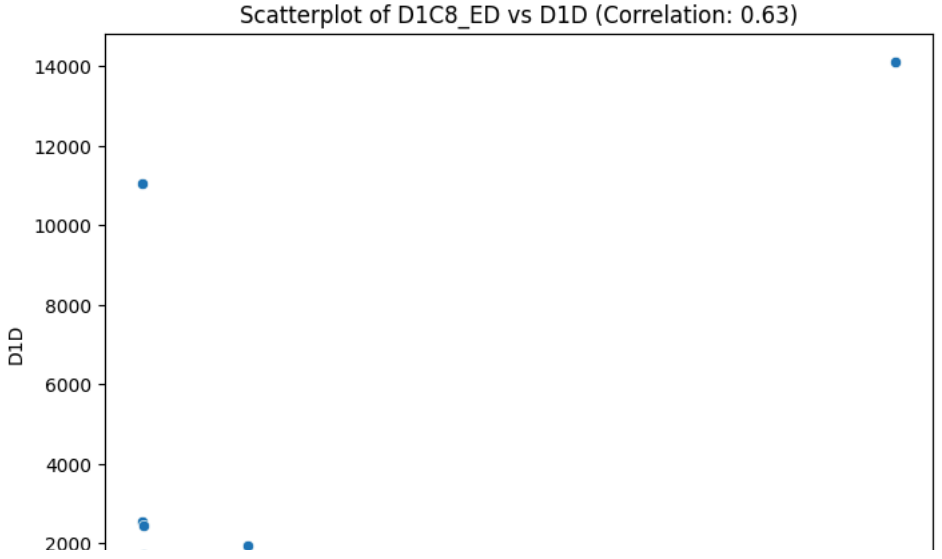

Scatterplot of D1C5_OFF vs D1D (Correlation: 0.61)
D1D
D1C5_OFF
0
2000
4000
6000
8000
10000
12000
14000

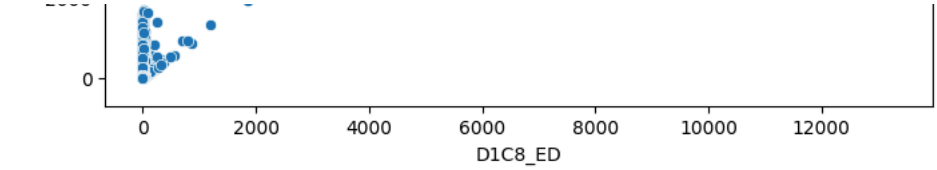

Scatterplot of D1C5_SVC vs D1C8_ED (Correlation: 0.93)
D1C8_ED
0
2000
4000
6000
8000
10000
12000

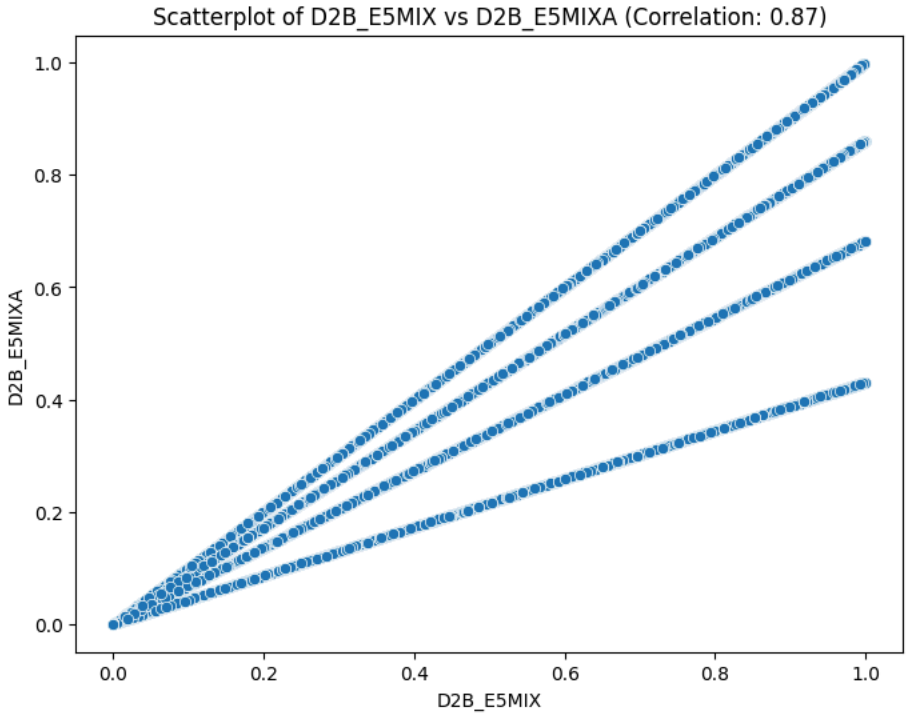

0
2000
4000
6000
8000
10000
12000
14000
D1C5_SVC

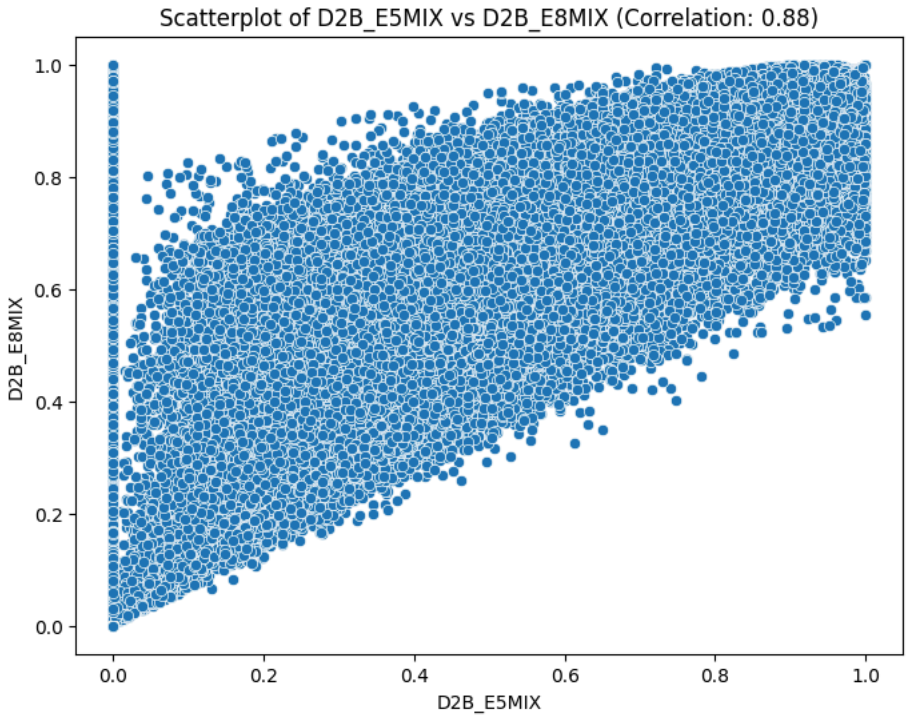

Scatterplot of D1C5_SVC vs D1D (Correlation: 0.78)
D1D
D1C5_SVC
0
2000
4000
6000
8000
10000
12000
14000

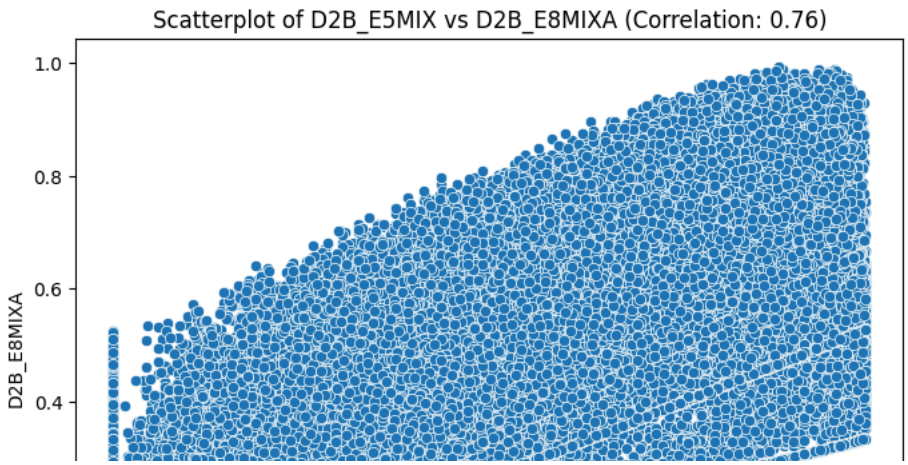

Scatterplot of D1C8_OFF vs D1C8_SVC (Correlation: 0.66)
D1C8_SVC
D1C8_OFF
0
200
400
600
800
1000

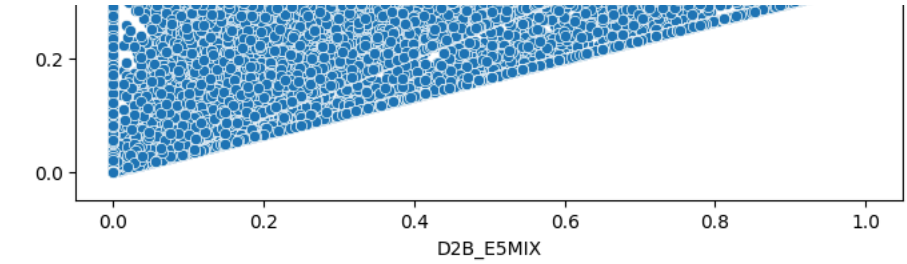

Scatterplot of D1C8_ED vs D1D (Correlation: 0.63)
D1D
2000
4000
6000
8000
10000
12000
14000

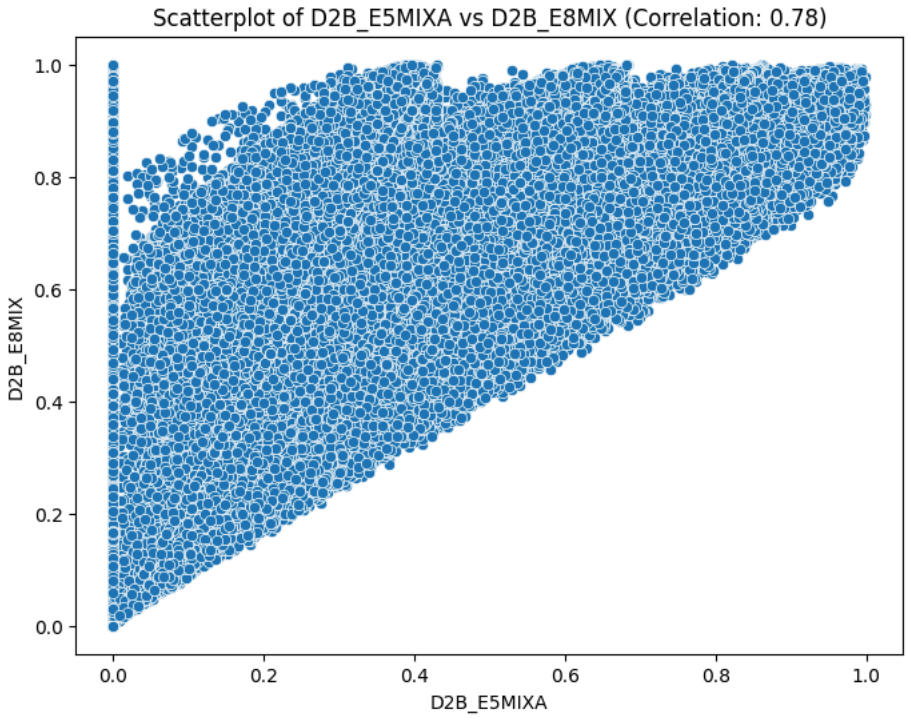

0
0
2000
4000
6000
8000
10000
12000
D1C8_ED

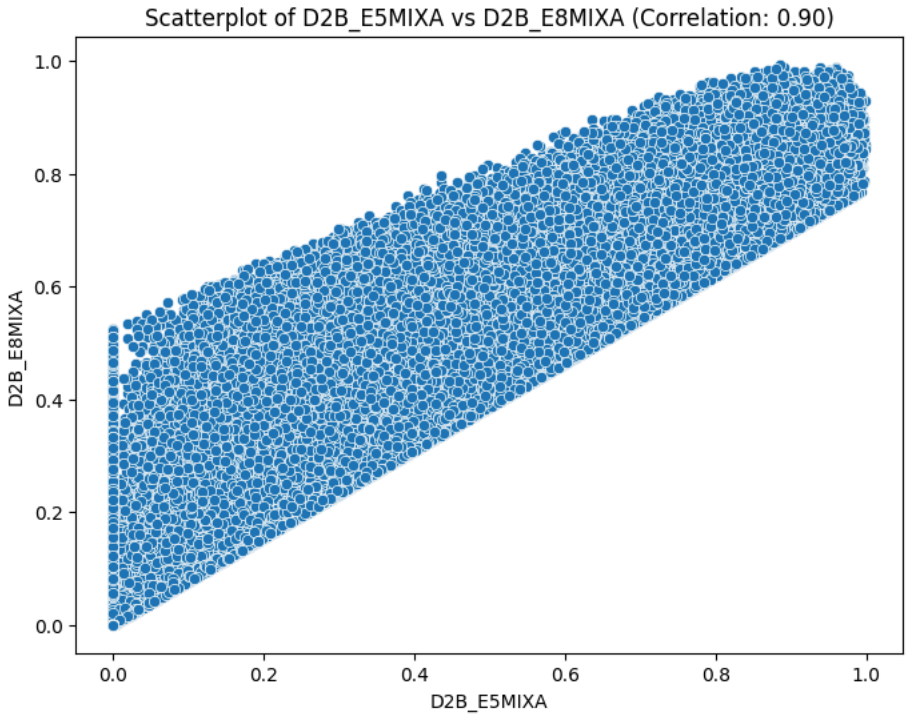

Scatterplot of D2B_E5MIX vs D2B_E5MIXA (Correlation: 0.87)
D2B_E5MIXA
D2B_E5MIX
0.0
0.2
0.4
0.6
0.8
1.0

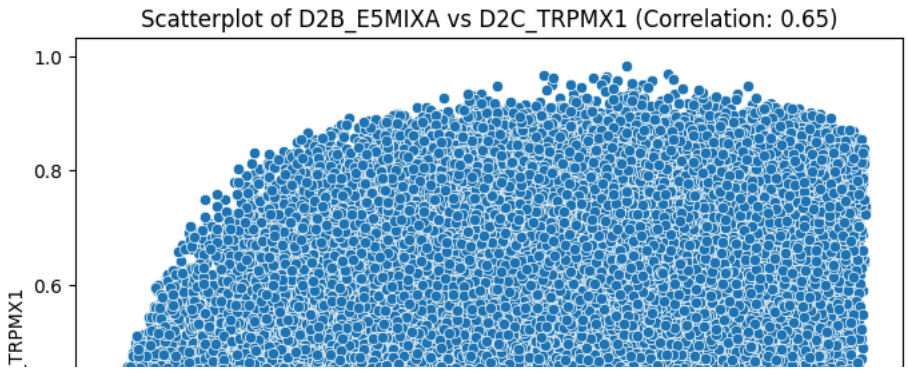

Scatterplot of D2B_E5MIX vs D2B_E8MIX (Correlation: 0.88)
D2B_E8MIX
D2B_E5MIX
0.0
0.2
0.4
0.6
0.8
1.0

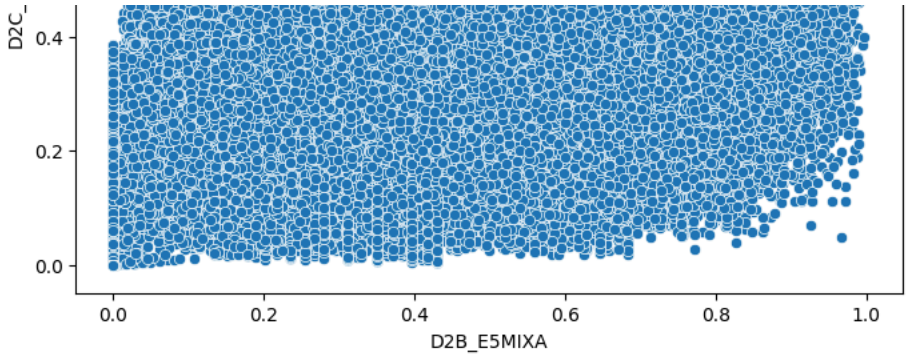

Scatterplot of D2B_E5MIX vs D2B_E8MIXA (Correlation: 0.76)
D2B_E8MIXA
0.4
0.6
0.8
1.0

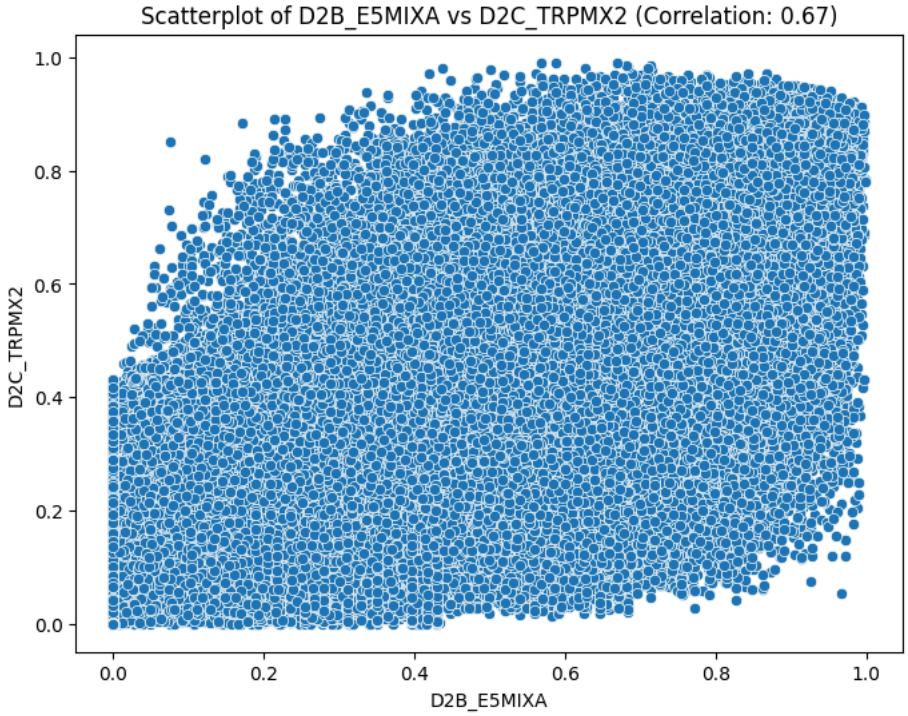

0.0
0.2
0.0
0.2
0.4
0.6
0.8
1.0
D2B_E5MIX

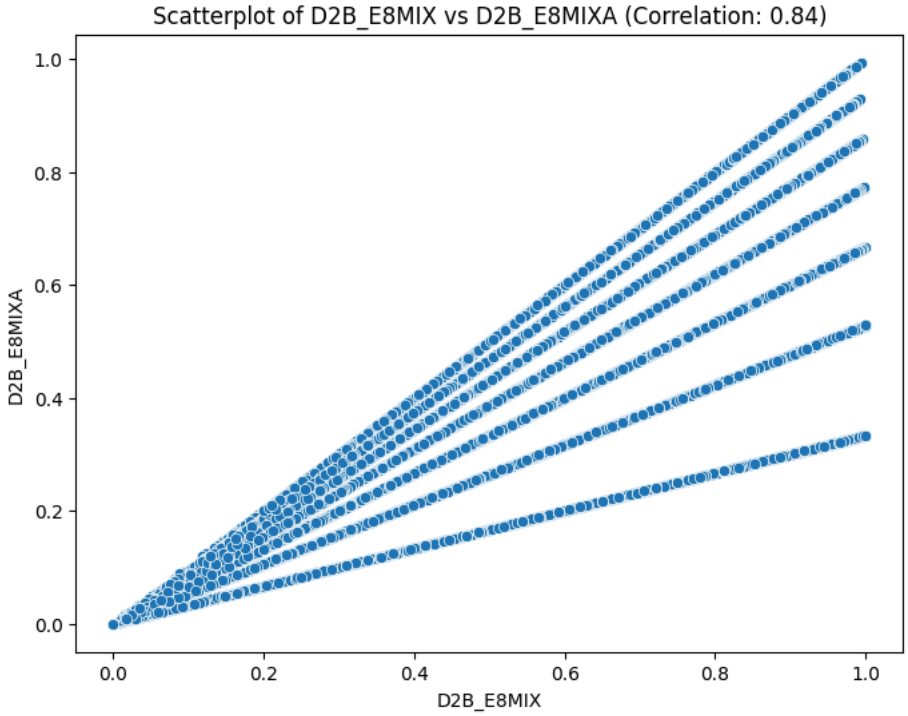

Scatterplot of D2B_E5MIXA vs D2B_E8MIX (Correlation: 0.78)
D2B_E8MIX
D2B_E5MIXA
0.0
0.2
0.4
0.6
0.8
1.0

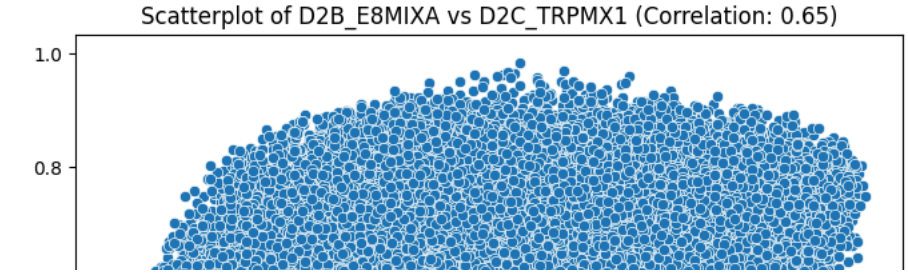

Scatterplot of D2B_E5MIXA vs D2B_E8MIXA (Correlation: 0.90)
D2B_E8MIXA
D2B_E5MIXA
0.0
0.2
0.4
0.6
0.8
1.0

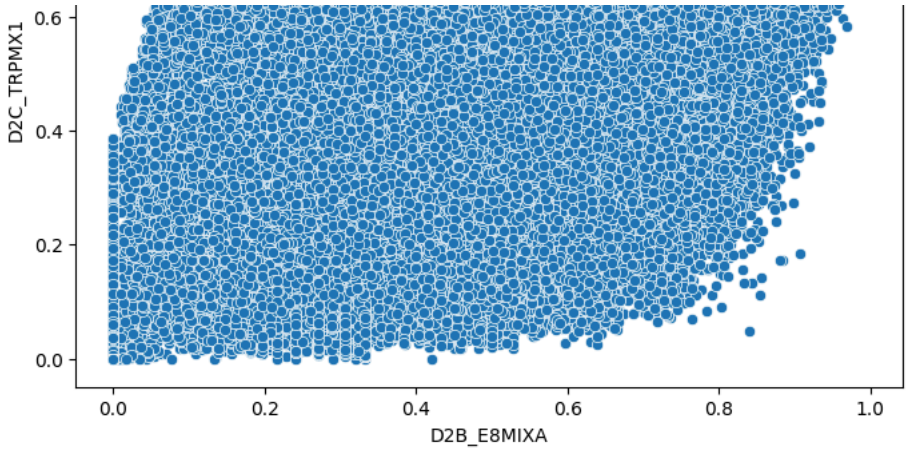

Scatterplot of D2B_E5MIXA vs D2C_TRPMX1 (Correlation: 0.65)
TRPMX1
0.6
0.8
1.0

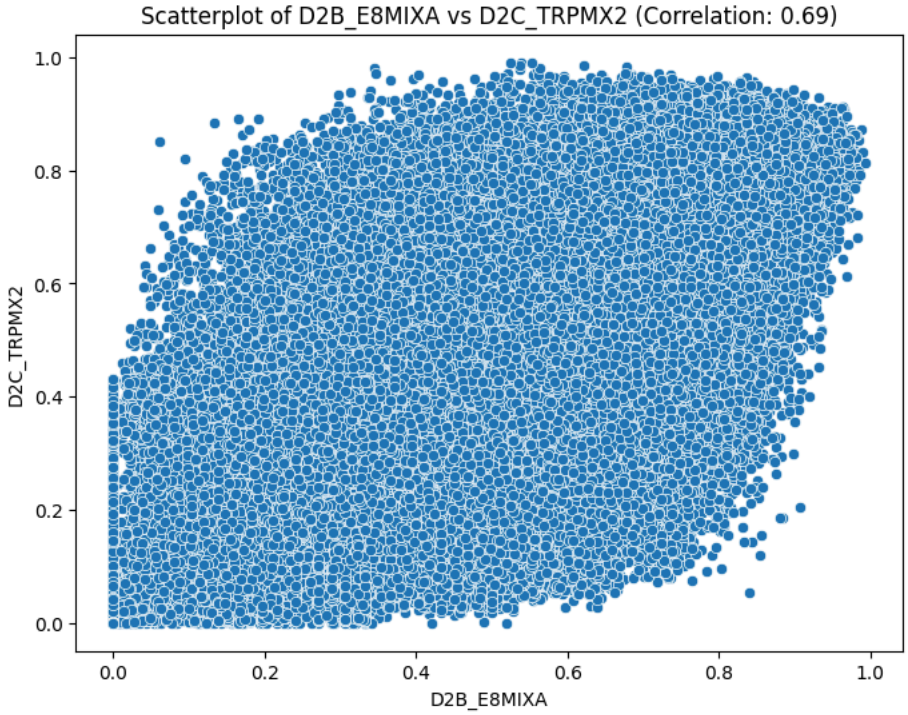

D2C_
0.0
0.2
0.4
0.0
0.2
0.4
0.6
0.8
1.0
D2B_E5MIXA

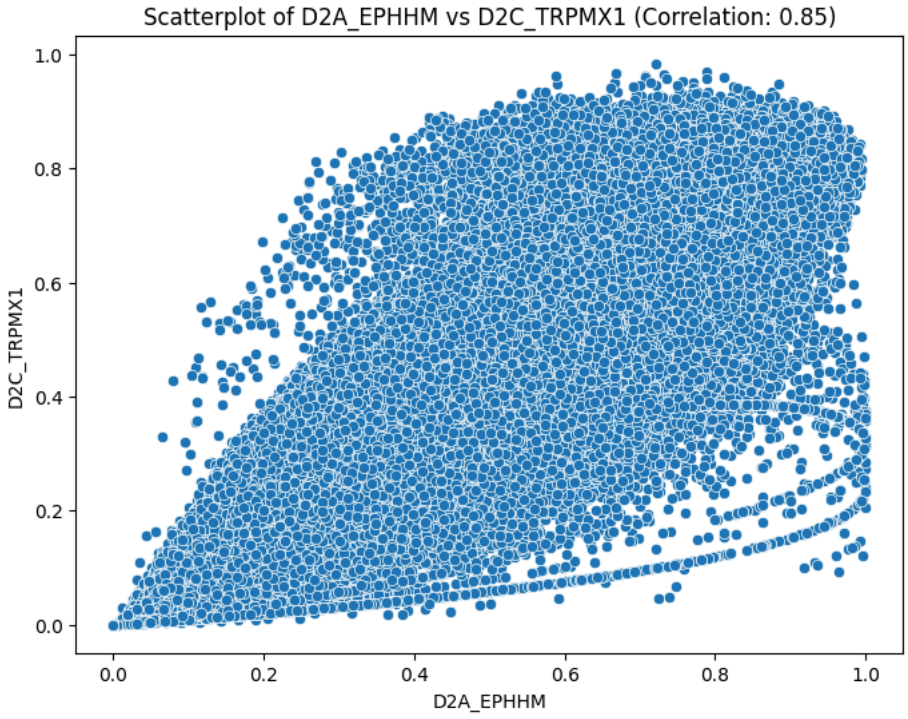

Scatterplot of D2B_E5MIXA vs D2C_TRPMX2 (Correlation: 0.67)
D2C_TRPMX2
D2B_E5MIXA
0.0
0.2
0.4
0.6
0.8
1.0

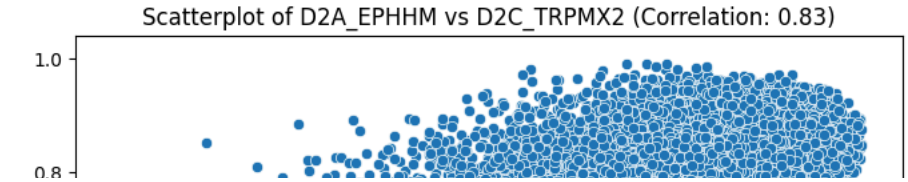

Scatterplot of D2B_E8MIX vs D2B_E8MIXA (Correlation: 0.84)
D2B_E8MIXA
D2B_E8MIX
0.0
0.2
0.4
0.6
0.8
1.0

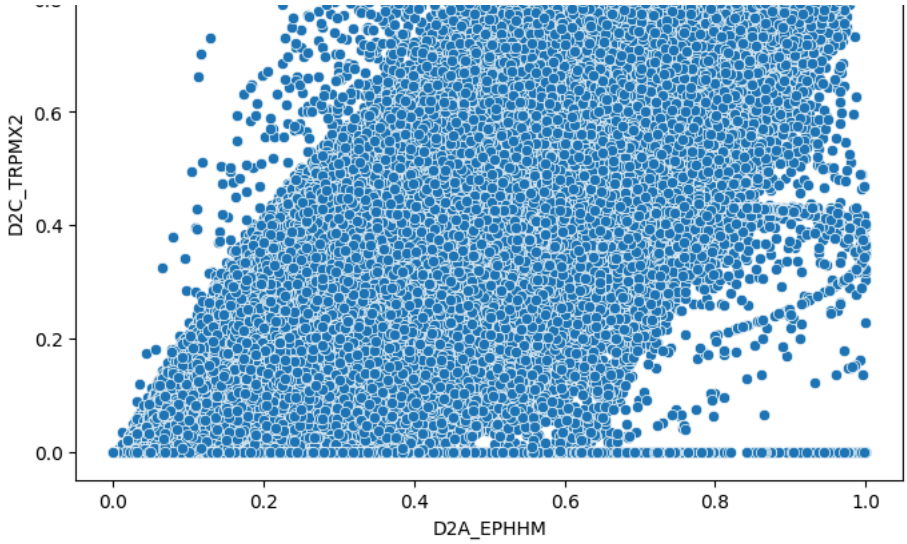

Scatterplot of D2B_E8MIXA vs D2C_TRPMX1 (Correlation: 0.65)
0.8
1.0

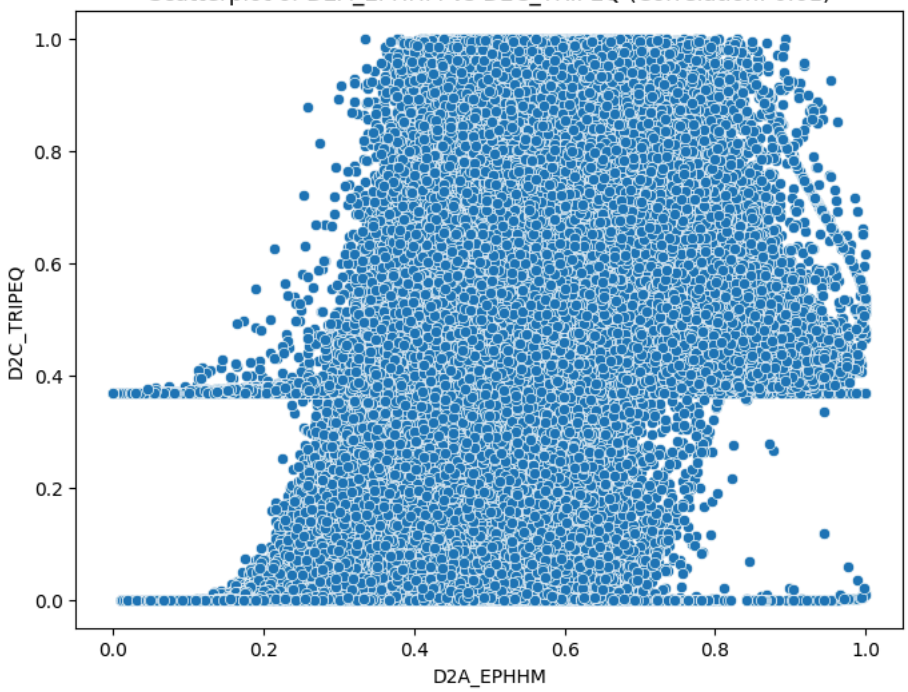

D2C_TRPMX1
D2B_E8MIXA
0.0
0.2
0.4
0.6
0.8
1.0

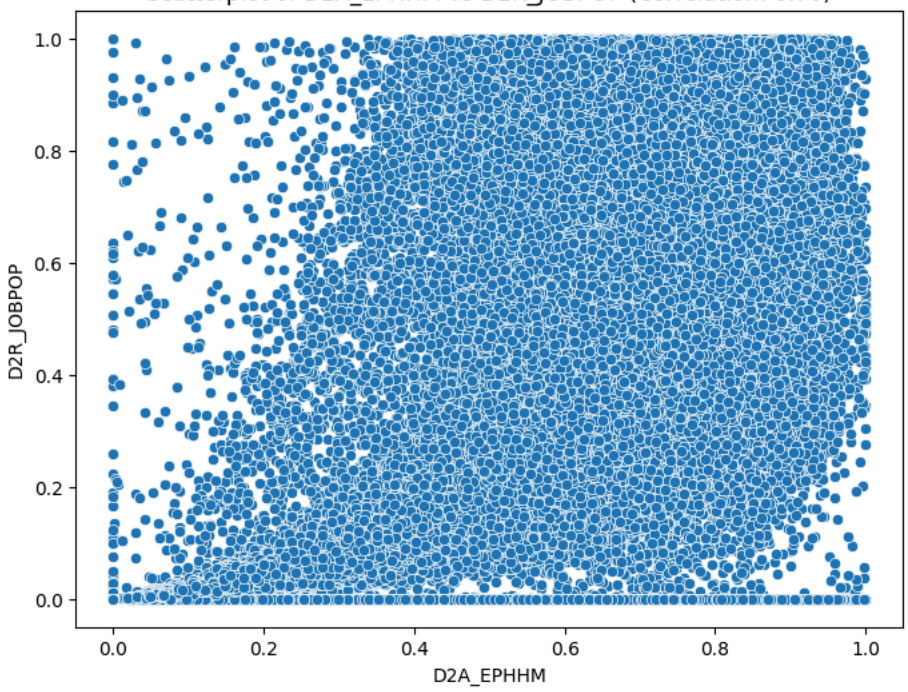

Scatterplot of D2B_E8MIXA vs D2C_TRPMX2 (Correlation: 0.69)
D2C_TRPMX2
D2B_E8MIXA
0.0
0.2
0.4
0.6
0.8
1.0

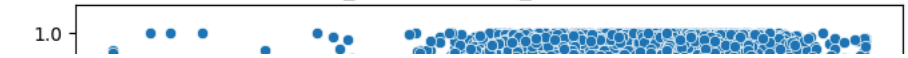

Scatterplot of D2A_EPHHM vs D2C_TRPMX1 (Correlation: 0.85)
D2C_TRPMX1
D2A_EPHHM
0.0
0.2
0.4
0.6
0.8
1.0

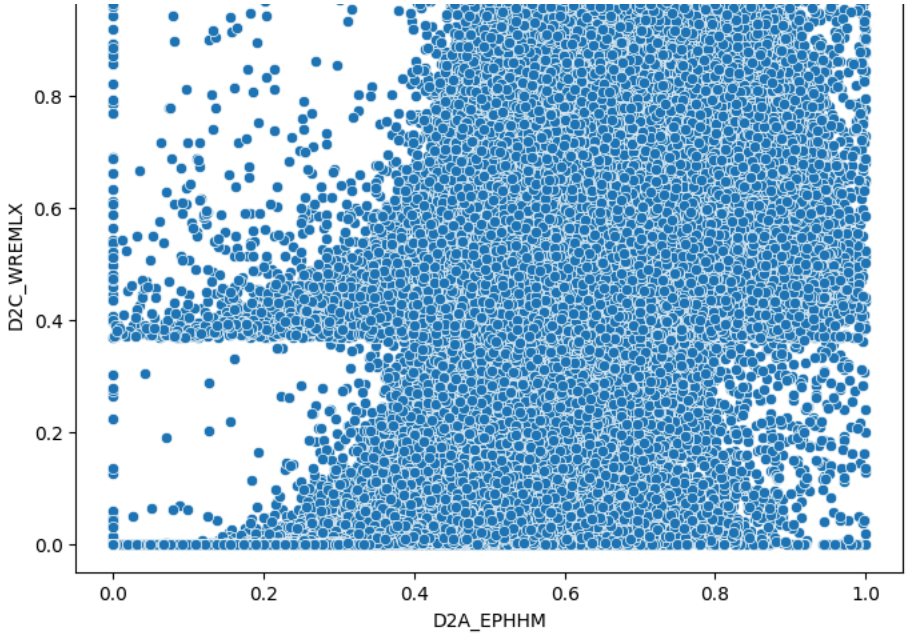

Scatterplot of D2A_EPHHM vs D2C_TRPMX2 (Correlation: 0.83)
1.0
0.8

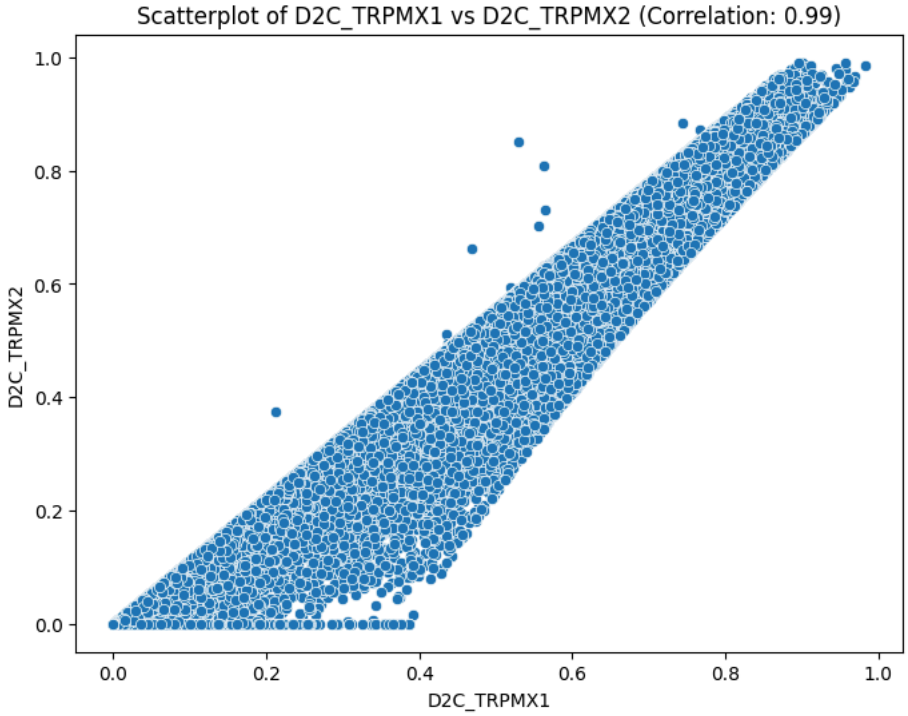

D2C_TRPMX2
D2A_EPHHM
0.0
0.2
0.4
0.6
0.8
1.0

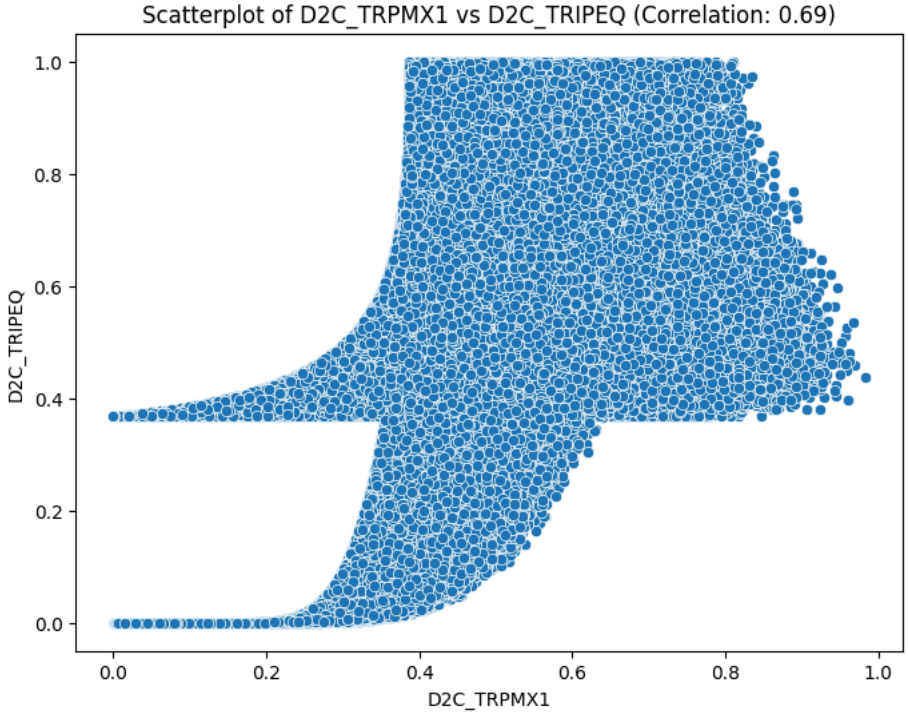

Scatterplot of D2A_EPHHM vs D2C_TRIPEQ (Correlation: 0.61)
D2C_TRIPEQ
D2A_EPHHM
0.0
0.2
0.4
0.6
0.8
1.0

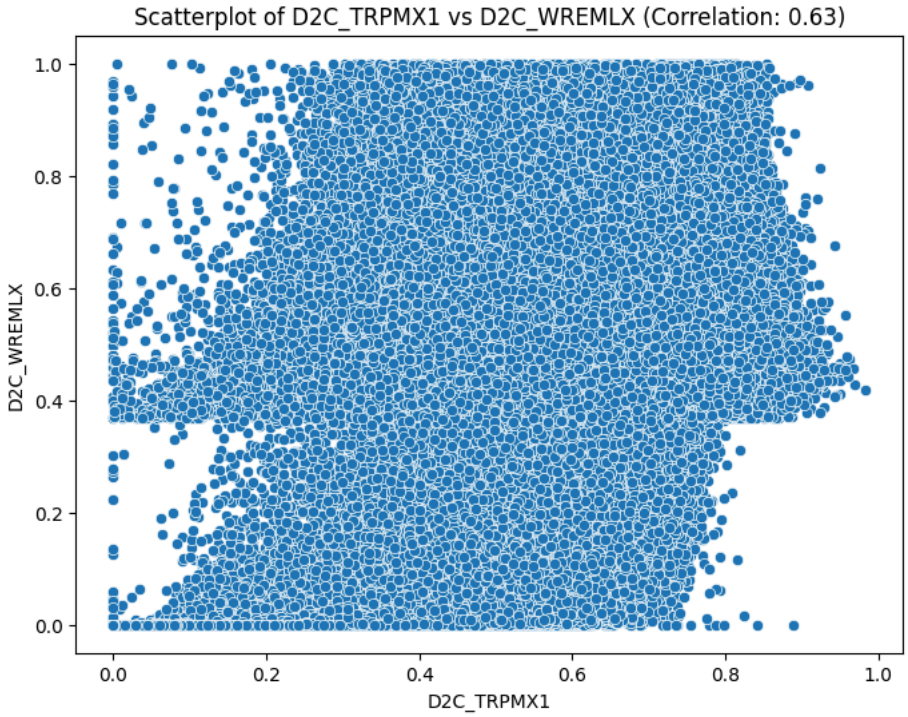

Scatterplot of D2A_EPHHM vs D2R_JOBPOP (Correlation: 0.70)
D2R_JOBPOP
D2A_EPHHM
0.0
0.2
0.4
0.6
0.8
1.0

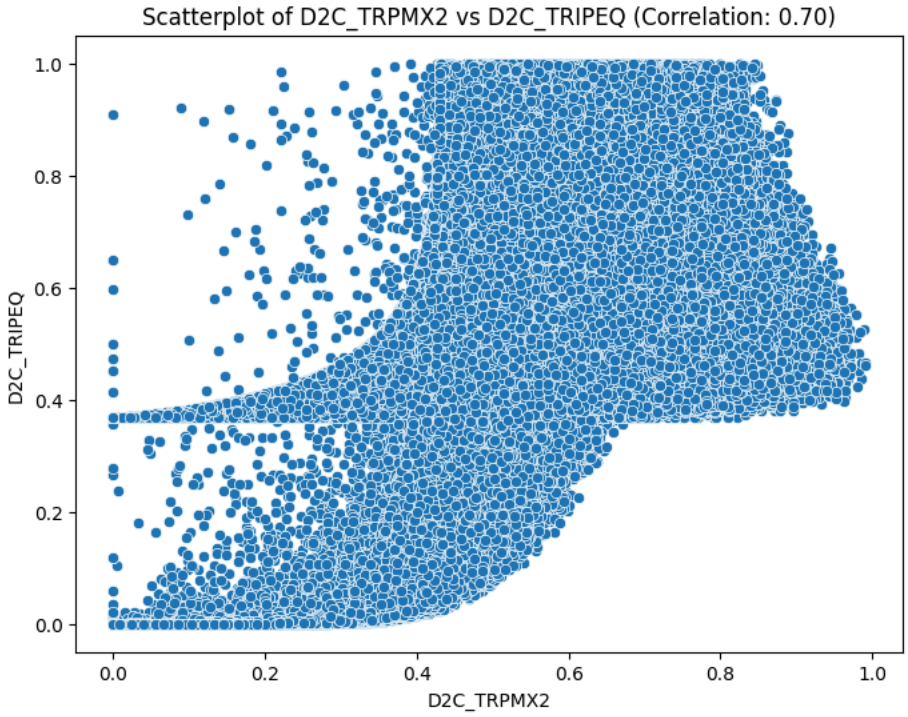

Scatterplot of D2A_EPHHM vs D2C_WREMLX (Correlation: 0.75)
1.0

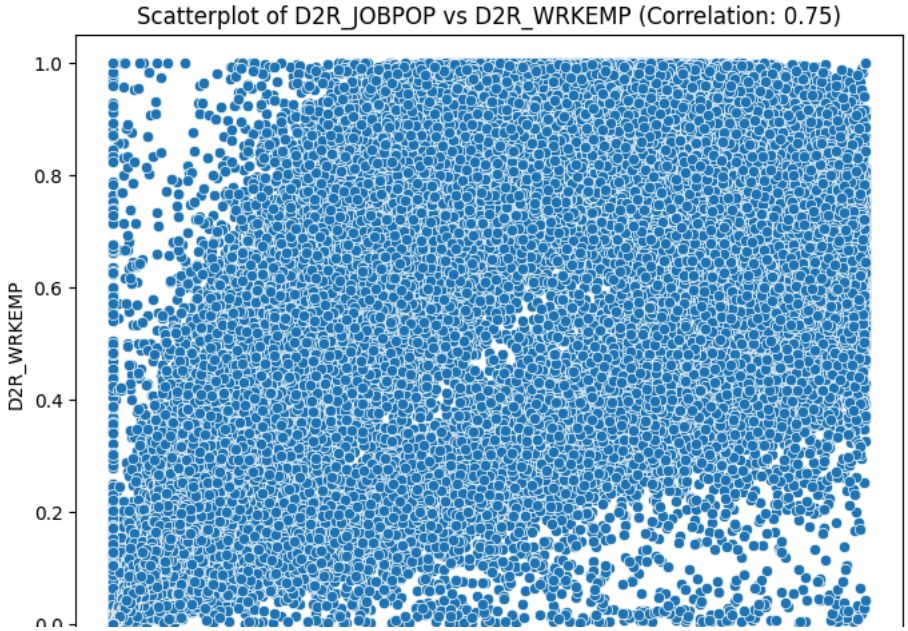

D2C_WREMLX
D2A_EPHHM
0.0
0.2
0.4
0.6
0.8
1.0

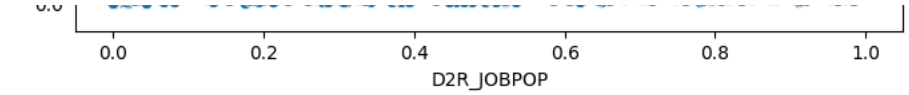

Scatterplot of D2C_TRPMX1 vs D2C_TRPMX2 (Correlation: 0.99)
D2C_TRPMX2
D2C_TRPMX1
0.0
0.2
0.4
0.6
0.8
1.0

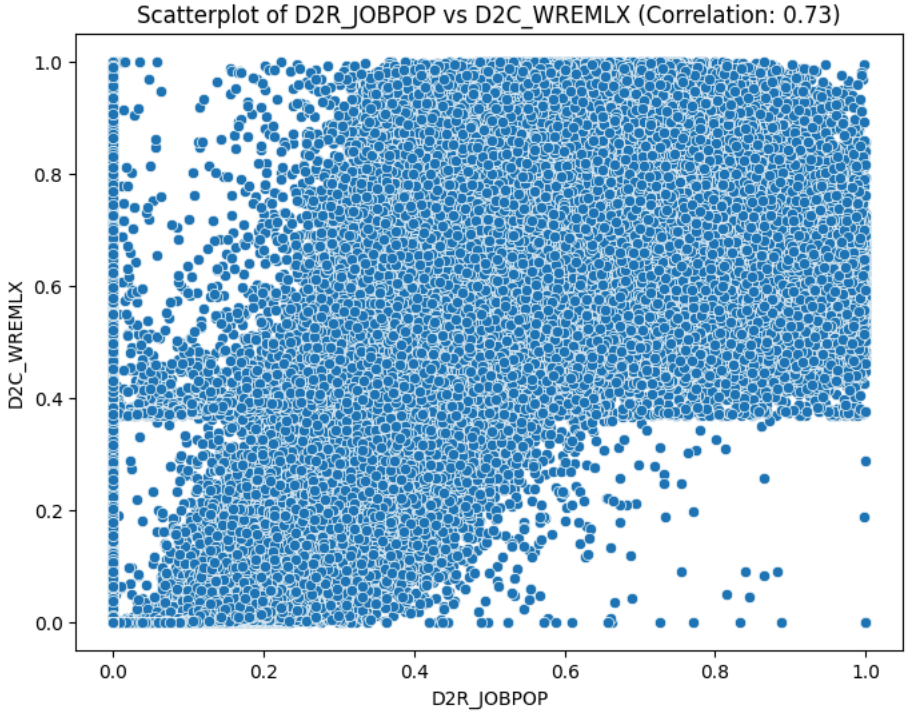

Scatterplot of D2C_TRPMX1 vs D2C_TRIPEQ (Correlation: 0.69)
D2C_TRIPEQ
D2C_TRPMX1
0.0
0.2
0.4
0.6
0.8
1.0

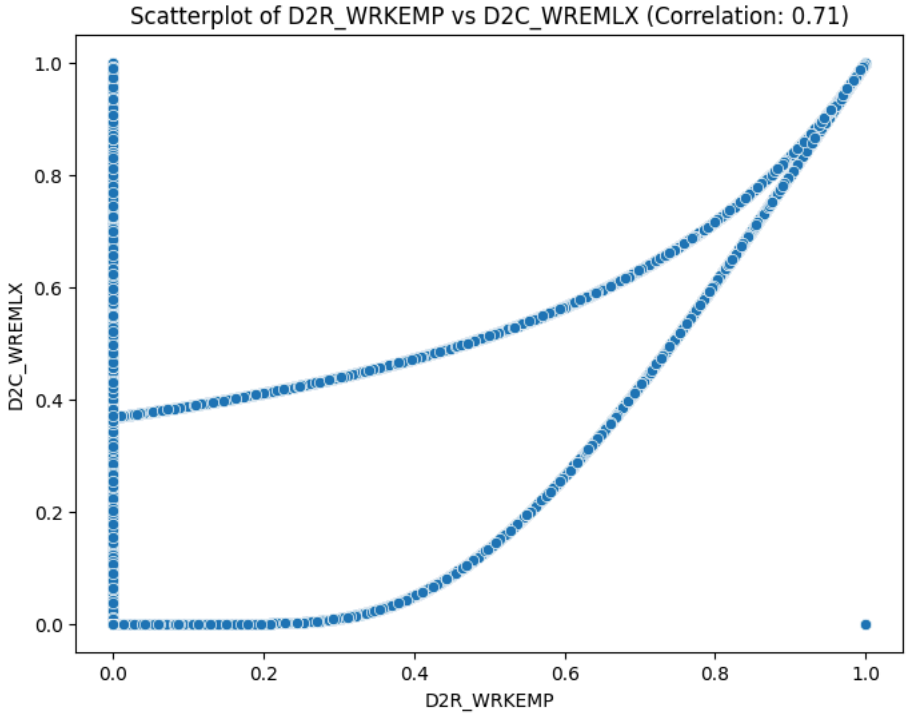

Scatterplot of D2C_TRPMX1 vs D2C_WREMLX (Correlation: 0.63)
D2C_WREMLX
D2C_TRPMX1
0.0
0.2
0.4
0.6
0.8
1.0

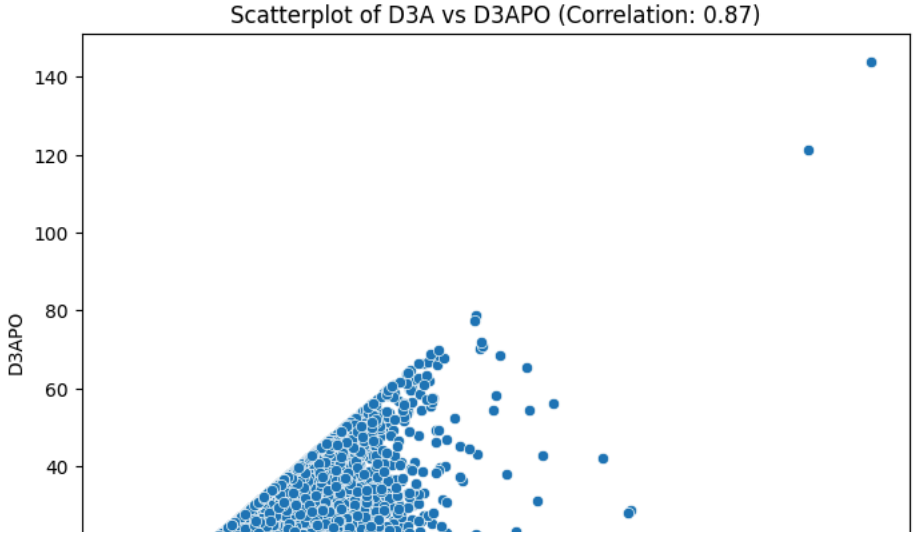

Scatterplot of D2C_TRPMX2 vs D2C_TRIPEQ (Correlation: 0.70)
D2C_TRIPEQ
D2C_TRPMX2
0.0
0.2
0.4
0.6
0.8
1.0

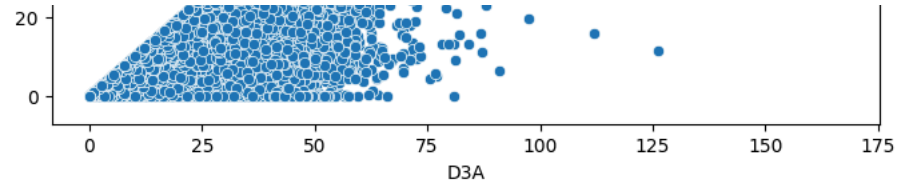

Scatterplot of D2R_JOBPOP vs D2R_WRKEMP (Correlation: 0.75)
D2R_WRKEMP
0.0
0.2
0.4
0.6
0.8
1.0

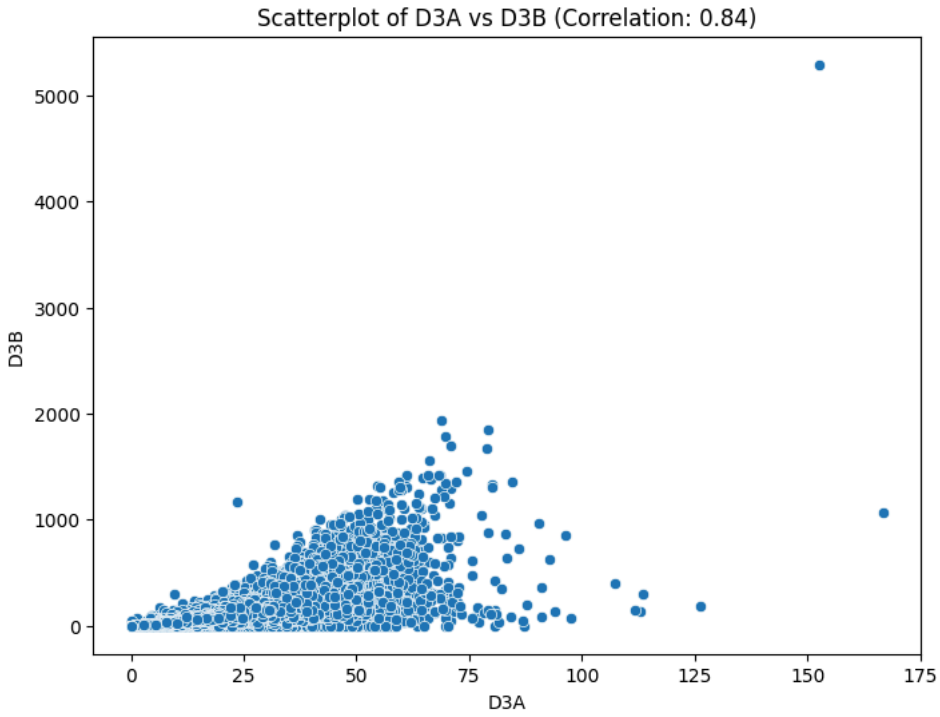

0.0
0.2
0.4
0.6
0.8
1.0
D2R_JOBPOP

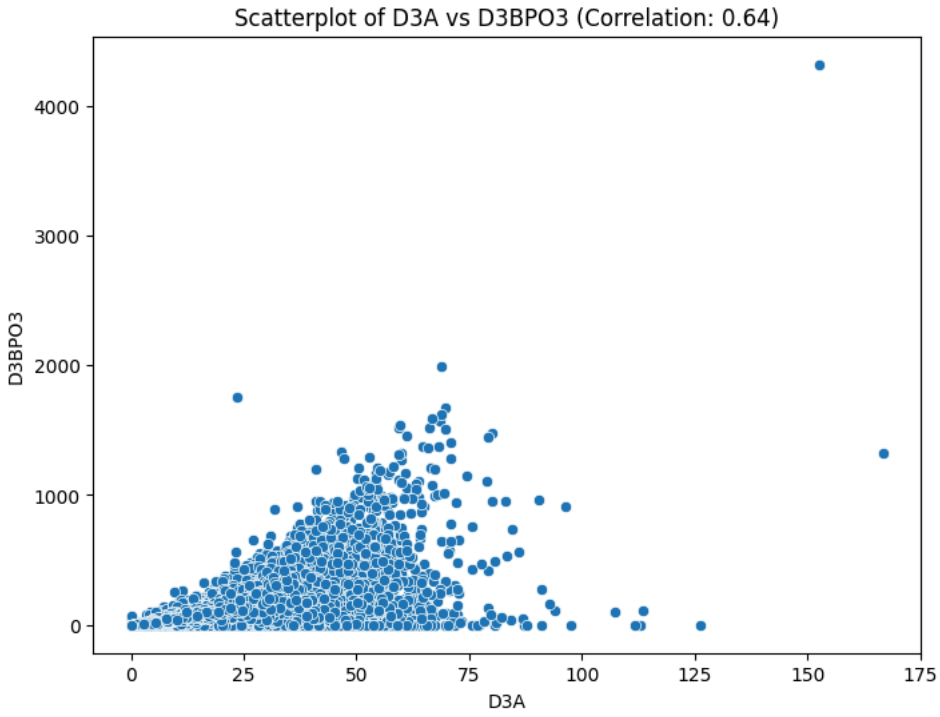

Scatterplot of D2R_JOBPOP vs D2C_WREMLX (Correlation: 0.73)
D2C_WREMLX
D2R_JOBPOP
0.0
0.2
0.4
0.6
0.8
1.0

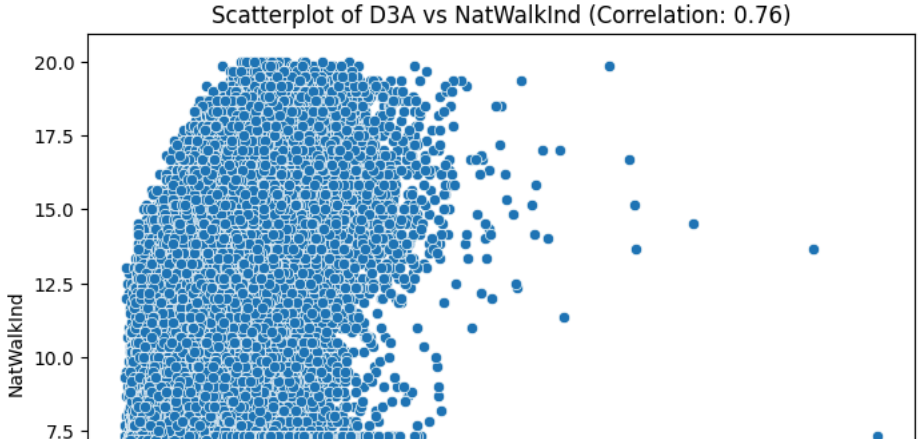

Scatterplot of D2R_WRKEMP vs D2C_WREMLX (Correlation: 0.71)
D2C_WREMLX
D2R_WRKEMP
0.0
0.2
0.4
0.6
0.8
1.0

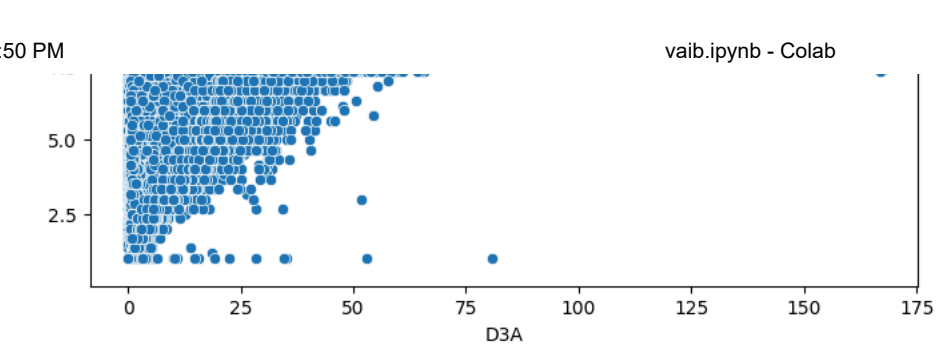

Scatterplot of D3A vs D3APO (Correlation: 0.87)
D3APO
40
60
80
100
120
140

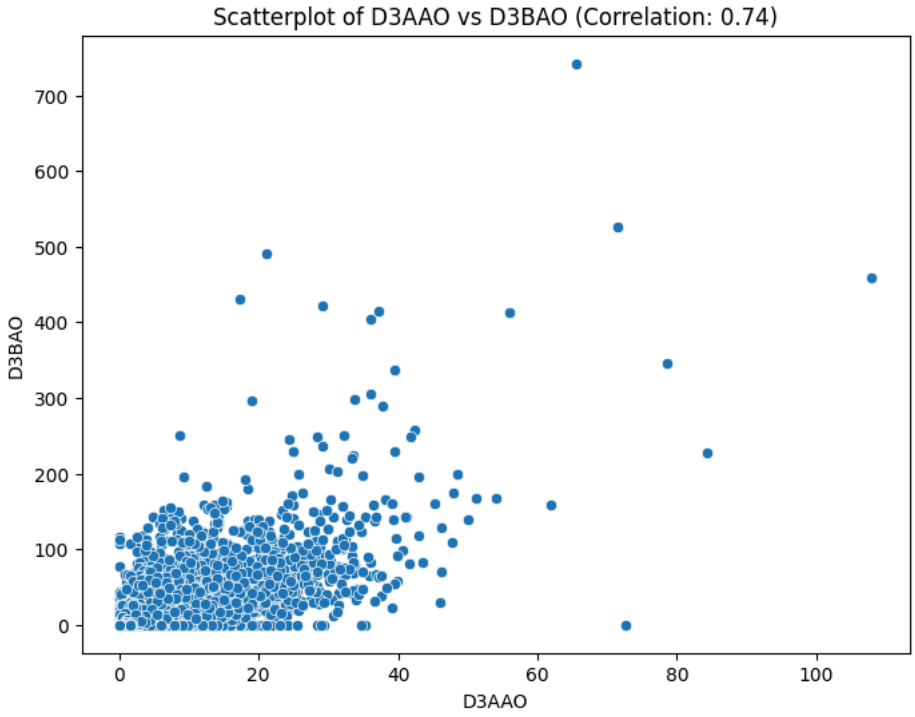

20
0
0
25
50
75
100
125
150
175
D3A

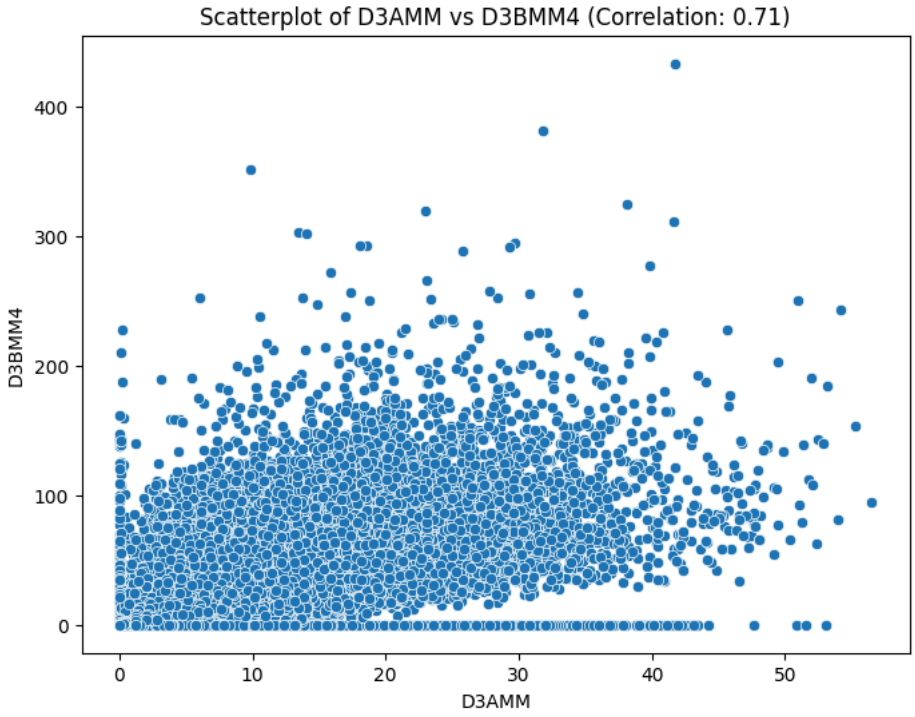

Scatterplot of D3A vs D3B (Correlation: 0.84)
5000
4000
3000
2000
1000
0
D3B
0
25
50
75
100
125
150
175
D3A

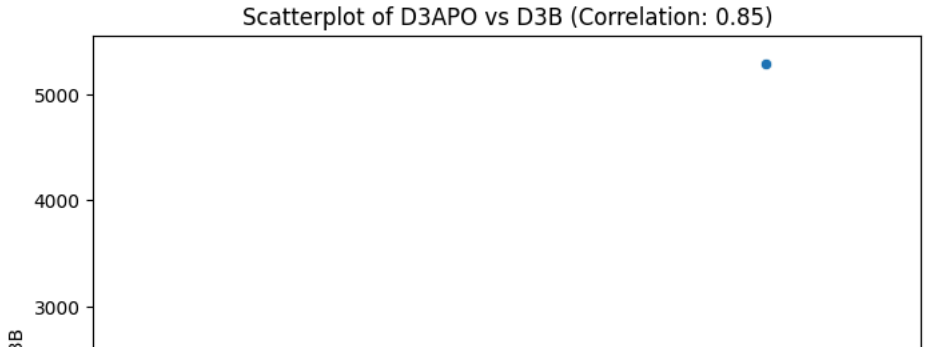

Scatterplot of D3A vs D3BPO3 (Correlation: 0.64)
4000
3000
2000
1000
0
D3BPO3
0
25
50
75
100
125
150
175
D3A

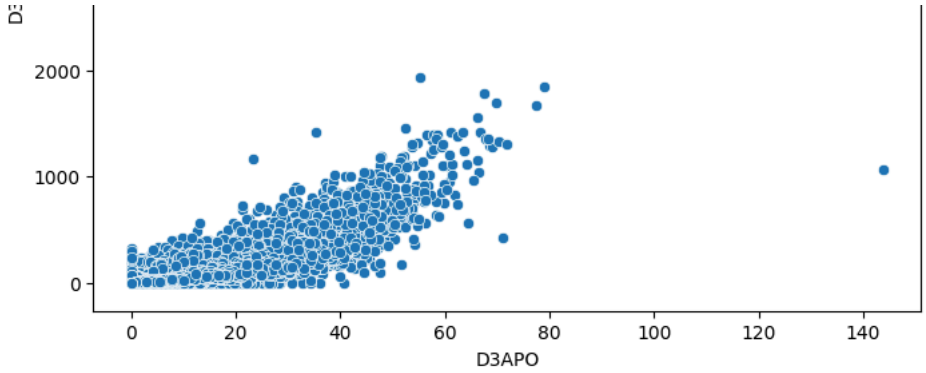

Scatterplot of D3A vs NatWalkInd (Correlation: 0.76)
20.0
17.5
15.0
12.5
10.0
7.5
NatWalkInd

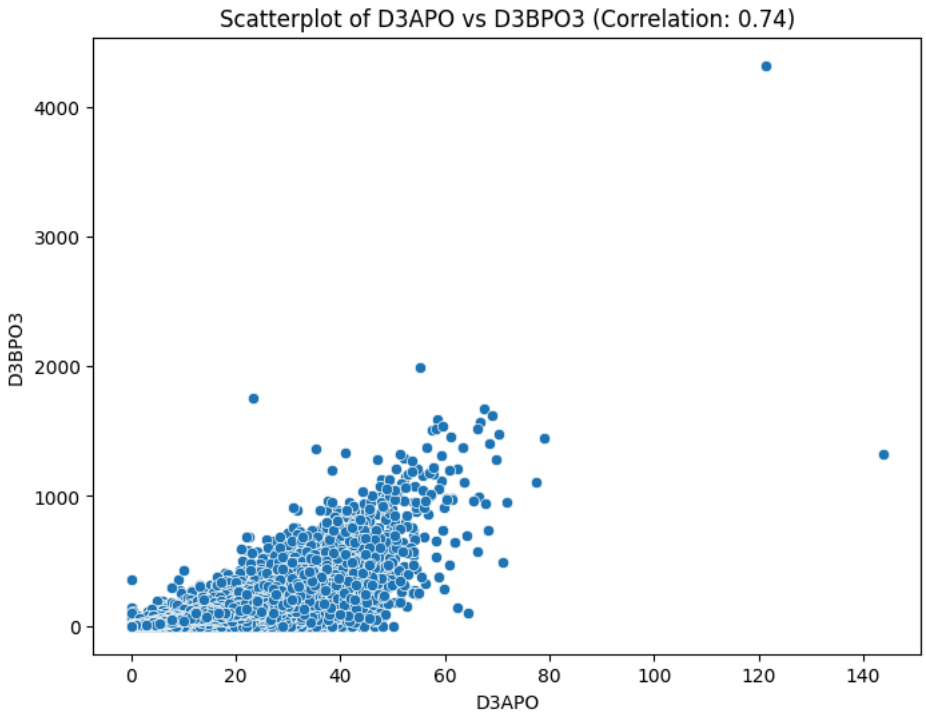

5.0
2.5
0
25
50
75
100
125
150
175
D3A

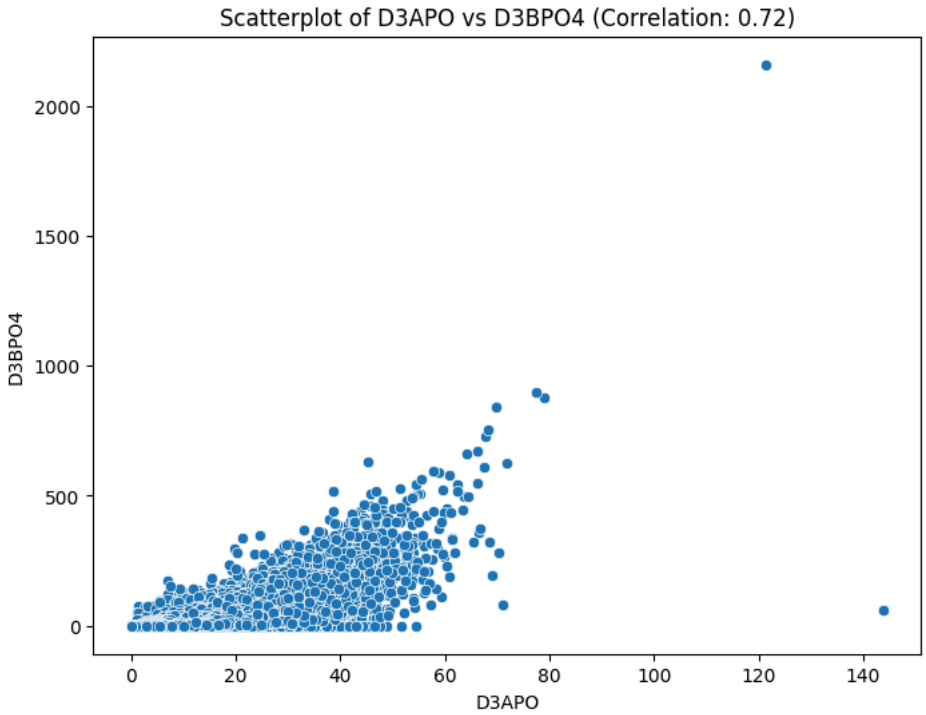

Scatterplot of D3AAO vs D3BAO (Correlation: 0.74)
D3BAO
D3AAO
0
100
200
300
400
500
600
700
0
20
40
60
80
100

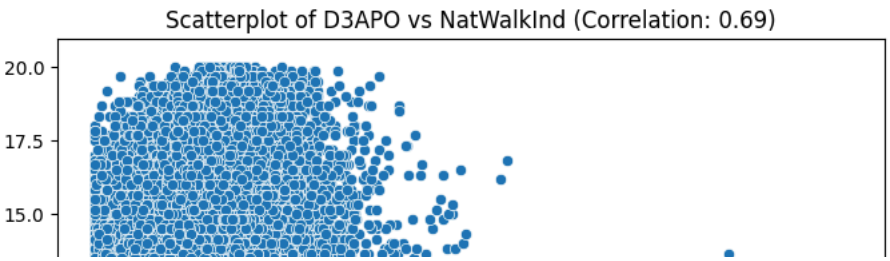

Scatterplot of D3AMM vs D3BMM4 (Correlation: 0.71)
D3BMM4
D3AMM
0
100
200
300
400
0
10
20
30
40
50

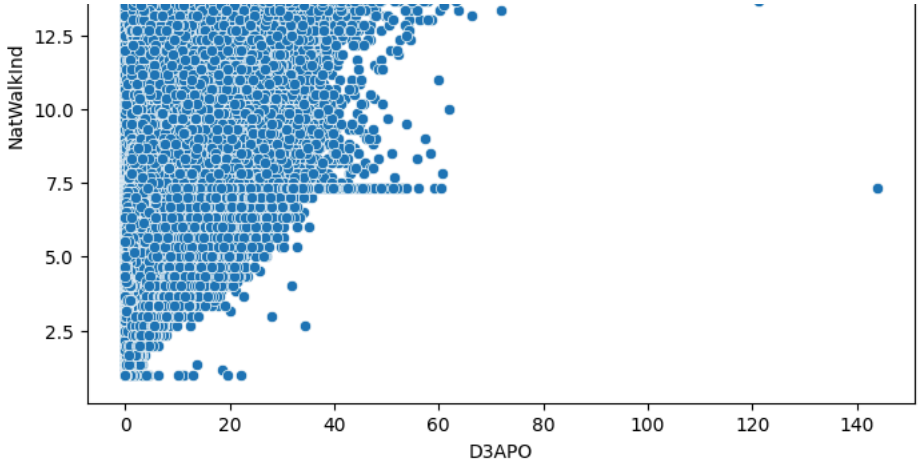

Scatterplot of D3APO vs D3B (Correlation: 0.85)
3000
4000
5000

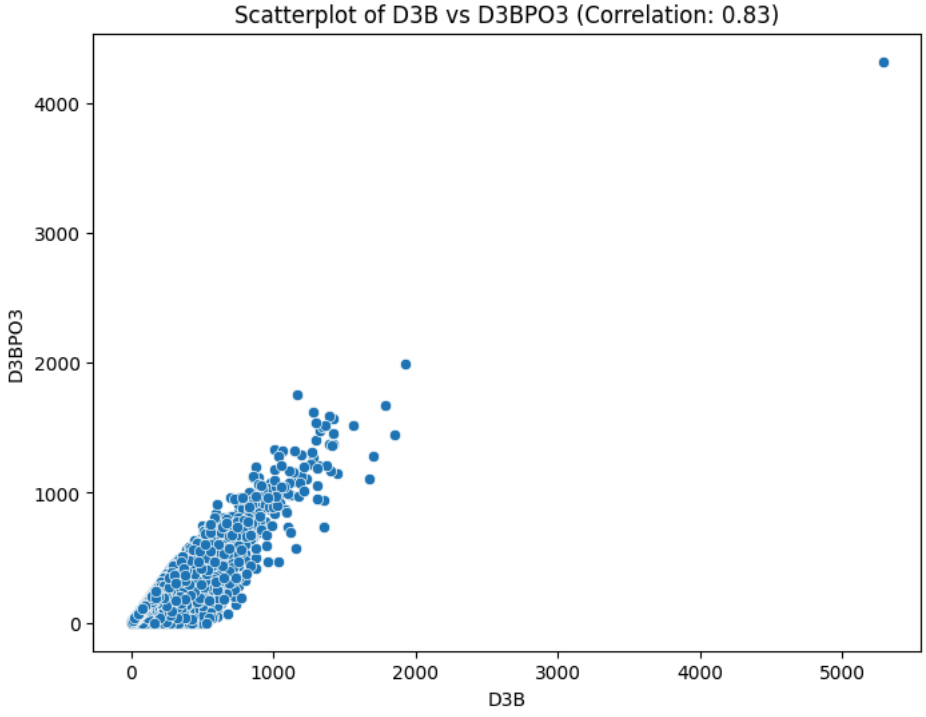

2000
1000
0
0
20
40
60
80
100
120
140
D3APO

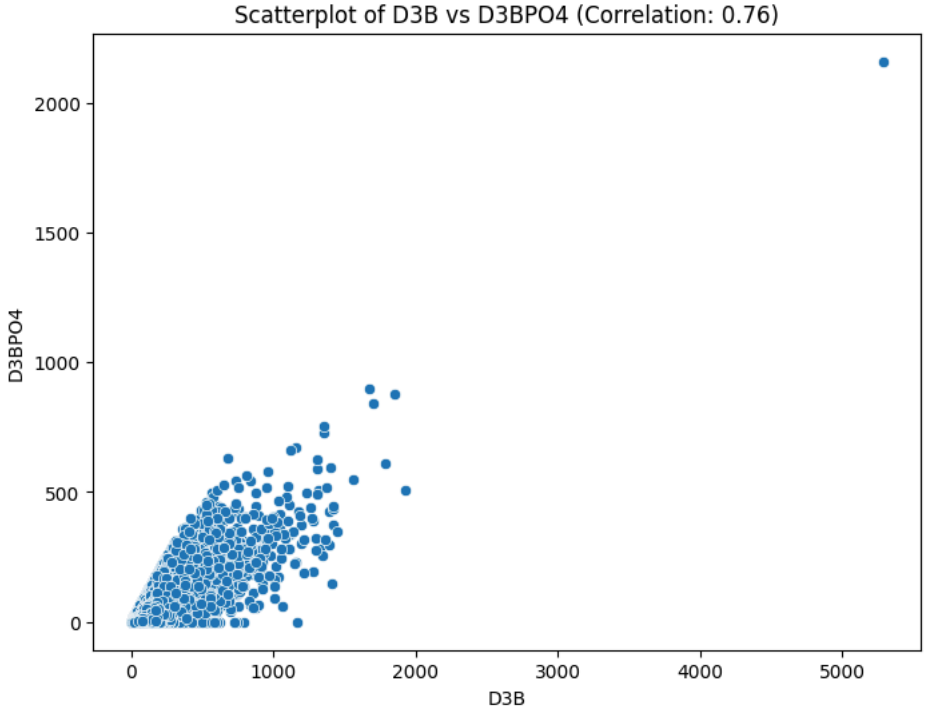

Scatterplot of D3APO vs D3BPO3 (Correlation: 0.74)
4000
3000
2000
1000
0
D3BPO3
0
20
40
60
80
100
120
140
D3APO

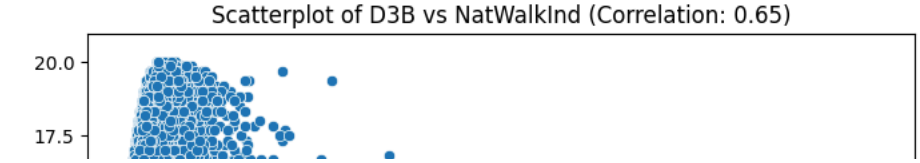

Scatterplot of D3APO vs D3BPO4 (Correlation: 0.72)
2000
1500
1000
500
0
D3BPO4
0
20
40
60
80
100
120
140
D3APO

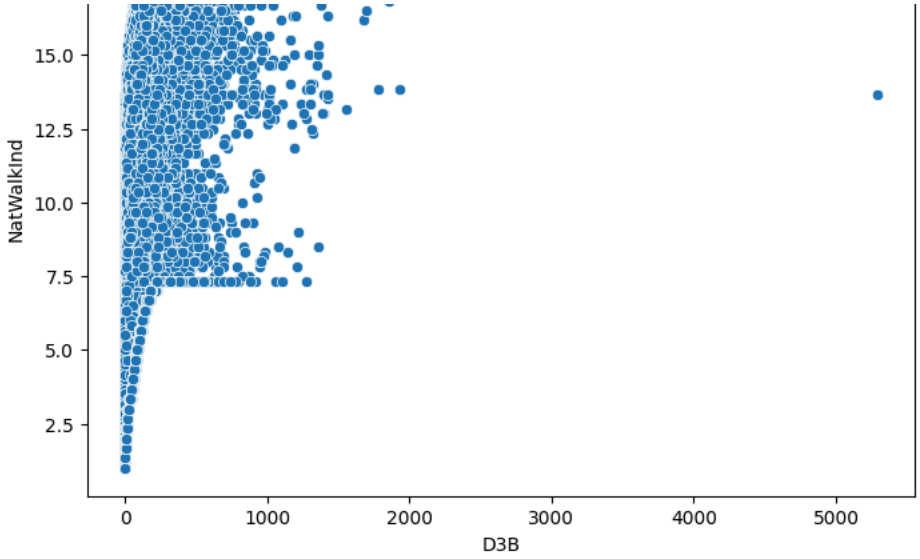

Scatterplot of D3APO vs NatWalkInd (Correlation: 0.69)
20.0
17.5
15.0

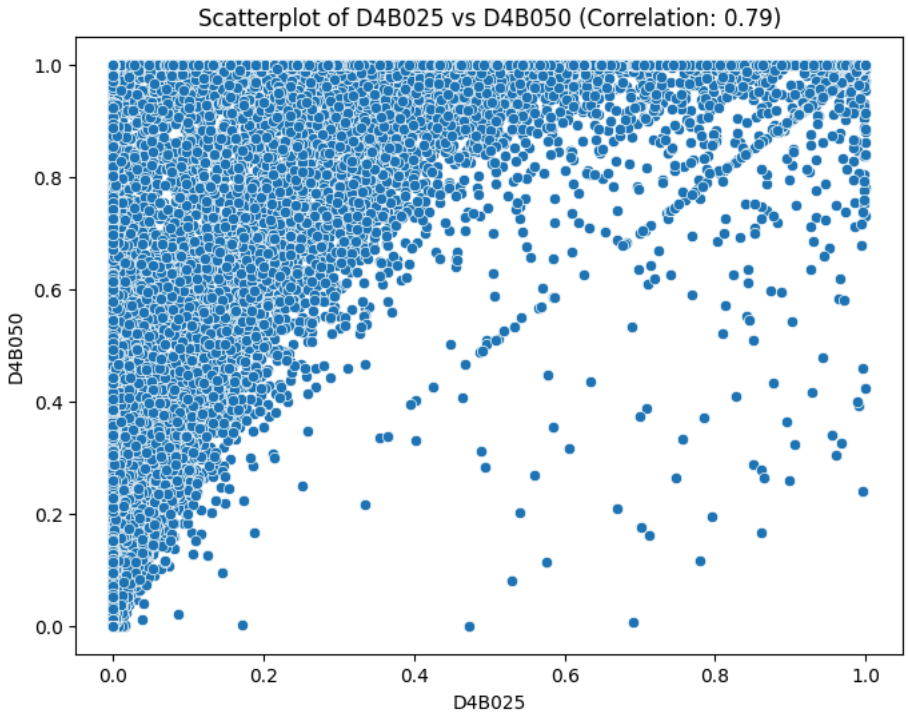

NatWalkInd
12.5
10.0
7.5
5.0
2.5
0
20
40
60
80
100
120
140
D3APO

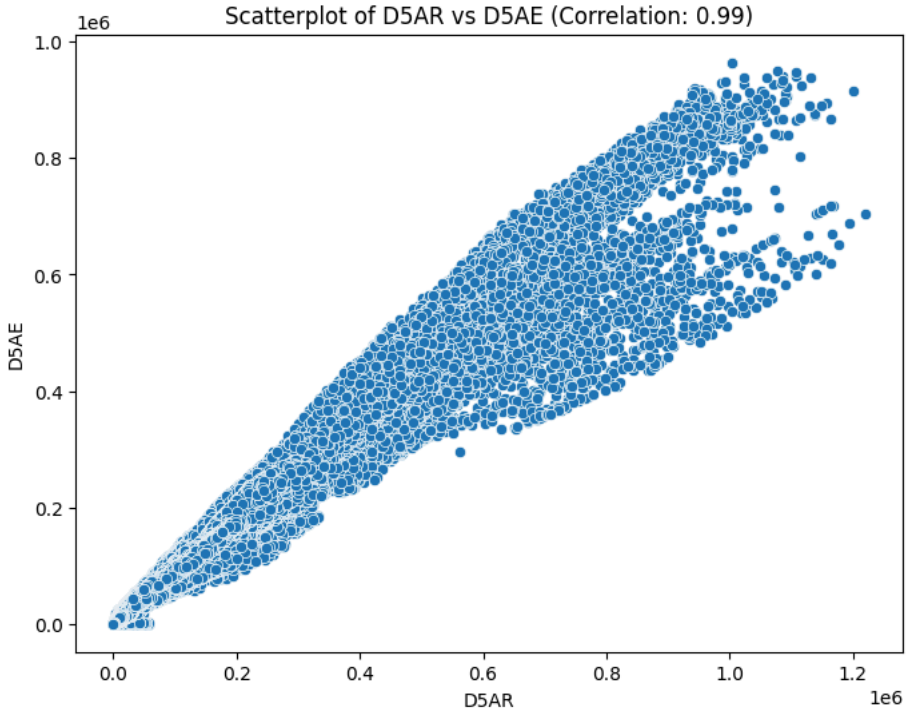

Scatterplot of D3B vs D3BPO3 (Correlation: 0.83)
D3BPO3
4000
3000
2000
1000
0
0
1000
2000
3000
4000
5000
D3B

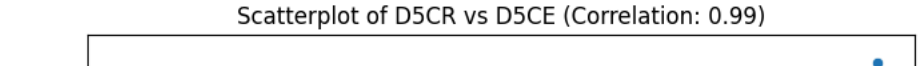

Scatterplot of D3B vs D3BPO4 (Correlation: 0.76)
D3BPO4
2000
1500
1000
500
0
0
1000
2000
3000
4000
5000
D3B

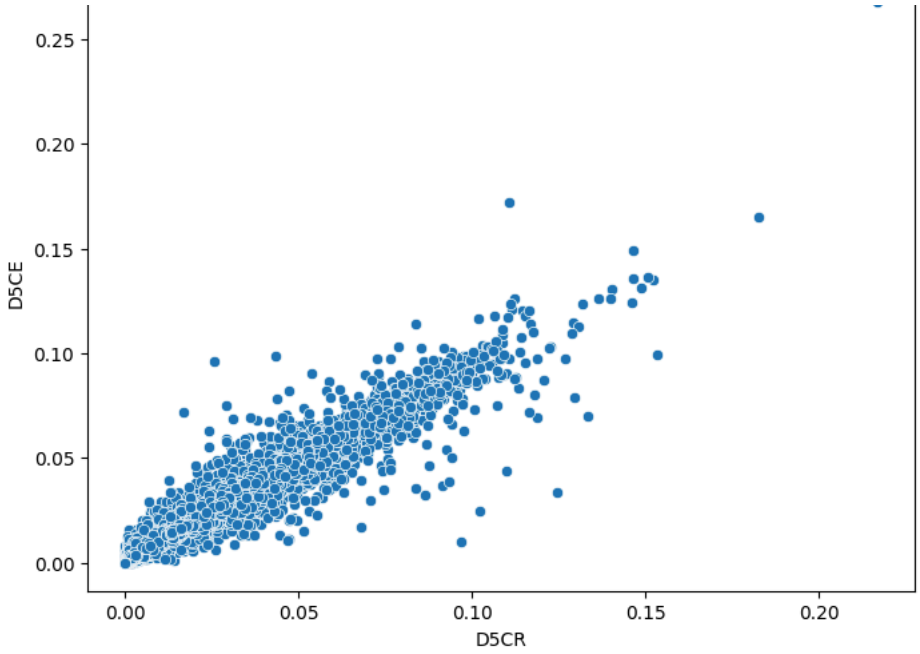

Scatterplot of D3B vs NatWalkInd (Correlation: 0.65)
20.0
17.5

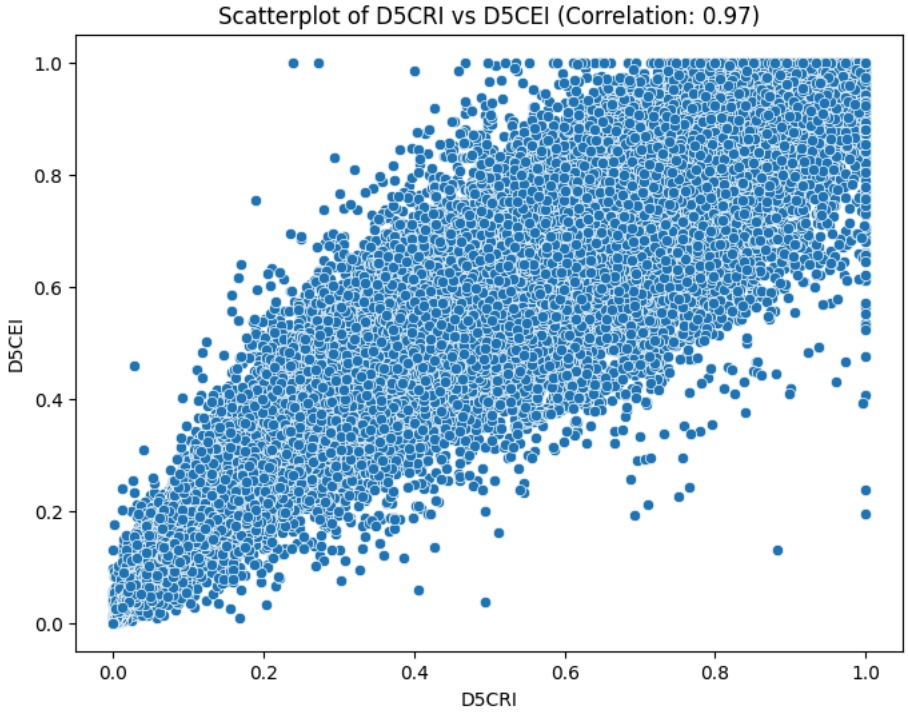

NatWalkInd
15.0
12.5
10.0
7.5
5.0
2.5
0
1000
2000
3000
4000
5000
D3B

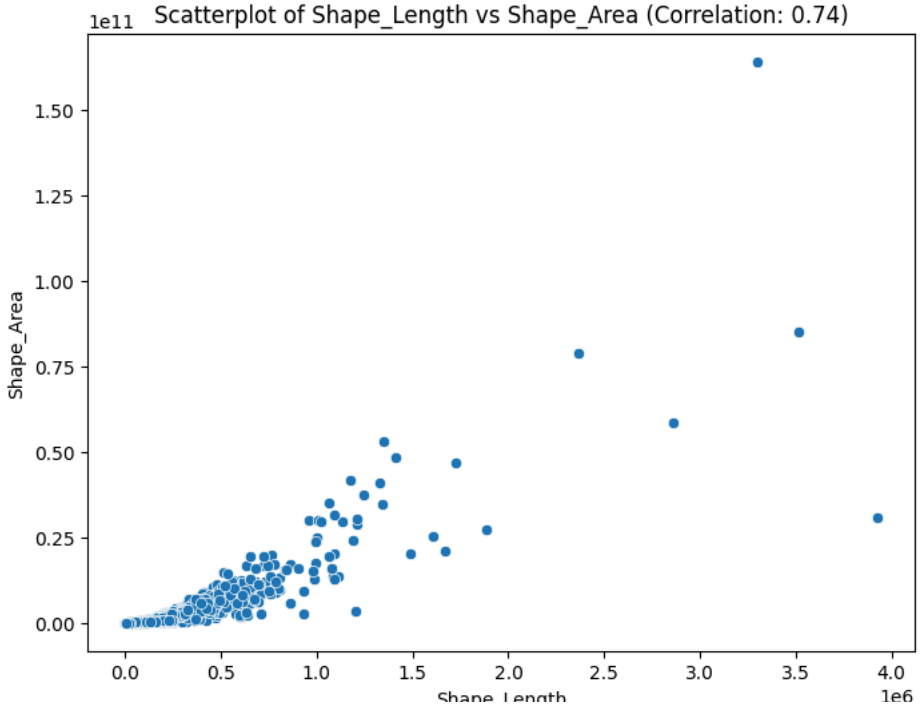

Scatterplot of D4B025 vs D4B050 (Correlation: 0.79)
D4B050
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
D4B025


Scatterplot of D5AR vs D5AE (Correlation: 0.99)
1e6
D5AE
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
D5AR
1e6


Scatterplot of D5CR vs D5CE (Correlation: 0.99)


D5CE
D5CR
0.00
0.05
0.10
0.15
0.20
0.25


Scatterplot of D5CRI vs D5CEI (Correlation: 0.97)
D5CEI
D5CRI
0.0
0.2
0.4
0.6
0.8
1.0

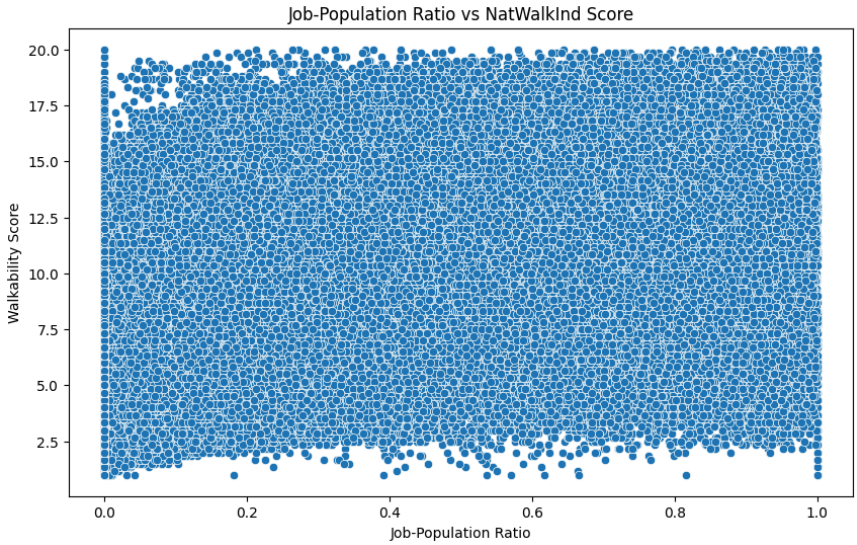

Scatterplot of Shape_Length vs Shape_Area (Correlation: 0.74)
1e11
Shape_Area
Shape_Length
1e6
0.00
0.25
0.50
0.75
1.00
1.25
1.50
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0

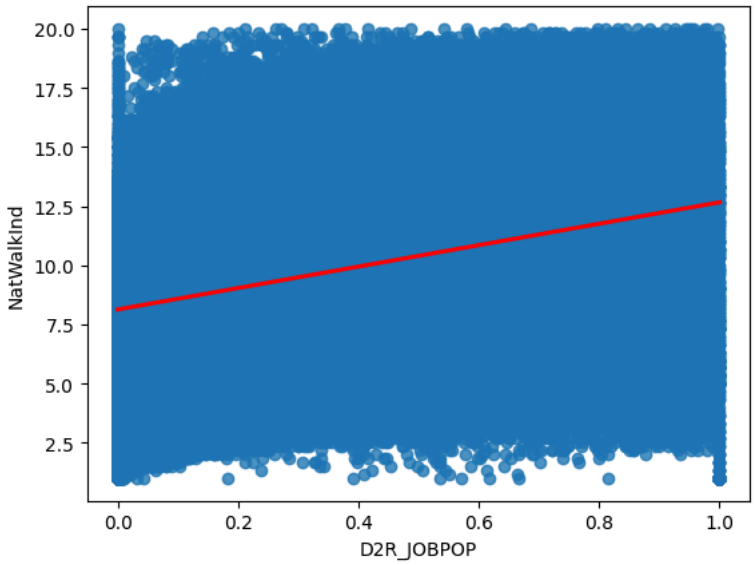
Shape_Length

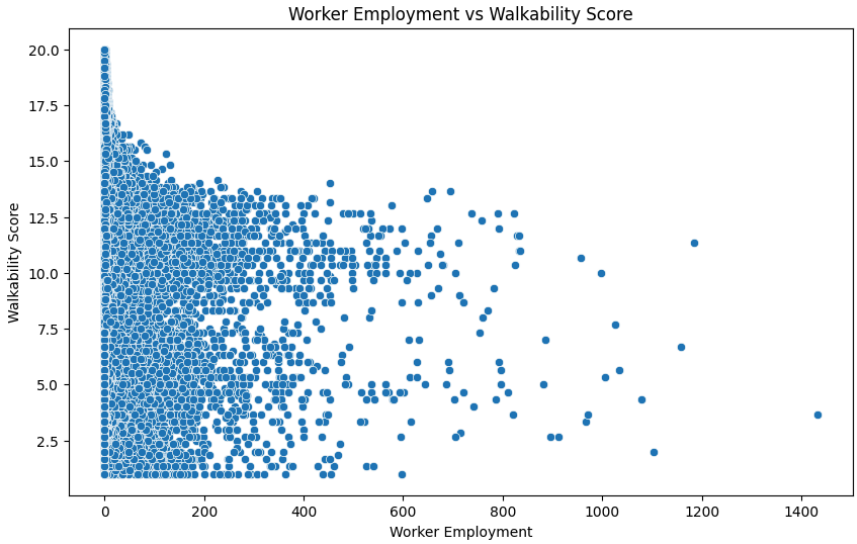
```
high_corr_df = high_corr_pairs.reset_index(drop=True)

# Display the DataFrame
print(high_corr_df.head(134))
```

```
    Variable1   Variable2  Correlation
0     GEOID10     GEOID20     0.999980
1     GEOID10     STATEFP     0.999962
2     GEOID20     STATEFP     0.999983
3         CSA        CBSA     0.638819
4    CBSA_POP    CBSA_EMP     0.998440
..        ...         ...          ...
129       D3B  NatWalkInd     0.653552
130    D4B025      D4B050     0.789571
131      D5AR        D5AE     0.985153
132      D5CR        D5CE     0.988001
133     D5CRI       D5CEI     0.970882

[134 rows x 3 columns]
```

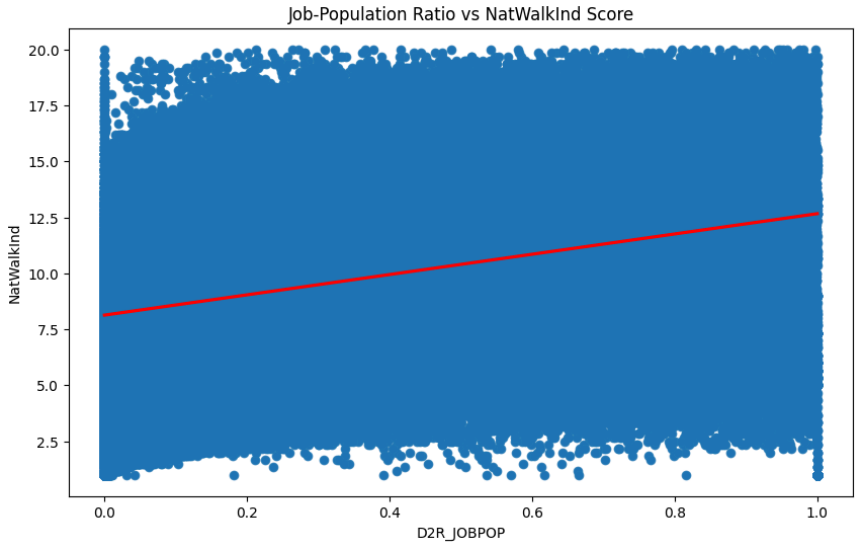
```
# 1. Relationship between jobs-housing balance and neighborhood walkability
plt.figure(figsize=(10, 6))
sns.scatterplot(x='D2R_JOBPOP', y='NatWalkInd', data=data)
plt.title('Job-Population Ratio vs NatWalkInd Score')
plt.xlabel('Job-Population Ratio')
plt.ylabel('Walkability Score')
plt.show()

# Add a regression line
sns.regplot(x='D2R_JOBPOP', y='NatWalkInd', data=data, line_kws={'color': 'red'})

plt.figure(figsize=(10, 6))
sns.scatterplot(x='D2A_WRKEMP', y='NatWalkInd', data=data)
plt.title('Worker Employment vs Walkability Score')
plt.xlabel('Worker Employment')
plt.ylabel('Walkability Score')
plt.show()
```

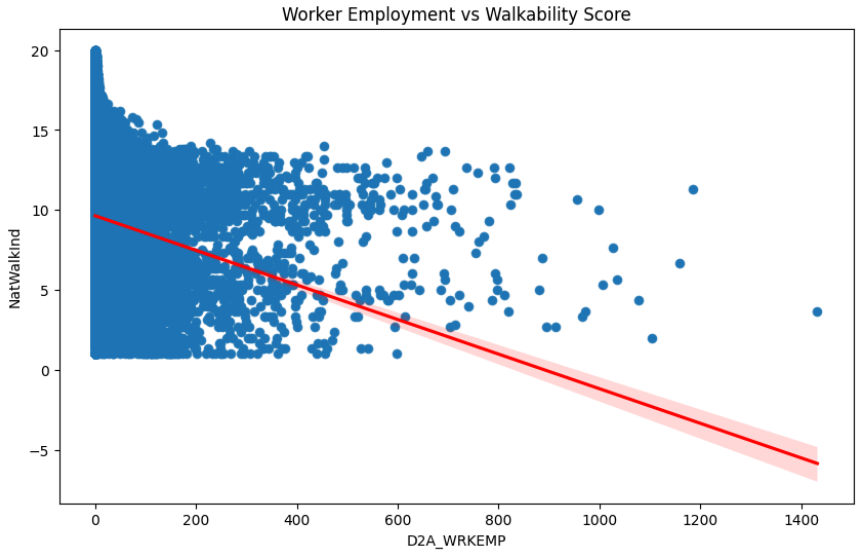

Job-Population Ratio vs NatWalkInd Score
Walkability Score
Job-Population Ratio
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0.2
0.4
0.6
0.8
1.0


NatWalkInd
D2R_JOBPOP
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0.2
0.4
0.6
0.8
1.0

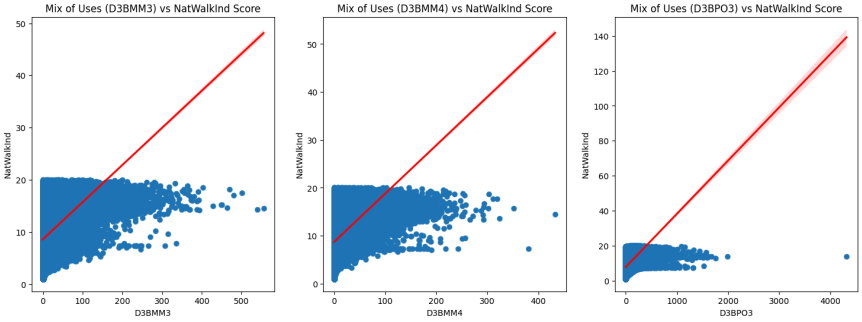

Worker Employment vs Walkability Score
Walkability Score
Worker Employment
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0
200
400
600
800
1000
1200
1400

```
# prompt: make the above plots more interprettable

# 1. Relationship between jobs-housing balance and neighborhood walkability
plt.figure(figsize=(10, 6))
sns.scatterplot(x='D2R_JOBPOP', y='NatWalkInd', data=data)
plt.title('Job-Population Ratio vs NatWalkInd Score')
plt.xlabel('Job-Population Ratio')
plt.ylabel('Walkability Score')

# Add a regression line
sns.regplot(x='D2R_JOBPOP', y='NatWalkInd', data=data, line_kws={'color': 'red'})

plt.show()

plt.figure(figsize=(10, 6))
sns.scatterplot(x='D2A_WRKEMP', y='NatWalkInd', data=data)
plt.title('Worker Employment vs Walkability Score')
plt.xlabel('Worker Employment')
plt.ylabel('Walkability Score')

# Add a regression line
sns.regplot(x='D2A_WRKEMP', y='NatWalkInd', data=data, line_kws={'color': 'red'})

plt.show()
```

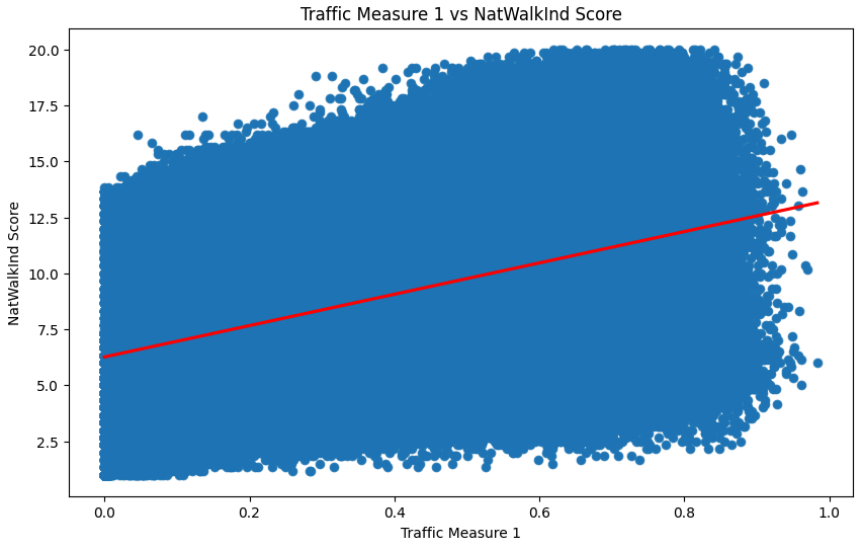


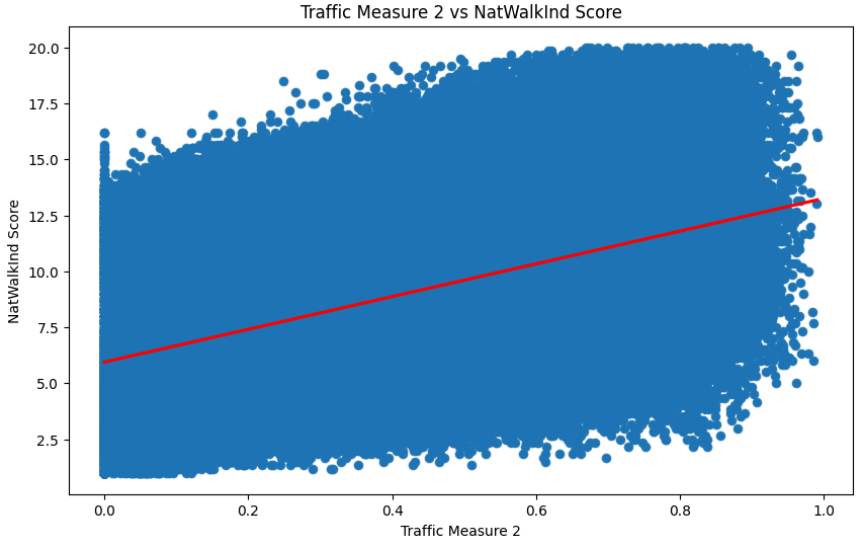



```
# 2. Impact of mix of uses on walkability scores
fig, axes = plt.subplots(1, 3, figsize=(18, 6))
sns.scatterplot(ax=axes[0], x='D3BMM3', y='NatWalkInd', data=data)
sns.scatterplot(ax=axes[1], x='D3BMM4', y='NatWalkInd', data=data)
sns.scatterplot(ax=axes[2], x='D3BPO3', y='NatWalkInd', data=data)

sns.regplot(ax=axes[0], x='D3BMM3', y='NatWalkInd', data=data, line_kws={'color': 'red'})
sns.regplot(ax=axes[1], x='D3BMM4', y='NatWalkInd', data=data, line_kws={'color': 'red'})
sns.regplot(ax=axes[2], x='D3BPO3', y='NatWalkInd', data=data, line_kws={'color': 'red'})

# Setting titles for each subplot
axes[0].set_title('Mix of Uses (D3BMM3) vs NatWalkInd Score')
axes[1].set_title('Mix of Uses (D3BMM4) vs NatWalkInd Score')
axes[2].set_title('Mix of Uses (D3BPO3) vs NatWalkInd Score')

plt.show()
```

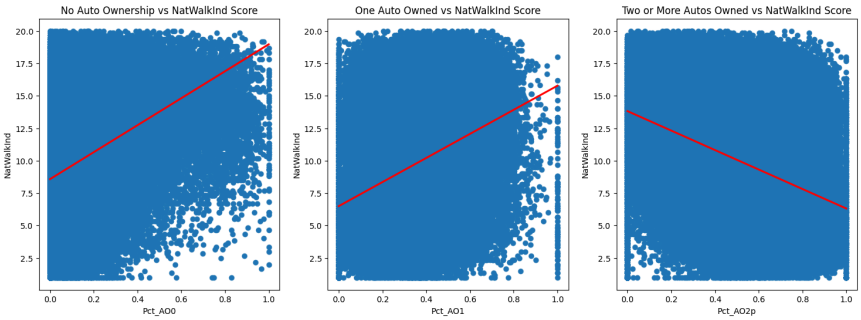



```
# 3. Traffic measures and walkability
plt.figure(figsize=(10, 6))
sns.scatterplot(x='D2C_TRPMX1', y='NatWalkInd', data=data)
sns.regplot(x='D2C_TRPMX1', y='NatWalkInd', data=data, line_kws={'color': 'red'})
plt.title('Traffic Measure 1 vs NatWalkInd Score')
plt.xlabel('Traffic Measure 1')
plt.ylabel('NatWalkInd Score')
plt.show()

plt.figure(figsize=(10, 6))
sns.scatterplot(x='D2C_TRPMX2', y='NatWalkInd', data=data)
sns.regplot(x='D2C_TRPMX2', y='NatWalkInd', data=data, line_kws={'color': 'red'})
plt.title('Traffic Measure 2 vs NatWalkInd Score')
plt.xlabel('Traffic Measure 2')
plt.ylabel('NatWalkInd Score')
plt.show()
```






```
# 4. Connection between auto ownership rates and walkability
fig, axes = plt.subplots(1, 3, figsize=(18, 6))
sns.scatterplot(ax=axes[0], x='Pct_AO0', y='NatWalkInd', data=data)
sns.scatterplot(ax=axes[1], x='Pct_AO1', y='NatWalkInd', data=data)
sns.scatterplot(ax=axes[2], x='Pct_AO2p', y='NatWalkInd', data=data)


sns.regplot(ax=axes[0], x='Pct_AO0', y='NatWalkInd', data=data, line_kws={'color': 'red'})
sns.regplot(ax=axes[1], x='Pct_AO1', y='NatWalkInd', data=data, line_kws={'color': 'red'})
sns.regplot(ax=axes[2], x='Pct_AO2p', y='NatWalkInd', data=data, line_kws={'color': 'red'})

axes[0].set_title('No Auto Ownership vs NatWalkInd Score')
axes[1].set_title('One Auto Owned vs NatWalkInd Score')
axes[2].set_title('Two or More Autos Owned vs NatWalkInd Score')
plt.show()
```




```
# Filter dataset for numerical columns
numerical_data = data.select_dtypes(include=[np.number]).drop('NatWalkInd', axis=1)
```

```
# Standardize the imputed data
scaler = StandardScaler()
scaled_data_imputed = scaler.fit_transform(numerical_data)

# Apply PCA
pca = PCA(n_components=25)
pca_result_imputed = pca.fit_transform(scaled_data_imputed)

print(pca_result_imputed.shape)
```

```
(220740, 25)
```

```
# Assuming pca_result_imputed contains the PCA-transformed data and 'NatWalkInd' is the target variable

# Splitting the PCA results into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(pca_result_imputed, data['NatWalkInd'], test_size=0.2, random_state=42)

# Initializing and fitting the linear regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Making predictions on the testing set
y_pred = model.predict(X_test)

# Calculating R^2 and MSE for evaluation
r2 = r2_score(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)

print(f"R^2: {r2}, MSE: {mse}")
```

```
R^2: 0.9317995207261803, MSE: 1.3082710503921329
```

```
def adjusted_r_squared(r_squared, n, k):
    return 1 - ((1 - r_squared) * (n - 1) / (n - k - 1))
```

```
import pandas as pd
import numpy as np
```

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, LeakyReLU, PReLU, ReLU, ELU, Lambda
from tensorflow.keras.activations import selu
from tensorflow.keras.optimizers import Adam
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
from tensorflow.keras.losses import MeanSquaredError
from tensorflow.keras.layers import Layer

# Definition of the Maxout custom layer
class Maxout(Layer):
    def __init__(self, units, pieces):
        super(Maxout, self).__init__()
        self.dense_layers = [Dense(units) for _ in range(pieces)]

    def call(self, inputs):
        outputs = [dense(inputs) for dense in self.dense_layers]
        return tf.reduce_max(outputs, axis=0)

# Define activation functions, ensuring each is correctly instantiated or wrapped
activation_functions = {
    'ReLU': ReLU(),
    'LeakyReLU': LeakyReLU(alpha=0.01),
    'PReLU': PReLU(alpha_initializer='zeros', alpha_regularizer=None, alpha_constraint=None),
    'RReLU': ReLU(max_value=10),
    'SReLU': LeakyReLU(alpha=0.5),
    'ELU': ELU(alpha=1.0),
    'PeLU': Lambda(lambda x: tf.nn.relu(x) - tf.nn.relu(-x)),
    'Selu': Lambda(lambda x: selu(x)),  # Wrapping as Lambda for consistency
    'Maxout': Maxout(units=2, pieces=2),
    'ELiSH': Lambda(lambda x: tf.nn.elu(x) * tf.nn.sigmoid(x)),
    'HardELiSH': Lambda(lambda x: tf.nn.relu(x) * tf.nn.sigmoid(x))
}

# Initialize the model
model = Sequential()
model.add(Dense(25, input_dim=25))  # Starting fresh with the input layer

# Loop through activation functions to compile, train, and evaluate models
models = {}
mse_scores = {}
mae_scores = {}
r2_scores = {}
adj_r_squared_scores = {}


for name, activation in activation_functions.items():
    model.add(Dense(3, activation=activation))
    model.add(Dense(1, activation='linear'))
    model.compile(optimizer=Adam(learning_rate=0.001), loss=MeanSquaredError())
    models[name] = model

    # Assuming X_train, y_train, X_test, y_test are defined elsewhere and accessible
    history = model.fit(X_train, y_train, epochs=100, batch_size=10, verbose=0, validation_split=0.2)
    predictions = model.predict(X_test)

    mse = mean_squared_error(y_test, predictions)
    mse_scores[name] = mse
    mae = mean_absolute_error(y_test, predictions)
    mae_scores[name] = mae
    r_squared = r2_score(y_test, predictions)
    r2_scores[name] = r_squared
    n = X_test.shape[0]
    k = X_test.shape[1]
    adj_r_squared = 1 - (1-r_squared)*(n-1)/(n-k-1)  # Adjusted R-Squared formula
    adj_r_squared_scores[name] = adj_r_squared

    # Reset the model for next activation function
    model = Sequential()
    model.add(Dense(25, input_dim=25))

# Create a DataFrame with the results
df = pd.DataFrame({
    'Activation Function': list(activation_functions.keys()),
    'MSE': list(mse_scores.values()),
    'MAE': list(mae_scores.values()),
    'R2': list(r2_scores.values()),
    'Adjusted R2': list(adj_r_squared_scores.values())
})

# Display the DataFrame
print(df.to_string())
```