Michael A Cwikielnik

New York, NY | 774-287-8786 | michael.cwikielnik@snhu.edu | https://github.com/mikecwikielnik |
https://www.hackerrank.com/mikecwikielnik

Education

Southern New Hampshire University

Master of Science in Data Analytics. GPA 3.923
Relevant Coursework: Advanced Data Analytics, Predictive Analytics, Decision Methods, and Modeling, Present & Visualize Data, Quantitative Analysis

New York University

Credit Risk Certificate
Relevant Coursework: Financial Statement Analysis, Corporate Finance, Credit Analysis, Financial Modeling

University of Massachusetts

Bachelor of Science in Biology
Relevant Coursework: Probability, and Statistics I, Applied ODE's, Linear Algebra, Microbiology, Calculus III, Population Biology

Projects

- Developed multiple Machine Learning models: neural network, linear regression, decision tree, random forest using Python. The packages that were utilized were pandas, keras, sklearn, and matplotlib. The dataset was the Boston Housing data set from Kaggle. The target variable was 'MEDV' which we renamed to 'PRICE'. From sklearn, the function train_test_split was used to generate a test set of 0.2. The feature variables were scaled to provide uniformity.
    - Neural Network: the NN model was defined with the Sequential function. The model had two hidden layers (128 and 64 nodes respectively). In both hidden layers, the activation function was ReLu. The output layer has one node. In the compile stage, the optimizer was adam. The loss (mean-squared-error) and accuracy (mean-absolute-error) was the metrics calculated. The model was fit with a validation split of 0.2 and 100 epochs.
        - Loss: 11.9250
        - Accuracy: 2.2668
        - Generated a visualization that mapped training, validation loss (mse) and accuracy (mae) over time (epochs). Both visualizations communicated that between 10 – 20 epochs is where both metrics leveled off.
    - Linear Regression: the model utilized from sklearn linear_model, mean_squared_error, mean_abosolute_error.
        - Loss: 24.2911
        - Accuracy: 3.1891
    - Decision Tree: the model utilized the Decision Tree Regressor from sklearn.
        - Loss: 10.0325
        - Accuracy: 2.2990
    - Random Forest: the model utilized the Random Forest Regressor from sklearn. The model has 30 trees.
        - Loss: 7.4281
        - Accuracy: 2.0110
        - In a Random Forest model, we are allowed to dive deep into the feature variables with the model.feature_importances_ attribute. The purpose of this Series is to yield what features were most important in creating the results. In this case, we had two variables 'RM' and 'LSTAT' have 0.4988 and 0.3073 respectively. The calculation of the feature variables are out of 1.0, that leaves approximately 20% for the remaining 11 variables.
            - RM: the average number of rooms per dwelling
            - LSTAT: percentage of lower status.

- Developed several insights into mass shootings in America. The data set covers shootings from September 2018 – May 2022. What was analyzed was shootings with at least 1 killed and how many. Implemented SQL window functions to facilitate comparisons within the data set. Broadened the insights gained by creating bar charts in Python. The analysis concluded that in the *period of September 2018 – May 2022* that the states with the most shootings where at least person died is *Texas, California, and Illinois.*

- Built a game in Python called Alien Invaders. The game is like space invaders from the 1980s. This was an experience that further developed my skills in Python. The project refactored code multiple times to optimize the function of classes and increase readability.
    - Compiled several classes and several files to produce a computer game.
    - Alien Invaders can be used without a mouse, keeps track of scoring, and a variety of settings that affect playability.

Experience

New York Office of Cannabis Management; New York, NY March 2023 - present
Data Analyst

- CDI Plus was a project that dived deeper into CDI characteristics. Utilized PostgreSQL, Power BI.
    - Built many tables, queries to understand the variables in various census datasets. Coded window functions, ranking systems, joins which yielded scope into variables like median household income, median home values, census tract arrests/rates, population, and vacancy rates. Samples were either NYC, ROS, or the population- NYS.
    - This is over the 2016 – 2020 timeframe and at the census tract or county level.
        - We found that Queens/Staten Island has the least count of arrests (249/141 census tract max ea.) where Manhattan stands out as number 1 (337 census tract max).
            - Manhattan has the census tracts with the largest population.
        - We uncovered that it doesn't matter where you live (CDI or non-CDI census tract) in the Bronx, your home is likely around $450k.
        - The largest discrepancy in the average median household income exists in Manhattan at 132.53%. While Queens has the smallest discrepancy at 6.41% The classification is between CDI and non-CDI census tracts.
- Mine New York City marijuana arrest data from 1980 - 2020. The goal is to plot the data on a map.
    - The boroughs cleaned: Brooklyn: 99%, Bronx: 98%, Manhattan: 89%, Queens: 90%, Staten Island: 99%
        - Outside NYC: 85%
    - The datasets combined were over 1.1 million with a match rate of 93% across NYS.
- Calculated cannabis arrests in total, percentile rank, and class rank for US Census NYC/NYS geoids in 6-year, 10-year time periods from 1980 – 2021.
    - As policing policies changed over time in NYC/NYS governments, we saw areas that were open-air drug markets in the 1980s turned into gentrified neighborhoods in the 2010s.
        - Census tracts that were leading in arrests in the 1980s were not as high as other census tracts in other areas during the later administrations. Example: LES in the 80s to Washington Heights in the late 00s.
        - PostgreSQL to calculate the total, percentile rank, and dense rank of the geoids.
        - Created my own Geocode Data Dictionary for the years/class's definitions.
- Built a NYS/NJ/CT (Tri-State area) analysis that utilized PostgreSQL- which calculated cannabis arrests, percentile rank, and dense rank.
    - Generated a 6-year, 10-year analysis comparison for each state in the Tri-State area.
    - Used a percentile rank of > 0.98.
        - Concluded that NJ/CT cannabis arrests can be throw out of our NYC/NYS analysis.
        - NYS cannabis arrests continuously outpaced NJ or CT in any given time-period between 1980 – 2021.
- Calculated the top 25%, top 15% census tracts in cannabis arrests in decade time-periods between NYC and ROS (outside of NYC).
    - Used R to create some appealing visualizations that communicated that NYC generally outpaced it's non-NYC counterpart in the 75^{th}, and 85^{th} percentiles.
    - PostgreSQL drew the data that went into R.
- Write R/PostgreSQL code to form tables that document discrete arrest data and calculated number of arrests per description.
- Create PostgreSQL queries to dive into attendance data for new licensees.

Triple Crown Ale House; New York, NY Nov 2022 – March 2023
Waiter

Miramar; New York, NY Feb 2021 – Oct 2022
Waiter

Core Competencies

APIs ● Black Scholes Merton Model ● Binomial Tree ● Bonds ● Central Clearing ● Country Risk ● Data Visualization (dashboards, categorical data, time series data) ● Data Mining ● DB Browser ● Decision Trees ● Derivatives ● Descriptive Analytics ● Excel ● Exchange and OTC Markets ● Exotic Options ● Fundamentals of Probability ● Fund Management ● Futures Market ● Futures for Hedging ● JavaScript ● Jupyter notebooks ● Linear Programming ● Logic and data analysis ● Math (calculus I-III, ODE, linear algebra, statistics, probability) ● Machine Learning ● Microsoft SQL Server ● Microsoft Visual Studio Code ● MySQL ● Neural Networks ● NumPy ● Operations Research ● Operational Risk ● Option Sensitivity Measures: Greeks ● Option Market ● Oracle ● pandas ● Pattern and Trend Identification ● Predictive Analytics ● Prescriptive Analytics ● Pricing Futures and Forwards ● Problem-solving and Troubleshooting ● Properties of Options ● PostgreSQL ● Power BI ● Python ● R ● Rattle ● Random Forest ● Random Variable ● Regression Analysis ● Relational Databases ● Risk Management ● Sci-Kit Learn ● Stocks ● SQL ● SQLite ● Tableau ● TensorFlow (Keras) ● Times Series ● Trading Strategies