**Original Case Study Final Project**

For the case study, I chose to analyze data from a loan data set to see who would be approved for a home loan. Many loan applications these days are automated. Knowing which variables (such as income, gender, education, credit history, etc.) contribute the most to the loan approval process will allow applicants to focus on the key variables affecting their loan approval.

**Step-by-Step Instructions:**

**Case Study: Analyze data to predict who will be approved for Loan**

1. Loaded the data from the csv file into a Dataframe.
2. Displayed the dimensions of the file to get a good idea the amount of data I was working with.
3. Displayed the first 5 rows of data to see the column headings and the type of data for each column.
   1. Notice that Credit\_History is represented as a 1 or 0
   2. Notice that missing data is represented as “NaN”
   3. The Loan-Status variable will be the “target” and the other variables will be the “features.”
4. Some questions about being approved for a loan:
   1. What do the variables look like? For example, are they numerical or categorical data? If they are numerical, what are their distribution; if they are categorical, how many are there in different categories?
   2. Are the numerical variables correlated?
   3. Are the distributions of numerical variables the same or different among survived and not survived? Is the approval rate different for different values? For example, were people more likely to be approved if they are married?
   4. Are there different approval rates in different categories? For example, did more men get approved then women?
5. Looked at the summary information about my data (total, mean, min, max, freq, unique, etc.)
6. Made some histograms of my numeric data.
   1. Applicant income was about $5.4k/mo. with the tail extending to the right.
      1. Many of the co-applicant’s income was zero, with the average being around $1.6k/mo.
   2. Loan amounts were mostly equally distributed, with the tail going to the right.
      1. Nearly all loan amount terms were 360 months, with some minor variation.
7. Made some bar charts for my categorical variables.
8. Made some Pearson Ranking charts to see if my data was correlated.
   * 1. The correlation between the variables is low. The results show there is some negative correlation between applicant and co-applicant income, but not much.
9. Used Parallel Coordinates visualization to compare the distributions of numerical variables between passengers that survived and those that did not survive.
   1. Applicants/Co-applicants with higher income received more loan approvals (usually based on only one of the salaries).
10. Used Stack Bar Charts to compare applicants who were approved to applicants who were not approved based on the other variables.
11. Reduced features and filled in missing values. Reduced features last, as I found it necessary to fill in the missing values prior to running tests (i.e. component analysis and thresholding binary feature selection) which were needed to determine which features to drop.
    1. Reduced features
       1. I applied component analysis to the numeric features to retain 99% of confidence but did not reduce number of numeric features. Ran again with 95% confidence, but still did not reduce numeric features. Concluded that the 4 numeric features must be important, so left as is.
       2. Determined categorical features to drop using thresholding binary feature selection after converting the categorical features to numbers. Array provided left 6 remaining categorical features. Filled in missing values for the following fields: Loan Amount (used median), Loan Amount Term (used most common: '360'), Gender (used most common: 'male'), Married (used most common: 'no'), Dependents (used most common: '0'), Self-employed (used most common: 'no'), and Credit History (used 'no' to be conservative, even though most common was 'yes').
12. Applicant and co-applicant income had a negative correlation with each other as the higher one applicant’s income the less the others tended to be. Also, the loan approval is based on both of these incomes combined (as opposed to just one). Log transformation did not seem appropriate for this case, so applied a custom transformation to combine columns for applicant and co-applicant incomes into total income. Ran histogram for total income, which was less skewed.
13. Converted categorical data to numbers (Gender, Married, Dependents, Education, Self-Employed, Property Area). Yes, represented as ‘1,’ no as ‘0.’
14. Created a whole features dataset to be used for the training and validation data splitting. Combined the 4 numerical features and the dummie features together.
15. Used logistic regression model since it’s good for classification problems.
    1. Metrics for the evaluation:
       1. Created confusion matrix for Approved/Not Approved. About 64% accuracy, so not as accurate as the Titanic Case Study.
       2. Ran classification report to get precision, recall, and F1 score. Only got precision of 65%, so was not sure about this one. Recall provided about 95% so this reassured me. F1 score, which is a combination of precision and recall, of 77%
       3. Ran ROC curve and all results were above the dotted line, so better results than if randomly guessed.

**Additional Steps Added:**

* For additional EDA, wanted to look at the non-numeric variables, so used describe, setting exclude to 'number' and added .info to determine the data types
* Additional Observations:
  + Approved - about twice as many applications wered approved vs. not-approved
  + Gender - About 5 times as many applied than women
  + Married - About twice as many applicants were married vs non-married
  + Dependents - most applicants had zero dependents; about 3 times as likely to be approved
  + Education - About 4 times as many applicants were graduates
  + Self-Employed - MOre than 5 times as many applicants were not self-employed
  + Credit History - About 5 times as many applicants had a credit history report
  + Property Area - roughly evenly distributed with slightly more for semiurban
* Added Seaborn Heatmap for revised correlations.
* Calculated AUC score of 0.73, so model appears to have improved.