**Week 3**

**Summary**

**Milestones achieved**

* Ran classifier model data set, and recorded the major metrics like: accuracy, TP rate, FP rate, precision, recall, F measure, the ROC area etc**.**
* Compared between different evaluation metrics.
* Performed data preprocessing, feature engineering, and a chi-square test for identifying significant feature correlations with the target variable (Satisfaction).

**Conclusions**

* Concluded which evaluation techniques worked best. **Random Forest** is the best option for both data sets due to its consistently high performance across all metrics. Both **Decision Tree** and **KNN** are good alternatives, depending on our preference for speed, complexity, or interpretability. **Logistic Regression** may be a simpler model to interpret but falls behind the other models in performance.
* We performed 5-fold cross-validation on our KNN model. This indicates that our model performs well and consistently across the different folds, with minimal variation in accuracy between the splits. High accuracy and low standard deviation suggest good generalization and stability of the model.
* Concluded evaluation metrics, If our goal is to balance false positives and false negatives (e.g., ensuring correct satisfaction predictions while minimizing errors), **F1 Score** is the most suitable metric. If false positives or false negatives are more important (e.g., predicting satisfied customers more accurately), choose **precision** or **recall** based on your priority. For an overall sense of how well the model distinguishes between satisfied and dissatisfied customers, **ROC AUC** is a solid metric to use.
* Concluded that **Significant combinations** are those where the p-value is less than a predefined threshold (0.05), indicating a statistically significant relationship between the feature and the target variable.

**References**

* <https://medium.com/@outside2SDs/an-overview-of-correlation-measures-between-categorical-and-continuous-variables-4c7f85610365>
* <https://colab.research.google.com/drive/1mqfvb_NTjm1g_OPYrnck_EreWclqHzhx?usp=sharing>
* <https://www.tutorialspoint.com/machine_learning_with_python/index.htm>
* Müller, AC, & Guido, S 2016, Introduction to Machine Learning with Python : A Guide for Data Scientists, O'Reilly Media, Incorporated, Sebastopol. Available from: ProQuest Ebook Central. [29 September 2024].
* <https://ebookcentral.proquest.com/lib/hw/reader.action?docID=4698164&ppg=28>
* <https://github.com/ageron/handson-ml2/blob/master/03_classification.ipynb>

**Next Steps**

* Preparation for next steps of the project.
* Setting up the project pitch with hypothesis and question of the project.
* Organize the Pipeline; Modularize the code: Each step (downloading, loading, preprocessing, visualization, etc.) should be a function, making it easier to maintain and extend.
* Ensure the pipeline runs end-to-end: From downloading the dataset (if necessary) to splitting it into train and test sets.