How-to-Use

This document serves as a guide to successfully navigate any* (with exceptions) DS/ML problem.
If there is an asterisk in front of an item, then it may not be applicable to all situations.
Made by Robin P.M. Kras, 2023

1. Problem Understanding

- **Goal Definition:** Understand the problem clearly. What are you trying to predict or classify?
- ***Stakeholder Communication:** Discuss with stakeholders (if any) to clarify objectives, assumptions, and expectations.
- ***Constraints and Requirements:** Determine any limitations (e.g., time, computational resources) and requirements (e.g., model interpretability, real-time inference).

2. Data Collection

- ***Gather Data:** Collect the relevant data from multiple sources (e.g., databases, APIs, public datasets).
- **Verify Data Availability:** Check for missing data or outliers in your data sources.

3. Data Exploration

- **Initial Data Assessment:** Perform basic data exploration (e.g., summary statistics, histograms, box plots) to get a sense of the dataset.
- **Data Types:** Check and ensure correct data types (numerical, categorical, dates, etc.).
- **Identify Issues:** Identify missing data, outliers, and anomalies.

4. Data Preprocessing

- **Handle Missing Data:** Decide how to deal with missing values (e.g., imputation, removal).
- **Outlier Detection:** Identify and handle outliers (e.g., removal or capping).
- **Feature Engineering:** Create new features that could help the model.
- **Data Transformation:** Normalize/scale data as needed (e.g., min-max scaling, standardization).
- **Categorical Encoding:** Convert categorical features into numerical values (e.g., one-hot encoding, label encoding).
- **Feature Selection:** Remove irrelevant or redundant features to improve model performance and interpretability.

5. Exploratory Data Analysis (EDA)

- **Visualizations:** Create visualizations to identify trends, relationships, and distributions (e.g., scatter plots, correlation matrices).
- **Statistical Tests:** Perform any relevant statistical tests (e.g., t-test, chi-square) to understand relationships between variables.
- **Check Data Balance:** For classification tasks, check if there's a class imbalance.

6. Model Selection

- **Choose Algorithms:** Based on problem type (regression, classification, clustering, etc.), choose suitable algorithms (e.g., linear regression, decision trees, random forests, neural networks).
- **Baseline Model:** Start with a simple model to establish a baseline (e.g., logistic regression for classification, linear regression for regression).
- **Hyperparameter Tuning:** Tune model hyperparameters using techniques like Grid Search or Random Search.

7. Model Training

- **Split Data:** Split data into training, validation, and test sets (typically 70% training, 15% validation, 15% testing).
- **Train Models:** Train your chosen algorithms on the training data.
- **Cross-Validation:** If needed, use cross-validation to assess model performance.

8. Model Evaluation

- **Assess Performance:** Evaluate your model using appropriate metrics (e.g., accuracy, precision, recall, F1 score, AUC for classification; RMSE, MAE for regression).
- **Compare Models:** If multiple models are used, compare their performance on validation data.
- **Check Overfitting:** Look for overfitting by comparing training and validation performance.

9. Model Refinement

- **Hyperparameter Optimization:** Use more advanced techniques like Bayesian optimization, or ensemble methods to refine the model.
- **Feature Engineering Revisit:** Sometimes iterating on the feature engineering can improve performance.
- **Ensemble Methods:** Consider ensemble models (e.g., bagging, boosting, stacking) to improve accuracy.

10. Final Model Evaluation

- **Test Set Evaluation:** Once the final model is selected, test it on the unseen test data to assess its generalization performance.
- **Error Analysis:** Investigate the cases where the model performs poorly (e.g., misclassified data points).

11. Model Deployment

- **Deployment Strategy:** Decide how you want to deploy the model (e.g., web application, batch processing).
- ***Model Monitoring:** Once deployed, monitor model performance to ensure it continues to perform well over time.
- ***Model Retraining:** Set up a process for retraining the model periodically as new data becomes available.

12. Documentation and Reporting

- **Document Process:** Ensure to document your analysis, findings, assumptions, and decisions made throughout the process.
- **Communicate Results:** Prepare visualizations and reports to communicate results to stakeholders effectively.

13. Crucial: Iterate and Improve

- **Continuous Improvement:** Based on feedback or performance in production, iterate on the model with new data, improvements in features, or algorithmic changes.