### **Project Title: Automated Model Selection and Explanation Through Counterfactual Analysis and Visualization**

#### **Author Information**

**Name:** Adir Elmakais  
**ID:** 316413640

#### **1. Project's Goal**

##### **1.1 What element in the DS pipeline are you trying to improve?**

This project focuses on improving model selection, model evaluation, and explainability in the data science pipeline. It specifically aims to enhance these elements by leveraging counterfactual explanations to provide a deeper understanding of model behavior and facilitate more informed model selection.

##### **1.2 Why the chosen element needs improvement?**

**Model Selection:** The current practice of model selection often relies heavily on trial-and-error and overall performance metrics (e.g., accuracy, F1-score). This approach can be time-consuming and may not reveal why a particular model is superior or inferior for specific instances or subgroups. A more nuanced understanding of model behavior is needed to guide selection effectively.

**Explainability:** Understanding how a model arrives at its predictions is crucial for building trust, debugging errors, identifying potential biases, and ensuring responsible use of AI. Traditional explainability methods might not fully capture the differences in decision-making processes between models.

**Counterfactuals:** Counterfactual explanations offer a powerful way to understand model behavior by showing how predictions would change if input features were altered. However, generating, analyzing, and comparing counterfactuals for multiple models is often complex and not integrated into standard model selection workflows.

##### **1.3 What are the desired results?**

* Develop an automated pipeline that streamlines the process of training multiple models, generating counterfactual explanations for each, and visualizing these explanations in a comparative manner.
* Provide data-driven insights into the strengths and weaknesses of different models based on their counterfactual explanations, going beyond standard performance metrics.
* Demonstrate how counterfactual analysis can aid in model selection by revealing model behavior that might not be apparent from overall performance metrics alone.
* Offer a user-friendly visualization tool that facilitates the comparison of counterfactuals across different models, making the analysis accessible to a broader audience.

##### **1.4 What is the relation/connection to the material we learn in class?**

This project directly relates to several key concepts covered in the Tabular Data Science course:

* **Model Selection:** We will apply and extend principles of model selection by incorporating counterfactual analysis as a selection criterion.
* **Model Evaluation:** We will utilize and potentially expand upon the evaluation metrics discussed in class (e.g., accuracy, F1-score, AUC-ROC) to assess model performance.
* **Explainability:** We will explore counterfactual explanations as a method for enhancing model interpretability, building on any discussions about explainable AI in the course.
* **Visualization:** The project will involve designing effective visualizations to communicate complex information, drawing on data visualization principles.

#### **2. Your Solution**

##### **2.1 Initial Ideas:**

**Automated Model Training:**

* Develop a Python-based system using libraries like scikit-learn, xgboost, and lightgbm to automatically train a diverse set of classification or regression models on a given dataset. The models will include Logistic Regression, Decision Trees, Random Forest, Support Vector Machines (SVM), and XGBoost.
* Incorporate hyperparameter optimization using GridSearchCV or RandomizedSearchCV to ensure each model is reasonably tuned.

**Counterfactual Generation:**

* Employ the DiCE (Diverse Counterfactual Explanations) library or implement similar algorithms to generate counterfactual explanations for a selected set of instances from the test set. These instances can include:  
  + Misclassified instances.
  + Instances with low prediction confidence.
  + A diverse and representative sample of correctly classified instances.
* Focus on generating counterfactuals that are:  
  + **Valid:** They lead to the desired change in the model's prediction.
  + **Proximate:** They involve minimal changes to the original feature values.
  + **Sparse:** They alter a small number of features.
  + **Diverse:** If multiple counterfactuals are generated for an instance, they should explore different ways to change the prediction.

**Counterfactual Visualization and Comparison:**

* Design and implement visualizations to compare counterfactual explanations across the different models. Potential visualization techniques include:  
  + **Parallel Coordinates Plots:** To show how feature values change across different counterfactuals for each model, highlighting which features are most frequently altered and the magnitude of those changes.
  + **Tabular Summaries:** To present counterfactuals in a tabular format, making it easy to compare feature changes across models side-by-side.
  + **Summary Statistics:** Calculate and display summary statistics of counterfactual changes (e.g., average number of features changed, average magnitude of change) for each model to provide a quantitative comparison.
* Emphasize differences in how models generate counterfactuals to gain insights into their decision-making processes.

**Model Selection Guidance:**

* Develop a set of heuristics or guidelines to aid in model selection based on the counterfactual analysis. For example:  
  + If a model consistently requires large or unrealistic feature changes to generate counterfactuals, it might indicate overfitting or a lack of robustness.
  + If a model frequently alters features that are known to be irrelevant or sensitive (based on domain knowledge), it might suggest biases or reliance on spurious correlations.
* Provide a summary report that highlights the strengths and weaknesses of each model based on both overall performance metrics and counterfactual analysis.

##### **2.2 Ways to Measure Your Solution:**

* **Model Performance:** Evaluate each model using standard metrics appropriate for the task (e.g., accuracy, F1-score, AUC-ROC for classification; R-squared, RMSE for regression).
* **Counterfactual Quality Metrics:**
  + **Validity:** Calculate the percentage of generated counterfactuals that successfully lead to the desired change in prediction.
  + **Proximity:** Measure the average L1 or L2 distance between the original instances and their corresponding counterfactuals.
  + **Sparsity:** Calculate the average number of features changed in the counterfactuals.
  + **Diversity:** For instances with multiple counterfactuals, compute the average pairwise distance between the counterfactuals to assess their diversity.
* **Computational Efficiency:** Measure the time taken for each stage of the pipeline (model training, counterfactual generation, visualization).

#### **3. Related Work**

**Counterfactual Explanations:**

* [Wachter, S., Mittelstadt, B., & Russell, C. (2017). Counterfactual Explanations without Opening the Black Box: Automated Decisions and the GDPR](https://arxiv.org/abs/1711.00399). Harv. JL & Tech., 31(2), 841-887.
* [Mothilal, R. K., Sharma, A., & Tan, C. (2020). Explaining Machine Learning Classifiers through Diverse Counterfactual Explanations](https://arxiv.org/abs/1905.07697). Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, 607-617.
* [DiCE: Diverse Counterfactual Explanations (DiCE) for Explaining Machine Learning Model Decisions](https://github.com/interpretml/DiCE).

**Model Selection and Comparison:**

* [Wolpert, D. H., & Macready, W. G. (1997). No Free Lunch Theorems for Optimization](https://ieeexplore.ieee.org/document/585893). IEEE Transactions on Evolutionary Computation, 1(1), 67-82.

**Visualization for Model Understanding:**

* [Doshi-Velez, F., & Kim, B. (2017). Towards A Rigorous Science of Interpretable Machine Learning](https://arxiv.org/abs/1702.08608). arXiv preprint arXiv:1702.08608.

#### **4. Experiments Plan**

**Datasets:**

We will use 5-6 publicly available tabular datasets from diverse domains, ensuring they have a variety of characteristics (e.g., different numbers of features, data types, and underlying relationships). Potential datasets include:

* **UCI Machine Learning Repository:**
  + [Adult Census Income (classification)](https://archive.ics.uci.edu/ml/datasets/adult)
  + [Breast Cancer Wisconsin (classification)](https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic))
  + [German Credit Data (classification)](https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data))
  + [Wine Quality (regression)](https://archive.ics.uci.edu/ml/datasets/Wine+Quality)
* **Kaggle:**
  + [Titanic (classification)](https://www.kaggle.com/c/titanic)
  + [House Prices - Advanced Regression Techniques (regression)](https://www.kaggle.com/c/house-prices-advanced-regression-techniques)
  + [Student Performance Factors](https://www.kaggle.com/datasets/lainguyn123/student-performance-factors) (regression)

**Baseline:**

* Train each of the selected models (Logistic Regression, Decision Tree, Random Forest, SVM, XGBoost) on each dataset using standard hyperparameter optimization (e.g., GridSearchCV or RandomizedSearchCV).
* Evaluate the baseline performance of each model using appropriate metrics for the task (e.g., accuracy, F1-score, AUC-ROC for classification; R-squared, RMSE for regression).

**Implement the Automated Pipeline:**

* Develop the Python code to automate the following steps:
  + Model training and hyperparameter optimization.
  + Counterfactual generation using the DiCE library (or a similar implementation).
  + Visualization of counterfactuals using techniques like parallel coordinates plots and tabular summaries.
  + Calculation of counterfactual quality metrics and summary statistics.

**Counterfactual Generation and Analysis:**

* For each dataset and each model, generate counterfactual explanations for a diverse set of instances from the test set. This set should include:  
  + Misclassified instances.
  + Instances where the model's prediction confidence is low.
  + A representative sample of correctly classified instances.
* Experiment with different counterfactual generation methods available in DiCE (e.g., random sampling, genetic algorithm) to explore variations in explanations.

**Visualization and Comparison:**

* Apply the developed visualizations to compare the counterfactual explanations generated by the different models for the selected instances.
* Analyze the differences in counterfactuals, focusing on:  
  + Which features are most frequently changed by each model?
  + The magnitude and direction of feature changes.
  + The plausibility and realism of the counterfactuals (based on domain knowledge, if available).
* Relate the observed differences to the known characteristics of the models (e.g., linear vs. non-linear, tree-based vs. instance-based).

**Model Selection Guidance:**

* Based on the counterfactual analysis and the overall performance metrics, develop guidelines or heuristics for model selection.
* For each dataset, identify the model(s) that appear most suitable based on these guidelines.
* Compare these recommendations to the model selection that would be made based solely on overall performance metrics.

**Evaluation:**

* Evaluate the entire pipeline based on:
  + **Model Performance:** Compare the performance of the selected models to the baseline models.
  + **Counterfactual Quality:** Report the validity, proximity, sparsity, and diversity of the generated counterfactuals.
  + **Computational Efficiency:** Measure and report the time taken for each stage of the pipeline.