**Team 6 Project Proposal: Stroke Prediction**

1. **Problem Introduction and Proposed System**

Stroke is the most significant cause of death and disability around the world. Early detection of risk is vital. In this project, we will develop a system that predicts a patient's risk of a stroke by using parameters such as age, gender, background diseases, BMI, and smoking status. The importance of early risk prediction and its potential impact on proactive medical care is significant, potentially saving countless lives and reducing life-altering consequences.

The data set used is the Stroke Prediction Dataset (Soriano, 2021) from Kaggle, the raw data contains patients with their medical information, health indicators, and labelled target variable 'stroke'. This is significant for our project because it offers a wide range of features that are directly linked to the stroke, along with rich features. The developed system will utilize deep learning algorithms to learn from the data and predict risk accurately.

The stroke dataset can be found at the following link:  
<https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset>

1. **Algorithm Selection**

Since the stroke dataset is a tabular dataset, we selected the following machine learning algorithms that are well-suited for this datatype to develop the prediction models:

* Gradient Boosting Machines
  + XGBoost
  + LightGBM
  + catBoost
* Artificial Neural Networks
* Support vector machine

These models will be trained using the cleaned dataset.

To evaluate the performance of the models listed above, we will use a technique called cross-validation to obtain reliable performance metrics. Based on the cross-validation results, we will generate scores and comparison graphs to visualize and compare the effectiveness of each algorithm.

1. **Expected Behaviors of the Systems or Type of Problem the Algorithms Investigate**

The goal of this project is to predict the likelihood of stroke occurrence in individuals using structured, non-temporal patient data, such as age, BMI, smoking status, hypertension, and other health-related attributes. The chosen models have been specifically selected for their effectiveness in binary classification tasks on tabular datasets. Each algorithm possesses unique characteristics that make it well-suited for this domain, as outlined below.

**XGBoost (Extreme Gradient Boosting) -**

XGBoost is an optimized gradient boosting framework known for its high performance and scalability. In this context, it is expected to model complex relationships between features and stroke outcomes effectively. It includes regularization to prevent overfitting and handles missing data natively, making it well-suited for clinical data.

Problem Type: Binary classification with imbalanced data.

Behavior: Learns non-linear feature interactions; provides high predictive accuracy.

**LightGBM (Light Gradient Boosting Machine) -**

LightGBM is a gradient boosting algorithm that utilizes histogram-based learning to achieve faster training speeds and reduced memory usage. It is especially useful when working with large datasets or high-dimensional feature spaces.

Problem type: Large-scale, high-dimensional tabular data.

Behavior: Efficient training; handles categorical and continuous variables; strong performance on structured data.

**CatBoost**

CatBoost is a gradient boosting algorithm that is highly effective with categorical data. It requires minimal preprocessing and automatically handles categorical encoding, making it particularly useful for medical datasets containing both categorical and numerical features.

Problem type: Mixed-type datasets with categorical features.

Behavior: Robust to overfitting; strong performance with minimal tuning; interpretable results.

**Artificial Neural Networks (ANNs) -**

ANNs are flexible, multi-layer networks capable of learning complex, non-linear patterns. In the context of stroke prediction, they are useful for modeling interactions among features that may not be easily captured by shallow models. However, they require careful regularization and more data to avoid overfitting.

Problem type: Tabular classification with complex patterns.

Behavior: Learns high-level abstractions and feature interactions; adaptable architecture.

**Support Vector Machine (SVM) -**

SVM is a powerful classifier that works by finding the optimal hyperplane that separates different classes. It performs well in high-dimensional spaces and can be effective in healthcare applications with clearly separable classes, especially when kernel functions are used.

Problem type: Binary classification in structured datasets.

Behavior: Handles small to medium sized datasets; provides stable decision boundaries; effective for both linearly and nonlinearly separable data.

**Cross-Validation (CV)**

To ensure robust evaluation and prevent overfitting, Stratified K-Fold Cross-Validation will be applied to all models. This technique ensures that each fold maintains the original distribution of the target class, which is critical in medical prediction tasks with imbalanced datasets. Cross-validation helps:

* Assess generalizability across different patient subsets.
* Improve model selection and hyperparameter tuning.
* Provide reliable performance metrics (e.g., accuracy, F1 score, AUC).

1. **Expected Issues to Focus On**

| **Issue** | **Description** |
| --- | --- |
| **Class Imbalance** | Stroke cases are rare, leading to biased models toward the majority class. |
| **Overfitting** | Complex models (e.g., ANN, XGBoost) may memorize training data without generalizing. |
| **Missing or Noisy Data** | Real-world medical data often have missing or inaccurate entries. |
| **Feature Correlation** | Some features may be redundant or strongly correlated, affecting model stability. |
| **Interpretability** | Models like ANN and boosting are hard to interpret, which is critical in healthcare. |
| **Hyperparameter Tuning** | Algorithms require careful tuning to balance bias-variance tradeoff. |
| **Generalization Across Populations** | Model may perform poorly on unseen demographics if not properly validated. |

1. **References or other resources**

* LeCun, Bengio, Hinton. (2015). Deep Learning (Nature).<https://www.cs.toronto.edu/~hinton/absps/NatureDeepReview.pdf>
* Attention is All You Need (Vaswani et al., 2017).<https://arxiv.org/pdf/1706.03762.pdf>
* Representations using RNN Encoder–Decoder for Statistical Machine Translation,<https://arxiv.org/pdf/1406.1078.pdf>
* Generating Sequences With Recurrent Neural Networks,<https://arxiv.org/pdf/1308.0850.pdf>
* PyTorch,<https://docs.pytorch.org/tutorials/beginner/introyt/introyt1_tutorial.html>
* TesnorFlow/Kera,<https://www.tensorflow.org/tutorials>
* Hugging Face Transformer,<https://huggingface.co/docs/transformers/installation>
* Transformer training and fine-tuning,<https://huggingface.co/learn/llm-course/chapter1/1>
* DeepLearningAI (Andrew Ng) – Transformers Playlist,<https://www.youtube.com/playlist?list=PLkDaE6sCZn6Ec-XTbcX1uRg2_u4xOEky0>
* DeepLearning.AI – Sequence Models (Andrew Ng, Coursera),<https://www.youtube.com/playlist?list=PLkDaE6sCZn6F6wUI9tvS_Gw1vaFAx6rd6>

**6. Team Contributions**

### **Data Cleaning & EDA Lead - Quang Tran**

1. Responsibilities:
   1. Data Preprocessing
   2. Remove rows with missing values
   3. Perform one-hot encoding (e.g., convert diagnosis to 0/1)
   4. K-Fold Cross-Validation dataset
   5. Exploratory Data Analysis
      1. Plot distributions of mean parameters
      2. Create boxplots for diagnosis readings
      3. Generate a correlation heatmap
   6. Model Building - **Artificial Neural Networks**
   7. Report Write-up
      1. Data preparation methods
      2. EDA visualizations and interpretation
   8. Collaboration
      1. Assist with model interpretation if needed
      2. Review and provide feedback to the overall final report for cohesiveness.

### **Modeling & Stats Lead - Surya Prakash**

1. Responsibilities:
   1. Inferential Statistics
      1. Conduct t-tests to analyze significant differences between features and diagnosis classes
   2. Model Building - **Gradient Boosting Machines**
   3. Report Write-up
      1. Model selection rationale and setup
   4. Collaboration
      1. Assist with other team members if needed
      2. Review and provide feedback to overall final report for cohesiveness

**Evaluation & Report Lead - Pros Loung**

1. Responsibilities:
   1. Model Building - **Support Vector Machine**
   2. Model Evaluation
      1. Generate confusion matrices (TP, TN, FP, FN)
      2. Plot ROC curves and compute AUC
   3. Produce classification reports (precision, recall, F1-score)
   4. Report Write-up
      1. Model evaluation section
      2. Final summary and conclusion
   5. Collaboration
      1. Assist with other team members if needed
      2. Review and provide feedback to overall final report for cohesiveness.

### Final Report Complication and Submission - **Pros Loung**

### Final Representation Video and Submissions - **Quang Tran**