Hypothesis/Research Question:

This study aims to evaluate the performance of logistic regression on datasets encrypted using the BFV and CKKS homomorphic encryption schemes. Initially, logistic regression will be applied to the unencrypted (plaintext) datasets to establish baseline performance metrics. Subsequently, the same analysis will be conducted on the encrypted datasets using BFV and CKKS. The results will be compared based on key evaluation metrics, including F1-score, accuracy, precision, and recall.

This data collection is to answer the following research question:

- **Can logistic regression be effectively performed on datasets encrypted with BFV and CKKS?**
- **How closely do the performance metrics of encrypted logistic regression align with those of the plaintext baseline?**

Process:

The initial phase of this project involves creating two separate Python scripts, each designed to perform logistic regression on a plaintext dataset. One script will focus on implementing the BFV encryption scheme, while the other will implement the CKKS scheme. Following this, I will manually develop a logistic regression model based on the TenSEAL tutorial, which can be found below.

https://github.com/OpenMined/TenSEAL/blob/main/tutorials/Tutorial%201%20-%20Training%20and%20Evaluation%20of%20Logistic%20Regression%20on%20Encrypted%20Data.ipynb

Once the encrypted logistic regression models are implemented using both BFV and CKKS, I will run each model on the corresponding encrypted dataset. Furthermore, I will change the BFV and CKKS parameters to measure if there are any discrepancies when security is increased. Finally, I will evaluate and compare the results by collecting performance metrics including accuracy, precision, recall, and F1-score, and generating classification reports for both the baseline (plaintext) model and the encrypted models.

The three datasets being utilized are the 3 Kaggle datasets that hold clinical data. The next step is to select a genomic dataset from XENA or develop a genomic dataset from MIMIC IV.

Note: I do not remove any columns, and I do not conduct any feature engineering besides a scale factor for the BFV encryption scheme as it only handles with integers. I'm purely just measuring if these metrics can be upheld through encryption.

Results:

Table 1: Heart Disease Dataset

		F1-Score	Precision	Recall	Accuracy
BFV (scaling factor = 100000)	**4096, [30,20,30]**	0.55	0.55	0.55	0.55
	8192, [30, 20, 30]	0.51	0.51	0.51	0.51
CKKS	**4096, [30,20,30], 2**20**	0.68	0.69	0.68	0.68
	8192, [30, 20, 30], 222**	0.70	0.70	0.70	0.70
Plain	**Not Applicable**	0.71	0.71	0.71	0.71

Table 2: Breast Cancer Dataset

		F1-Score	Precision	Recall	Accuracy
BFV (scaling factor = 100000)	**4096, [30,20,30]**	0.56	0.56	0.56	0.56
	8192, [30, 20, 30]	0.52	0.53	0.53	0.52
CKKS	**4096, [30,20,30], 2**20**	0.92	0.92	0.92	0.92
	8192, [30, 20, 30], 222**	0.91	0.91	0.92	0.91
Plain	**Not Applicable**	0.93	0.92	0.93	0.93

Table 3: Diabetes Dataset

		F1-Score	Precision	Recall	Accuracy
BFV (scaling factor = 100000)	**4096, [30,20,30]**	0.42	0.42	0.42	0.42
	8192, [30, 20, 30]	0.53	0.53	0.53	0.53
CKKS	**4096, [30,20,30], 2**20**	0.75	0.83	0.76	0.76
	8192, [30, 20, 30], 222**	0.76	0.76	0.76	0.76
Plain	**Not Applicable**	0.78	0.78	0.78	0.78

Other Notes:

Due to variability in the initial measurements, I conducted three consecutive trials to obtain the most consistent and accurate results across all datasets. I plan to apply the same testing methodology when evaluating multiple genomic datasets in the future.

The results demonstrate that the CKKS scheme is significantly more effective than BFV for logistic regression on clinical datasets. CKKS is particularly well suited for this application because it supports approximate arithmetic and floating-point operations, which are essential for regression models. To maintain high accuracy, it is important to use a large global scaling factor, typically set to 2**20 or 2**22. This scaling factor enables CKKS to handle floating-point computations effectively.

Although I attempted to replicate this scaling in BFV manually, the results were not comparable due to the fundamental differences in how each scheme operates. BFV is designed for exact integer arithmetic and lacks native support for floating-point numbers, which limits its suitability in this context.

Increasing the security level by adjusting the encryption parameters did not have a direct impact on model performance. However, a higher security level allows for the use of larger scaling factors, which improves numerical precision and ultimately enhances the effectiveness of CKKS in logistic regression tasks.