Cloud Platforms – Final Project

Description of the problem

The direct healthcare costs associated with diabetic patients are substantial, placing a significant financial burden on healthcare systems like the Spanish one (6.3-7.4% of the Spanish National Health System expenditure). Early identification of individuals at risk of developing diabetes presents an opportunity for proactive intervention, potentially reducing long-term healthcare costs. As a healthcare system like the Spanish one, there is a vested interest in identifying such individuals to offer preventive measures and manage future risks effectively.

Luckily, if a risk state for diabetes is identified early enough and a lifestyle adjustment diabetes prevention program is initiated, the risk of developing a condition can be reduced dramatically.

Thus, being able to deploy a model that is able to predict if a person is at risk of developing a diabetes condition could significantly reduce the financial burden of diabetes on the Spanish healthcare system.

In order to evaluate the financial impact of our ML model, we gathered the following information and took the following assumptions to quantify the healthcare costs related to diabetes patients in the Spanish healthcare system:

- The annual cost per diabetes patient can be quantified at about 1,290 – 1,476€. For simplification, we will calculate with the average of 1,383€.
- The annual cost per non-diabetes patient is 865€.
- The lifestyle adjustment diabetes prevention program has a one-time cost of 300€.
- The program's effectiveness is 83%.
- Assumption: When at risk for diabetes, the likelihood of developing it without the prevention program is 85%.
- We evaluate the costs of treating diabetes for a period of 3 years.

The dataset

The dataset will be attached in the submission as a csv file called "diabetes_prediction_dataset.csv" took from https://www.kaggle.com/datasets/iammustafatz/diabetes-prediction-dataset/data

The final outcome

- **Performance metrics of our XGBoost model deployed in Sagemaker:**
  - The hyperparameter tuning process has optimized the model to achieve a validation AUC of 0.978689948425293, indicating our model has

excellent capability in distinguishing between the classes for the problem at hand. The tuning results show that our model has a strong performance on the validation set, with low log loss (0.08252300322055817) and error (0.02920000688433467), suggesting high accuracy and good calibration of predicted probabilities. This model is expected to generalize well to new data, based on the validation metrics provided.

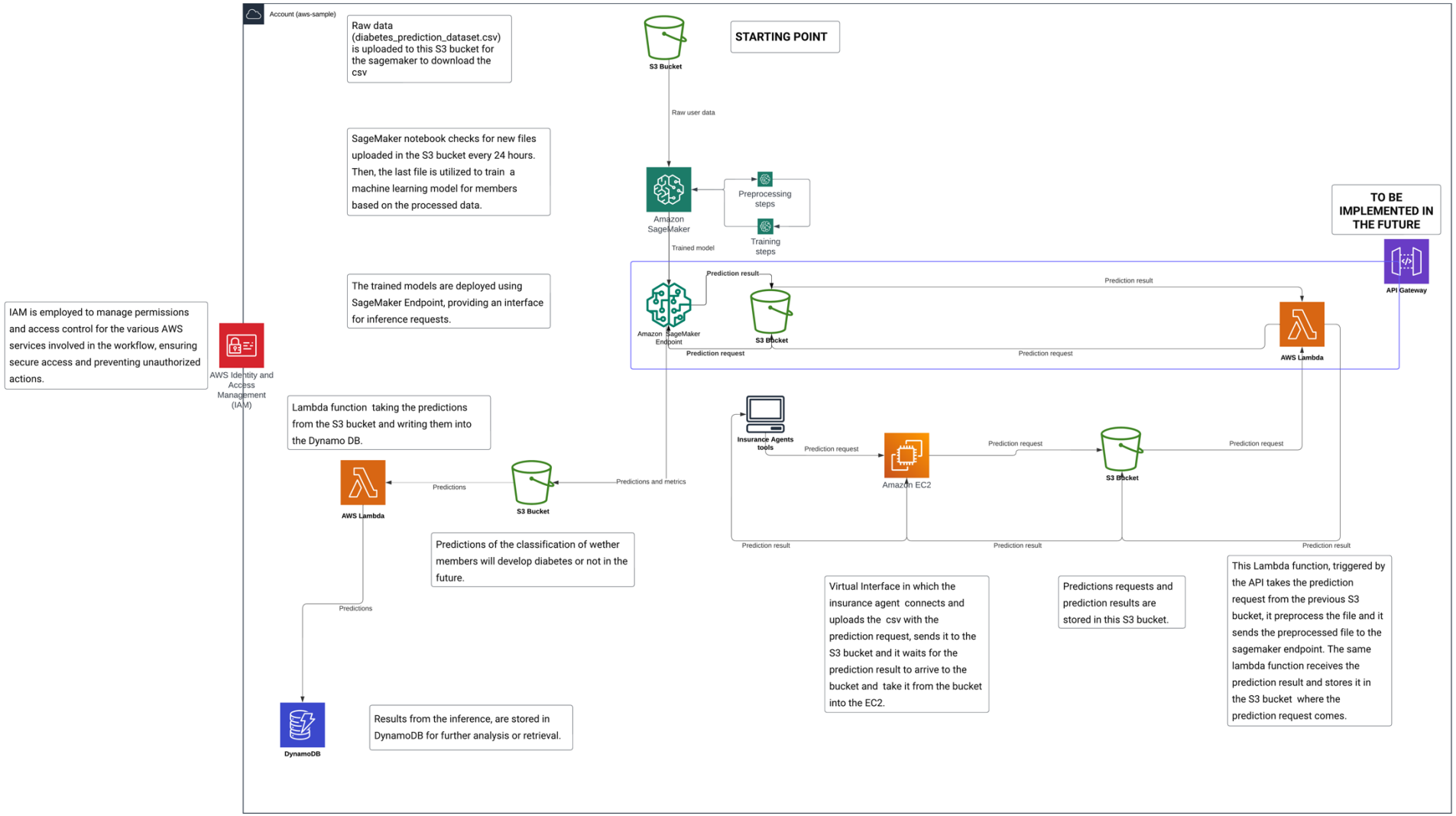
- The selected hyperparameters include a significant alpha (L1 regularization term) of 0.3225647940164943 4 and a substantial lambda (L2 regularization term) of 0.7578470196080217, suggesting a model that is robust against overfitting, by applying both L1 and L2 regularization effectively.
- A gamma value of 3.5499706277804544 specifies the minimum loss reduction required to make a further partition on a leaf node of the tree. This relatively high value means the model is more conservative and less prone to overfitting.
- The learning rate (eta) is 0.11035511928866, which is a moderate value, balancing the speed of learning and the risk of overfitting by controlling the contribution of each tree to the final model.
- The maximum depth of the trees, 'max_depth': '8', is relatively deep, allowing the model to capture complex patterns in the data, which could be beneficial given the complex nature of healthcare data.
- The 'min_child_weight' of 9.837971841334964 is relatively high, which can help prevent overfitting by making the algorithm more conservative and requiring a significant number of instances to make a child node.
- The 'subsample' rate is 0.9180113714220905, suggesting that each tree is built using approximately 91.8% of the data, which helps in preventing overfitting by adding more randomness into the model.
- The 'colsample_bytree' parameter is set to 0.55858184126595706, indicating that each tree uses around 55.9% of features, allowing the model to perform feature sampling, providing a diversity of trees and further guarding against overfitting.
- 'num_round' (the number of boosting rounds) is 1000, indicating a substantial model complexity and potential for learning intricate patterns in the data.
- The early stopping parameter 'early_stopping_rounds': '10' means that the model training will stop if the validation metric does not improve for 10 consecutive rounds, helping to prevent overfitting and unnecessary computations.

- **Interpretation of the outcome with regard to the context/business case**
  - Based on our generation of **confusion matrix** derived from measuring the model's performance against a test set of 10,000 patients, we can:
    - **Combine this with financial estimates** per field of confusion matrix

        - **Extrapolate our results** to entirety of Spanish healthcare system (46.5m people)
    - Based on those calculations, we recommend the Spanish National Health System to deploy our models as specified as they can lead to strong saving potentials as opposed to a normal world without diabetes prevention – the **potential is about EUR 2 bn**
    - As we have shown with AWS, the **deployment of such models can be easy and cost-efficient**
    - In the future, it is **key to the success** of the implementation of such a model that **reliable data on all patients is gathered**
    - Doctors and nurses must be sensitized to **check data consistency when recording the data** (e.g., regarding glucose levels)
    - It is **imperative to add the smoking history** from the doctor‘s side as smoking is a feature known to cause diabetes

Architecture diagram

- see on the following page


Account (aws-sample)
Raw data (diabetes_prediction_dataset.csv) is uploaded to this S3 bucket for the sagemaker to download the csv
S3 Bucket
STARTING POINT
Raw user data
SageMaker notebook checks for new files uploaded in the S3 bucket every 24 hours. Then, the last file is utilized to train a machine learning model for members based on the processed data.
Amazon SageMaker
Preprocessing steps
Training steps
Trained model
TO BE IMPLEMENTED IN THE FUTURE
API Gateway
The trained models are deployed using SageMaker Endpoint, providing an interface for inference requests.
Prediction result
Amazon SageMaker Endpoint
S3 Bucket
Prediction request
AWS Lambda
IAM is employed to manage permissions and access control for the various AWS services involved in the workflow, ensuring secure access and preventing unauthorized actions.
AWS Identity and Access Management (IAM)
Lambda function taking the predictions from the S3 bucket and writing them into the Dynamo DB.
Insurance Agents tools
Amazon EC2
Predictions
Predictions and metrics
Predictions of the classification of wether members will develop diabetes or not in the future.
Virtual Interface in which the insurance agent connects and uploads the csv with the prediction request, sends it to the S3 bucket and it waits for the prediction result to arrive to the bucket and take it from the bucket into the EC2.
Predictions requests and prediction results are stored in this S3 bucket.
This Lambda function, triggered by the API takes the prediction request from the previous S3 bucket, it preprocess the file and it sends the preprocessed file to the sagemaker endpoint. The same lambda function receives the prediction result and stores it in the S3 bucket where the prediction request comes.
DynamoDB
Results from the inference, are stored in DynamoDB for further analysis or retrieval.