Mathematical Modeling of Cancers Using Machine Learning Algorithms

Ananya Dutta*  
Gauhati University, Guwhati, Assam, India-781014

Abstract  
This paper shows a mathematical modeling method using different machine learning algorithms for prediction of probability of procuring Pancreatic Cancer (PC). Each algorithm reports its own accuracy, precision, recall and F1-score. Also, a Bayesian network model is used to determine the probability each subject has in contracting PC on the basis of certain preconditions, like his dietary habits and other biological attributes. This paper makes use of the PC dataset as provided by the National Cancer Institute in collaboration with National Institute of Health (NIH). The features obtained from this dataset can have either a binary value or a scalar value. The dataset consists of three questionnaires distributed to 155000 subjects. In each of these questionnaires, the subject is asked about his dietary habits and illness history.

Keywords: NIH-PLCO Dataset, Feature Selection, Bayesian Network, Prediction Model, Feature Graph-Trends

1. Introduction  
Pancreatic cancer (PC) is a disease with poor prognosis and survival rate. About 95% of people who contract PC would not make it to the five-year survival period [1]. Pancreas is an inner organ of the human body, surrounded by the duodenum and the small intestine; hence early symptoms are hard to detect [2]. Malicious cells in the pancreas are typically detected at a very advanced stage when it is impossible to save the patient. There is a pertinent need for a PC prediction model that can lead to early detection of this disease. Many researchers are in search of biomarkers for early diagnosis of PC (see for example, [3–8]). However, evidence for identified biomarkers has not been very conclusive. Image analysis and machine learning algorithms have been used for distinguishing between benign and malignant tissues in endoscopic ultrasound (EUS) and computed tomography (CT) images see for example, [9–12]. These models can detect PC only at an advanced stage and hence are not very useful.

As a follow-up on our previous work on pancreatic cancer, this paper uses machine learning algo-rithms to identify a subset of features from the PLCO dataset as useful predictors of PC [13]. The Prostate, Lung, Colorectal and Ovarian (PLCO) cancer dataset is collected by the National Cancer Institute from approximately 155,000 participants. Each participant responded to three questionnaires consisting of 65 questions (or features) about demographics, dietary habits, illness history, and family background. The dataset is highly imbalanced. To solve the unbalancing problem, we can use some data balancing algorithms to oversample the minority datapoints or undersample the majority datapoints so that both these datapoints are equal in number.

2. Problem Statement  
Our problem is to infer whether a subject has pancreatic cancer or not given information about his demographic characteristics, dietary habits, illness history, and family background. This information is encoded as a vector of features. Formally, given a set of data points \( X = \{x_1, ..., x_N\} \) and a set of labels \( \text{T rue, F alse} \), the goal is to map each data point \( x_i \) into one of the labels, where \( d \) is the dimension of each data point, and \( N \) is the number of data points in the dataset.

In this paper, we will use the PLCO dataset where \( N = 155,000 \). Each data point represents a subject as a \( d = 65 \)dimensional feature vector. The biggest challenge with the PLCO dataset is that it is highly imbalanced. Only 0.48% of the data points belong to the \( \text{T rue} \) class; rest are \( \text{False} \). To classify this imbalanced dataset, we use techniques from data visualization, data balancing using oversampling and undersampling, feature selection, and probabilistic inference. Along the way, we find interesting insights into the correlates of pancreatic cancer, some of which are consistent with what has been reported in the medical sciences/ healthcare literature while a few others are yet to be investigated thoroughly.

3. Models and Methods  
3.1 Data Visualization Methods  
t-distributed stochastic neighbor embedding (t-SNE) algorithm. t-SNE is a widely-used algo-rithm for dimensionality reduction that can be used to visualize high dimensional data by embedding the datapoints into a 2D or 3D-space. As Van der Maaten and Hinton explained [1, 14]. “The similarity of datapoint
to datapoint $x_i$ is the conditional probability $p_j i$, that $x_i$ would pick as its neighbor $x_j$ if neighbors were picked in proportion to their probability density under a Gaussian centered at $x_i$.

Adaptive Synthetic (ADASYN) algorithm. ADASYN is a method of generating synthetic examples for minority classes using a weighted distribution as shown in Figure 2. The algorithm flowchart is described in detail in [15].

3.2 Data Balancing Methods

k-means clustering. k-means clustering performed on the majority class of the dataset yielded 743 cluster centers. The value of k is based on idea of equalizing minority class with majority class and generating 743 clusters for majority class. These points were mixed with the datapoints of the minority class to remove bias, and generate a total of 1486 datapoints. The 24 prediction algorithms were run on this new dataset and the results were validated using 5-fold cross validation. Table 1 shows the validation results.

<table>
<thead>
<tr>
<th>HSF</th>
<th>KSF</th>
<th>Ref data</th>
<th>FNN</th>
<th>Logistic Regression</th>
<th>Random Forest</th>
<th>SVM</th>
<th>Deep Learning</th>
<th>CADE</th>
<th>Ensemble Boosted SVM</th>
<th>Coarse KNN</th>
<th>Medium KNN</th>
<th>Coarse SVM</th>
<th>Cubic SVM</th>
<th>Gaussian Naive Bayes</th>
<th>K nearest Neighbors</th>
<th>Logistic Regression</th>
<th>Quadratic SVM</th>
<th>Gaussian Naive Bayes</th>
<th>K nearest Neighbors</th>
<th>Logit Regression</th>
<th>Gaussian Naive Bayes</th>
<th>K nearest Neighbors</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.39</td>
<td>0.49</td>
<td>0.51</td>
<td>0.52</td>
<td>0.53</td>
<td>0.54</td>
<td>0.55</td>
<td>0.56</td>
<td>0.57</td>
<td>0.58</td>
<td>0.59</td>
<td>0.60</td>
<td>0.61</td>
<td>0.62</td>
<td>0.63</td>
<td>0.64</td>
<td>0.65</td>
<td>0.66</td>
<td>0.67</td>
<td>0.68</td>
<td>0.69</td>
<td>0.70</td>
<td>0.71</td>
</tr>
<tr>
<td>0.49</td>
<td>0.50</td>
<td>0.51</td>
<td>0.52</td>
<td>0.53</td>
<td>0.54</td>
<td>0.55</td>
<td>0.56</td>
<td>0.57</td>
<td>0.58</td>
<td>0.59</td>
<td>0.60</td>
<td>0.61</td>
<td>0.62</td>
<td>0.63</td>
<td>0.64</td>
<td>0.65</td>
<td>0.66</td>
<td>0.67</td>
<td>0.68</td>
<td>0.69</td>
<td>0.70</td>
<td>0.71</td>
</tr>
</tbody>
</table>

Table 1: Table showing accuracy, precision, recall and F1-score using 24 prediction algorithms with selected features for the feature selection algorithms for k-means clustering.
SMOTE method of oversampling. The minority class was oversampled and the new dataset was run through the algorithms. The highest accuracy was given by Fine Decision Tree of 95.4%.

Downsampling method. The majority class dataset can be downsamped by an integer sampling factor, n. It samples the dataset by keeping the first sample and then every nth sample after that. In case of several columns in the dataset, each column will be treated as a separate sequence. After feeding the downsampled dataset into 24 algorithms, the highest accuracy was reported by Quadratic SVM of 56.4% only.

3.3 Feature Selection Methods
Infinite Latent Feature Selection (ILFS) [16]. Consider a training set \( X = \{ x_1, ..., x_n \} \), such that the distribution of the values assumed by the \( r \)th features is given by \( m \) vector \( x_r \), taking into account \( m \) samples. An undirected graph \( G \) is formed so that the features are represented by the nodes and the inter-node relationships are represented by the edges. If \( a_{ij} \) is an element of the adjacency matrix, \( A \) associated with \( G \), which represents the pairwise relationship between the features \( x_i \) and \( x_j \) (\( 1 \leq i, j \leq n \)). \( G \) can be represented by the binary function \([16]\).

\[
a_{ij} = \phi (x_i, x_j) \tag{2}
\]

where \( \phi \) is a real valued potential function. The probability of each co-occurrence in \( x_i \) and \( x_j \) is framed as a mixture of conditionally independent multinomial distribution, where parameters are learned using Expectation Maximization (EM) algorithm.

Feature selection via Eigenvector Centrality (ECFS). The adjacency matrix of the above graph \( G \) can be written as \([17]\).

\[
A = \alpha K + (1 - \alpha) \sum (i,j) \tag{3}
\]

where \( \alpha \in [0, 1] \) is a loading coefficient. In Eigenvector Centrality measure (EC), \( v_i \) is calculated as the eigen vector of \( A \) associated with the largest eigen value. If \( e \) is any vector,

\[
\lim_{l \to \infty} [A^l e] = v_e \tag{4}
\]

Relieff. Relieff is an algorithm developed by Kira and Rendell in 1992 that uses filter-method approach for feature selection. If a dataset consists of \( n \) instances of \( p \) features, belonging to two classes. At each iteration, \( X \) is a feature vector belonging to one random instance and the feature vectors of the instances closest to \( X \) from each class using Manhattan L1 norm are chosen. 'Near hit' is the closest instance of the same class and 'Near miss' is the closest instance of different class. The weight vector is updated as follows \([1]\).

\[
W_i = W_i - (x_i - \text{nearHit})^2 - (x_i - \text{nearMiss})^2 \tag{5}
\]

**Feature Selection Concave (FSV).** If matrices \( A \in R^{mxm} \) and \( B \in R^{mxn} \) are two point sets, then they can be discriminated by a separating plane, \( P \) as in \([18]\).

\[
P = \{ x \mid x \in R^n, x^T w = \gamma \} \tag{6}
\]

where normal \( w \in R^n \) and 1-norm distance to the origin is defined as \( \| w \|_{1} \). Laplacian. A parameter used in this algorithm is the Laplacian Score (LS) which means that two points are related to the same topic if they are close to each other. Laplacian score of the \( r \)th feature is calculated as follows \([19]\).

\[
L_r = \frac{\bar{v}_r^T L \bar{v}_r}{\bar{v}_r^T D \bar{v}_r} \tag{7}
\]

Unsupervised Discriminative Feature Selection (UDFS). UDFS aims to select the most discriminative features for data representation, where manifold structure is considered. \( X = \{ x_1, x_2, ..., x_n \} \) is the training set, \( x_i \in R^d \) (\( 1 \leq i < n \)) is the \( i \)th datum and \( n \) is the number of data points in the training set. The objective function of this algorithm is: For an arbitrary matrix, \( A \in R^{n \times p} \), its \( l_2,1 \)-norm [20]. Is

\[
\| A \|_{2,1} = \sum_{i=1}^{R} \sqrt{\sum_{j=1}^{p} A_{ij}^2} \tag{8}
\]

Local Learning Clustering based Feature Selection (LLCFS). This algorithm constructs the \( k \)-nearest neighbor graph in the weighted feature space. It performs joint clustering and feature weight learning \([21]\).

**Correlation based Feature Selection (CFS).** This algorithm performs feature selection on the basis of the hypothesis,“good feature subsets contain features highly correlated with the classification, yet uncorrelated to each other” \([22]\). A merit function is a function that measures the agreement between data and the fitting model, for a particular choice of parameters. By definition, the merit function is small when the agreement is good. The merit function of a feature subset \( S \) consisting of \( k \) features is given as \([1]\).

\[
\text{Merit}_k = \frac{k \bar{r}_{ef}}{\sqrt{k + k(k - 1) \bar{r}_{ff}}} \tag{9}
\]

Where is the mean of all feature-classification correlations, and \( r_{ff} \) is the mean of all feature-feature correlations.

**Feature Selection with Adaptive Structure Learning (FSASL).** In this algorithm the features are ranked in descending order of their weights \([21]\). The optimal sparse combination weight matrix \( S \in R^{n \times s} \) can be obtained by solving the following problem \([23]\).

\[
\min \sum_{i=1}^{n} \left( \| x_i - X s_i \|^2 + \alpha \| s_i \|_1 \right), \text{ such that } S_{i} s_i = 0 \tag{10}
\]
Lasso. Consider a sample consisting of N cases, each of which consists of p covariates and a single outcome. If y be the outcome and \( x_i = (x_{i1}, x_{i2}, ..., x_{ip})^T \) be the covariate vector for the \( i^{th} \) case. The parameters are estimated by solving the following optimization problem [24].

\[
\hat{\beta}(\lambda) = \arg \min_{\beta} \left( \frac{1}{n} \sum_{i=1}^{n} (y_i - x_i^T \beta)^2 + \lambda \| \beta \|_1 \right)
\]  

(11)

where \( \| \cdot \|_2 = \sqrt{\sum_{i=1}^{d} \beta_i^2} \), \( \| \cdot \|_1 = \sum_{i=1}^{d} |\beta_i| \). \( \lambda \geq 0 \), is the parameter that controls the strength of the penalty.

**Dependence Guided Unsupervised Feature Selection (DGUFS).** The objective of this algorithm is to select \( m \) most discriminative features (\( m < d \)) whose learned pseudo-label indicators are much closer to the cluster groups. It can be stated by the following discrimination promotion function [25].

\[
\min_{\Theta} J(X, V, Y)
\]

(12)

where \( Y = \text{diag}(s)X, V \in \Omega, S \in \{0, 1\}^d, s^T 1_d = m, \Omega=\text{candidate set of data cluster label matrices.}

**3.4 Classification Methods**

**Bayesian network.** A Bayesian network is a directed acyclic graph with some quantitative probability information assigned to each node that corresponds to a random variable. It has many other synonyms, viz, belief networks, probabilistic network, causal network and knowledge map [26]. A conditional probability distribution \( P(x, \text{parents}(X_i)) \) defines the relationship between each node and its parents. It is defined by the following equation [26].

\[
P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | \text{parents}(X_i))
\]

(13)

where \( P(x_1, \ldots, x_n) \) = probability of joint conjunction of events \( x_1, x_2, \ldots, x_n \).

**Decision tree.** The goal attribute is true if and only if the input attributes follow the paths towards a leaf with value true. This assertion gives a decision tree and its propositional logic can be written as follows [26].

\[
\text{Goal} \iff P(\text{ath}_1 \lor V \text{P(ath}_2 \lor V \ldots
\]

(14)

In MATLAB definition, *fine trees* have the highest model flexibility as they have many leaves to make many fine distinctions between classes [27]. They allow a maximum of 100 splits. In case of medium trees, the model flexibility is *medium*. They allow a maximum of 20 splits. In case of *coarse* trees, the model flexibility is low and they allow a maximum of 4 splits.

**Logistic regression.** The logistic function is given by the following equation [26].

\[
\text{Logistic}(z) = \frac{1}{1 + e^{-z}}
\]

(15)

It gives the *probability* of belonging to the class labeled 1. The process of fitting the weights of this model to minimize loss on a data set is called logistic regression.

**RUS boosted trees.** Random under-sampling (RUS) is used to balance an imbalanced class that is a common problem for any datasets having rare occurrences of a particular event, the algorithm of which can be found in [28].

**Bagged trees.** “Bagging predictors is a method for generating multiple versions of a predictor and using these to get an aggregated predicton”- Breiman [29].

Consider data \( (y_n, x_n) \), \( n = 1, \ldots, N \) in a learning set, where the \( y \)'s are either class labels or a numerical response. If the input is \( x \) we predict \( y \) by \( \phi(x, L) \), taking repeated bootstrap samples \( L^b \) from \( L \), and forming \( \phi(x, L^b) \) and if \( y \) is numerical

\[
\phi_b(x) = \text{av}_b \phi(x, L^b)
\]

(16)

If \( y \) is a class label, let the \( \phi(x, L^b) \) vote form \( \phi_b(x) \). This is called “bootstrap aggregating” or bagging. k-means clustering. In k-means clustering [30, 1], the data \( (x_1, x_2, \ldots, x_n) \) with \( n \) observations is segregated into \( k \) clusters \((k \leq n) \), \( S = S_1, S_2, \ldots, S_k \) so that the *within cluster sum of squares* (WCSS) or variance is minimum.

\[
\arg \min_{\mu} \sum_{i=1}^{k} \sum_{x \in S_i} \| x - \mu_i \|^2 = \arg \min_{\mu} \sum_{i=1}^{k} |S_i| \text{Var } S_i
\]

(17)

where \( \mu_i \) is the centroid of cluster \( S_i \) imbalanced dataset by increasing the number of samples of the minority class. The algorithm flowchart is described in detail in [31].

**Support Vector Machine.** SVM is a type of supervised learning, where data that is not linearly separable can be easily separated by mapping them into higher dimensional space. The optimal SVM separator is found by solving the following [26].

\[
\arg \max_{\alpha} \sum_{j} \alpha_j - \frac{1}{2} \sum_{j,k} \alpha_j \alpha_k y_j y_k (x_j, x_k)
\]

(18)

where \( \alpha_j \geq 0 \) and \( \alpha_j y_j = 0 \). Solution of this equation is done using software called as quadratic programming.

**K-nearest neighbor** KNN algorithm classification is a type of clustering where the nearest \( k \) datapoints \( N(k, x) \) are considered. The distance metric is measured using Minkowski distance as follows [26].

\[
L^p(x_j, x_q) = \left( \sum_{i} |x_{j,i} - x_{q,i}|^p \right)^{\frac{1}{p}}
\]

(19)

When \( p = 2 \), it is called Euclidean distance and if \( p = 1 \), it is Manhattan distance.
3.5 Evaluation Matrices
The statistical parameters calculated are as follows [1].

\[
\begin{align*}
\text{Accuracy} &= \frac{t_p + t_n}{\text{total}} \\
\text{Precision} &= \frac{t_p}{t_p + f_p} \\
\text{Recall} &= \frac{t_p}{t_p + f_n} \\
F_1 - Score &= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}
\end{align*}
\]

where \(t_p, t_n, f_p, f_n\) are the number of true positives, true negatives, false positives, false negatives respectively.

4. Experimental Results
4.1 Dataset
The dataset used for this work is the Pancreas Cancer Dataset made accessible by the National Cancer Institute by NIH. Around 155,000 individuals have participated in the PLCO data collection [32]. Each of them have filled out three questionnaires—the Baseline Questionnaire (male-BQM/female-BQF), Other Cancer Form (OCF), and the Annual Study Update (ASU) form. Figure 2 shows the 2D representation using AdaSyn algorithm and t-SNE algorithm [14] for the down-sampled dataset using selective sampling of majority class (PC=0), down-sampled dataset using \(k\)-means clustering and oversampled minority class (PC=1) of dataset using SMOTE algorithm.

4.2 Classification
Twenty four machine learning algorithms, briefly described in Section 3 were used and their statistical parameters are reported.

Using classification ensemble. In this ensemble algorithm, the weights or costs can be modified to correctly train the algorithm to predict PC. The weights are normalized to add unity, depicting the prior probabilities. Suppose \(ij\) \((i, j \in \cdots, ii=0)\) is the cost of misclassification of the example of the \(i_a\) class to the \(j_a\) class, where \(c\) is the number of classes. Then, the weight assigned to the \(i_a\) class after rescaling is given as [33].

\[
\begin{align*}
w_i &= \frac{n \times \epsilon_i}{\sum_{k=1}^{c} (n_k \times \epsilon_k)}
\end{align*}
\]

where, \(n\) is the number of training examples. \(\epsilon_i = \sum_{j=1}^{c} \epsilon_{ij}\)

It uses the algorithms as described in [33–35]. For example, we can say the weight of predicting no PC for subjects with PC (False positive) is 1000 times more serious than predicting PC for subjects with no PC (False negative). Accordingly, we can change the weights to get a confusion matrix as per our need. Figure 1a shows confusion matrix without any weights where it improperly classified all PC cases and Figure 1b shows a very high weight applied to false positives which leads to misclassification of all non-PC cases.

![Figure 1: Figure showing confusion matrix using classification ensemble for (a) no weight, and (b) very high weights applied to False Positive.](image)

4.3 Feature Selection
Several feature selection algorithms [17–22, 24, 25, 36–38] have been used to extract features. These algorithms have been implemented in MATLAB. Table 2 shows the features selected by these algorithms. \(D\) = downsampled dataset using fixed sampling rate, \(K\) = downsampled dataset using \(k\)-means clustering. Figure 3 and figure 4 shows the variation of \(P(C—E=true)\) where \(P(E)\) = probability of the feature, choosing some of the most probable features for Pancreatic Cancer from Table 3.
<table>
<thead>
<tr>
<th>Table 2: Features(rows) selected by the different algorithms(columns) highlighting most selected features</th>
<th>&gt;=6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Education</td>
<td>D</td>
</tr>
<tr>
<td>Status</td>
<td>K</td>
</tr>
<tr>
<td>Occupation</td>
<td>K</td>
</tr>
<tr>
<td>Smoking</td>
<td>K</td>
</tr>
<tr>
<td>blood pressure</td>
<td>K</td>
</tr>
<tr>
<td>No of Sisters</td>
<td>D</td>
</tr>
<tr>
<td>No of Brothers</td>
<td>D</td>
</tr>
<tr>
<td>Use of Aspirin regularly</td>
<td>K</td>
</tr>
<tr>
<td>Use of Ibuprofen regularly</td>
<td>D</td>
</tr>
<tr>
<td>No of Tubal/ Ectopic pregnancies</td>
<td>K</td>
</tr>
<tr>
<td>Ever taken that</td>
<td>D</td>
</tr>
<tr>
<td>Ever had hypertensive breast disease?</td>
<td>K</td>
</tr>
<tr>
<td>Ever had ovarian tumor / cyst?</td>
<td>D</td>
</tr>
<tr>
<td>Ever had endometriosis?</td>
<td>K</td>
</tr>
<tr>
<td>Ever had uterine fibroid tumors?</td>
<td>D</td>
</tr>
<tr>
<td>Ever want to become pregnant for a year or so without success?</td>
<td>K</td>
</tr>
<tr>
<td>Age of first pregnancy</td>
<td>D</td>
</tr>
<tr>
<td>Age of hysterectomy</td>
<td>K</td>
</tr>
<tr>
<td>Age at first pregnancy</td>
<td>K</td>
</tr>
<tr>
<td>Taken female hormones</td>
<td>K</td>
</tr>
<tr>
<td>Total years taken</td>
<td>K</td>
</tr>
<tr>
<td>Female hormones</td>
<td>D</td>
</tr>
<tr>
<td>Any family members reported to urinate more than once in night</td>
<td>D</td>
</tr>
<tr>
<td>Age when told had enlarged prostate</td>
<td>D</td>
</tr>
<tr>
<td>Age when told had inflamed prostate</td>
<td>D</td>
</tr>
<tr>
<td>Age at vasectomy</td>
<td>K</td>
</tr>
<tr>
<td>Age at vasectomy</td>
<td>K</td>
</tr>
<tr>
<td>Ever had prostate surgeries</td>
<td>D</td>
</tr>
<tr>
<td>Ever had prostate surgeries</td>
<td>D</td>
</tr>
<tr>
<td>Ever in hospital</td>
<td>D</td>
</tr>
<tr>
<td>Race</td>
<td>K</td>
</tr>
<tr>
<td>Hispanic origin</td>
<td>D</td>
</tr>
<tr>
<td>Ever had diabetes</td>
<td>D</td>
</tr>
<tr>
<td>Ever had biopsy of prostate</td>
<td>D</td>
</tr>
<tr>
<td>Ever had transurethral resection of prostate?</td>
<td>D</td>
</tr>
<tr>
<td>Ever had prostatectomy of benign disease?</td>
<td>D</td>
</tr>
<tr>
<td>Ever had any prostate surgery</td>
<td>D</td>
</tr>
<tr>
<td>Ever been pregnant?</td>
<td>D</td>
</tr>
<tr>
<td>Ever had hysterectomy</td>
<td>K</td>
</tr>
<tr>
<td>Removed ovaries?</td>
<td>K</td>
</tr>
<tr>
<td>Ever had enlarged prostate?</td>
<td>D</td>
</tr>
<tr>
<td>Ever had inflamed prostate?</td>
<td>K</td>
</tr>
<tr>
<td>Ever had prostatitis</td>
<td>D</td>
</tr>
<tr>
<td>Ever had prostatectomy</td>
<td>D</td>
</tr>
<tr>
<td>Ever had had control pills</td>
<td>D</td>
</tr>
<tr>
<td>Ever had female hormone</td>
<td>K</td>
</tr>
<tr>
<td>Ever had family history</td>
<td>D</td>
</tr>
<tr>
<td>Ever smoke regularly</td>
<td>D</td>
</tr>
<tr>
<td>Ever smoke regularly now?</td>
<td>D</td>
</tr>
<tr>
<td>Usually filtered or not filtered?</td>
<td>D</td>
</tr>
<tr>
<td>Ever smoke cigarettes</td>
<td>D</td>
</tr>
<tr>
<td>Ever had history of pancreatic cancer</td>
<td>K</td>
</tr>
<tr>
<td>Family history of pancreatic cancer</td>
<td>D</td>
</tr>
<tr>
<td>No of relatives with pancreatic cancer</td>
<td>D</td>
</tr>
<tr>
<td>Ever had history of any cancer</td>
<td>D</td>
</tr>
<tr>
<td>Family history of any cancer</td>
<td>D</td>
</tr>
</tbody>
</table>

Int J Cancer ResTher, 2023  Volume 8 | Issue 3 | 121
Figure 2: PLCO dataset visualized in 2D using ADASYN algorithm (left) and t-SNE algorithm (right). (First row) Original dataset (total 154,897 points; False class 154,148; True class 749). (Second row) Balanced dataset using fixed-rate down-sampling (total 1486 points; False class 743; True class 743). (Third row) Balanced dataset by downsampling using $k$-means clustering (total 1486 points; False class 743; True class 743). (Fourth row) Balanced dataset by oversampling using SMOTE algorithm (total 1486 points; False class 743; True class 743).
Figure 3: Figure showing variation with $P(C|E = \text{true})$ using the most frequently chosen features from Table 3.
Figure 3: Figure showing variation with $P (C|E = true)$ using the most frequently chosen features from Table 2. The orange line is a fitting curve.
5 Conclusion
Certain features have been identified to have a direct relationship with pancreatic cancer, for example, smoking history, no. of cigarettes smoked in a day, genetics etc, whereas others have been identified by features selection algorithms and also by graphical representation to have an unproved connection with causing Pancreatic Cancer, for example, no. of brothers, total years taken female hormones [13]. Future work remains in order to find a mechanism in which a robot can predict with the highest accuracy the probability of a person having pancreatic cancer by getting answers to a set of features from the subject.

Acknowledgements
The author of this paper would like to thank Dr Bonny Banerjee and Dr Chrysanthe Preza from the University of Memphis, Tennessee and Dr Subhash Chauhan and Dr Sheema Khan from University of Texas, Rio Grand Valley, Texas for their support and guidance in writing the paper.

References