Fraud Firm Regression And Classification

1st Aishwarya Bhavsar
Department of Computer Science
California State University Long Beach
Los Angeles, United States
aishwaryavivek.bhavsar01@student.csulb.edu

2nd Aditya Sarma
Department of Computer Science
California State University Long Beach
Los Angeles, United States
madgula.sarma01@student.csulb.edu

3rdJustin Kieu
Department of Computer Science
California State University Long Beach
Los Angeles, United States
justin.kieu01@student.csulb.edu

4th Janki Patel
Department of Computer Science
California State University Long Beach
Los Angeles, United States
janki.patel01@student.csulb.edu

5th Mudra Chaudhary
Department of Computer Science
California State University Long Beach
Los Angeles, United States
MudraMonghjibhai.Chaudhary01@student.csulb.edu

***Abstract*—Fraud has always been a critical issue worldwide and can have severe consequences on economy and individuals.The Frauds can be of many types like Credit card frauds,Fraud emails or even some firms perform unfair practices.The auditors are responsible of detecting such frauds.This task needs higher accuracy in analysis which is possible by using Pattern recognition with Machine Learning Algorithms.We have built a classification and regression model that can predict the fraudulent firm on the basis of the present and historical risk factors.**

I. Introduction

The dataset contains values such as the historical risk value of a target value, the unique ID of a city or providence, and discrepancies found in the planned or unplanned expenditure of inspection and summary reports. It also contains score and risk values for each report alongside a total amount of discrepancies discovered and their historical score. The dataset contains: Rows - 776, Columns - 27 and It contains null and missing values. The information about the sectors and the counts of firms are listed respectively as: Irrigation (114), Public Health (77), Buildings and Roads (82), Forest (70), Corporate (47), Animal Husbandry (95), Communication (1), Electrical (4), Land (5), Science and Technology (3), Tourism (1), Fisheries (41), Industries (37), Agriculture (200) The project will use two CSV files AuditRisk and Trial to find suitable models with highest accuracy inorder to eliminate false negative and false positive fraud firms. We have considered AuditRisk as target column for regression tasks, and Risk(Trial.csv) as the target column for classification tasks.We will first make use of various regressors to predict the variable - 'AuditRisk' based on the various features present in the dataset.

A. Regression Models:

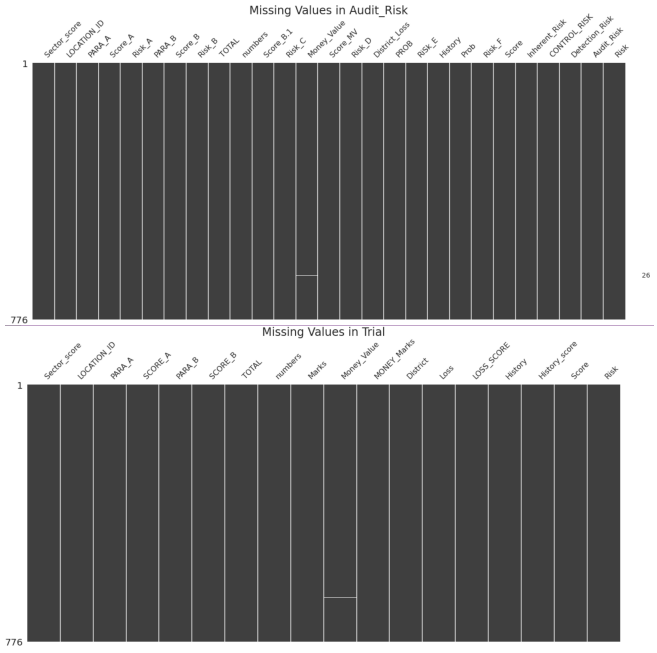
- **KNN Regressor**
- **Linear Regression**
- **Polynomial Regression**
- **Ridge Regression**
- **Lasso Regression**
- **Support Vector Machine with Kernel trick – Linear, Rbf, Poly**

B. Classification Models:

- **KNN Classification**
- **Logistic Regression**
- **Support Vector Machine - Linear SVC**
- **Support Vector Machine with Kernel trick – Rbf, Poly**
- **Decision Tree**

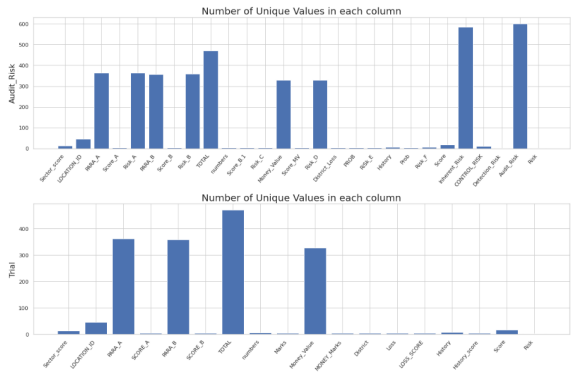
II. Data Pre Processing and Feature Selection

A. Check the Missing Values



B. Find and Remove Duplicate values

While training a supervised learning algorithm, the usual assumptions are that:

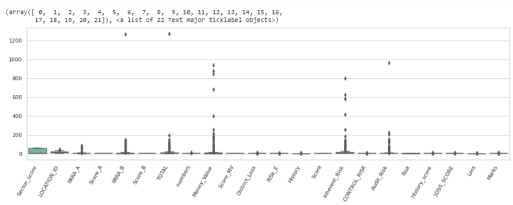


Data points are independent and identically distributed Training and testing data is sampled from the same distribution In light of these assumptions, we are removing duplicate records from both dataframes.

C. Merging Data Frames

Here, We have tried a different combination of common columns of both data frames to create a unique column which gives 763 unique values so we can merge two data frames using that common column.

D. Checking and Removing outliers

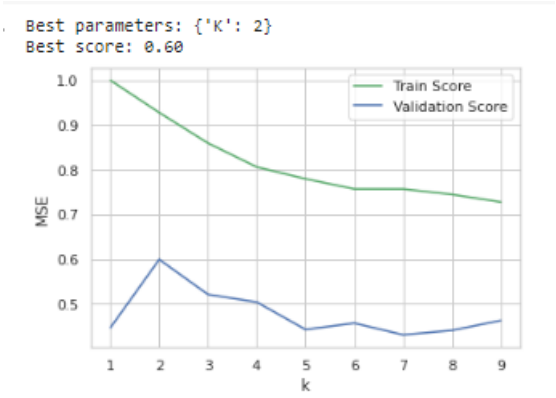


First, we tried to impute outliers with a median value of that particular feature but because of highly skewed nature of the data, the model was not fitting properly to data so we have decided to remove the outlier with an intent that we remove as less data as possible with high accuracy of the models. So, we tried a different range of outliers and finalized range in such a way that we get rid of around 5 to 6 outliers which we have seen above. We have plotted the above scatter plot again and we confirmed that we were able to remove all those outliers.

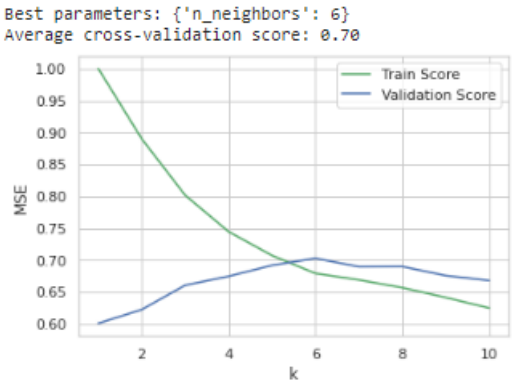
III. Application of Regression Models

A. KNN Regression

Let us fit model - KNN Regressor using train and validation set and find the best parameter - 'K' using naive grid search. To predict a new record, this method relies on finding "similar" records in the training data. These "neighbors" are then used to derive a prediction for the new record by averaging the outcome value of the K-nearest neighbors.



The best parameter value of K for this model is 2 which gives an score of 0.60 on the validation dataset. We review this parameter using cross validation. In cross validation, the dataset is split into K "folds" of equal size. Average testing performance is used as the estimate of the performance. Usually the k-folds is set as 10. The average cross-validation score for the parameter K = 2 is 0.62 Let us find the best parameter for this model using GridSearchCV: GridSearchCV allows to define a grid of parameters that will be searched using K-fold cross-validation. The best parameter for this model using GridSearchCV is 6. Eventhough we found the best parameter as K = 2 using the naive grid search, but with parameter K = 6, the average cross validation is 0.70. These results are much better when compared to average cross validation of the naive grid search. GrisSearchCV gives a better score for KNN regressor compared to naive grid search. Thus, we will choose the best parameter for KNN regressor as K = 6 because the model might run into overfitting issue for parameter K = 2.



B. Linear Regression

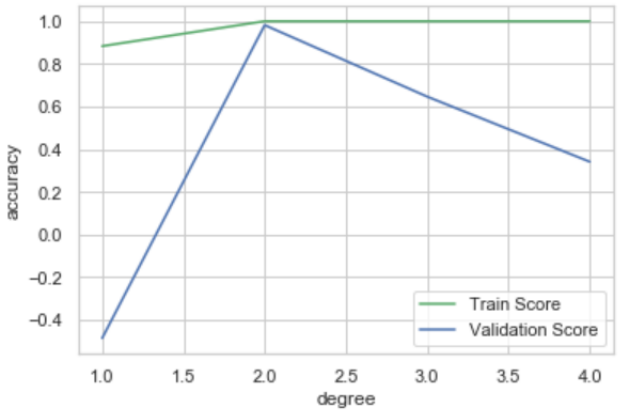
In statistics, linear regression may be a linear approach for modeling the link between a scalar response and one or more explanatory variables (also referred to as dependent and independent variables). In regression a equation is fitted to the observed data by which the connection is modeled. Out of two variables one is explanatory and the other one is a dependent variable. In regression a equation is fitted to the observed data by which the connection is modeled. Such models are called linear models. most typically, the conditional mean of the response given the values of the explanatory variables (or predictors) is assumed to be an affine function of these

values; less commonly, the conditional median or another quantile is employed. Like all varieties of multivariate analysis, simple regression focuses on the probability distribution of the response given the values of the predictors, instead of on the probability distribution of all of those variables, which is the domain of statistical method. The foremost popular model for creating predictions is the linear regression model. This model is used to fit a linear relationship between a numerical outcome variable Y and a set of predictors - X1;X2; : : : ;Xp.

As we had seen in the Feature Distribution section, the relationship between a few predictors and the outcome variable was non-linear. Because of this non-linear relationship between the target variable and the predictors, Linear Regression would not be an apt model for prediction of this dataset. By computing the cross model validation and finding the average score for our model comes close to 0.71 which is still better than KNN Regressor. However, it might not be the best regression model because of the non-linear relationship between the target variable and the predictors. Also, as there are no parameters in Linear regression, Grid Search is not required.

C. Polynomial Regression

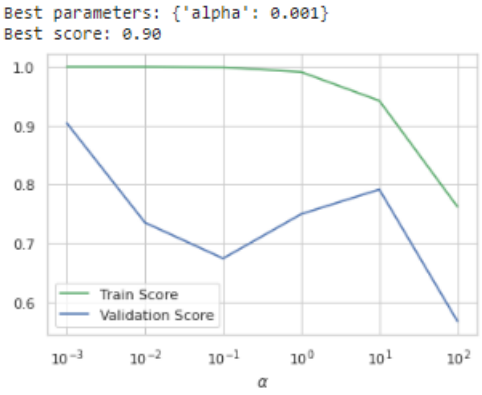
Polynomial Regression falls under the category of regression analysis where it models the relationship between the x and y variable as an nth degree. This method uses the least squares method by minimizing the variance in estimators. We try testing out polynomial regression as there is a non-linear relationship between our two variables. This makes polynomial regression more favorable than linear regression. We discovered that the 2nd degree with an accuracy score of 0.98 would be the best parameter value. We test if this is the best parameter by using cross validation score. When testing the cross validation score, we see the average is the same as what we got previously. We then run another Grid Search to check if there is another parameter that fits better. We discover that the 3rd degree would be better, but would cause overfitting. We decide to stick with the 2nd degree as it would fit the parameters better.



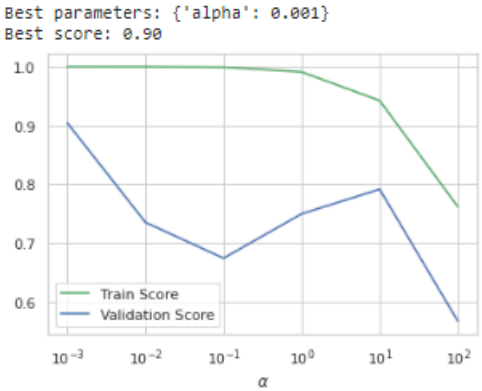
D. Ridge Regression

Ridge regression is an extension for linear regression. It's basically a regularized linear regression model. It reduces the model complexity by coefficient shrinkage. It shrinks the parameter. As a result, it is widely used to prevent multi-collinearity. It uses L2 regularization technique. We control the penalty term by changing the values of alpha. The penalty is big when the value of alpha is high. As a result the magnitude of coefficients are reduced.

Also, as there is a non-linear relationship between the variables. So, we fit the train and validation set using PolynomialFeatures and apply Ridge Regression instead of Linear Regression on the fitted data. We will consider degree for PolynomialFeatures to be 2 as it is the best parameter for Polynomial Regression model.

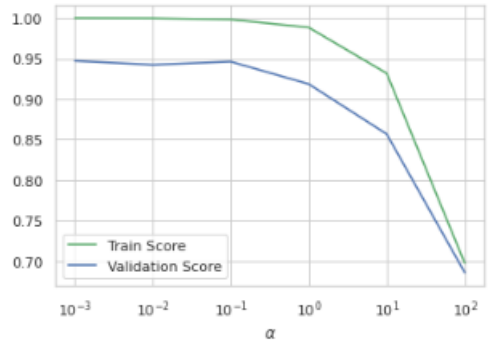


The best parameter value of alpha for this model is 0.001 giving a score of 0.90 on the validation dataset. Let's check the parameter tuning using cross validation score.



Applying this regularization technique on the train+validation set using the value of alpha = 0.001 found using naive grid search gives an average cross-validation score of 0.96. Let's check for the best parameter value of alpha using GridSearchCV.

```
Best parameters: {'alpha': 0.001}
Best cross-validation score: 0.95
```

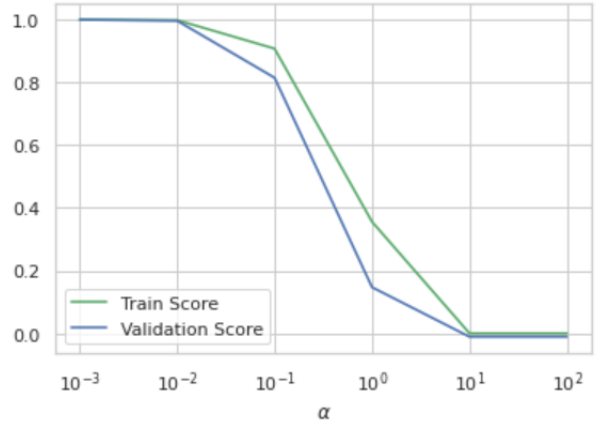


E. Lasso Regression

Lasso regression method to shrink and select a variable for linear regression models. The purpose of lasso regression is to determine the subset of predictors that produces the least amount of prediction error for a quantitative response variable. The lasso accomplishes this by imposing a constraint on the model parameters that leads some regression coefficients to decrease toward zero. After the shrinking process, variables having a regression coefficient of zero are removed from the model. Different values of alpha are tried to control the penalty in the model.

Also, as done in Ridge regression, we fit the train and validation set using PolynomialFeatures and apply Lasso Regression instead of Linear Regression on the fitted data. We will consider degree for PolynomialFeatures to be 2 as it is the best parameter for Polynomial Regression model.

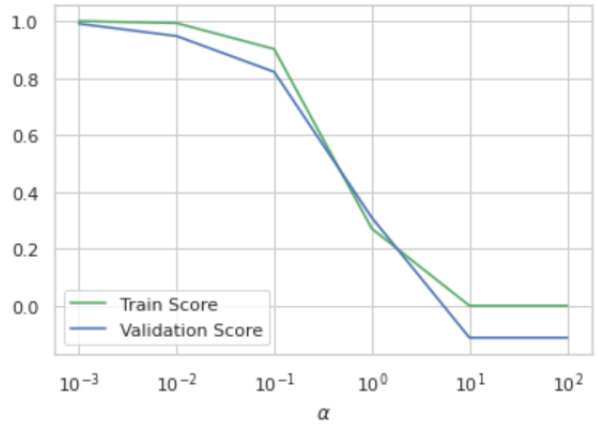
```
Best score: 1.00
Best parameters: {'alpha': 0.001}
```



The best parameter value of alpha for this model is 0.001 which ultimately gives a perfect score of 1.00 on the validation dataset. Cross validation score function yields an score of 0.99.And Hence,It signifies that this model performs good in identifying Auditrisk on the validation dataset.

Further with GridSearchCV we got same alpha value as the best parameter with a score of 0.99.

```
Best parameters: {'alpha': 0.001}
Best cross-validation score: 0.99
```



The above graph shows that with the increase in the value of alpha, the training and validation score becomes 0 due to which this model has penalized the magnitude of the coefficients to 0 for higher values of alpha.

F. SVM Regression with Kernel Trick

SVM with kernel trick

Kernel trick is a way to make optimization efficient when there are features having a linear or non linear decision boundary. Using the kernel trick, we find a higher dimensional space in which these points(linear or non-linear) were linearly separable; Map the original features to this higher transformer space;We capture the set of weights corresponding to the decision boundary hyperplane; After that we map this hyperplane into the original 2D space to obtain a linear/non-linear decision boundary.

In another way, kernel is a way to change the relative dimensions of the data points. It's possible to create a very complex decision boundary based on a high dimensional feature mapping because of the kernel representation.

SVM - rbf(Radial Basis Function)

The parameters that effect this model are parameters gamma and C.

Parameter C behaves as the regularization parameter. The gamma parameters are the inverse of the radius of influence of samples which are selected by the model as support vectors.

The gamma parameter affects the model. The model is sensitive for this paramter. The radius of the area of influence of the support vectors only includes the support vector when the gamma is exceptionally high/very large. Even if we try to regularize the C parameter, we can't prevent overfitting.

```
svr_rbf_grid = SVR(kernel='rbf', gamma = best_Gamma, C = best_C)

scores = cross_val_score(svr_rbf_grid, X_trainval, y_trainval, cv = 10, scoring
print("Cross-validation scores: {}".format(scores))

print("Average cross-validation score: {:.2f}".format(scores.mean()))
```

```
Cross-validation scores: [0.95028014 0.97814188 0.94131845 0.92768304 0.94773525
 0.59650107 0.99684185 0.88086693 0.87799994]
Average cross-validation score: 0.90
```

The model is constrained when gamma is small. Hence, it cannot capture the complexity or in other words the ”shape” of the data.

SVM - Poly

Apart from gamma and regularization parameter C, this model takes into account another parameter degree which defines the non-linear relationship between the features and the target variable. The best parameters for this model are Degree = 2 and C = 100 giving a validation score of 0.66.

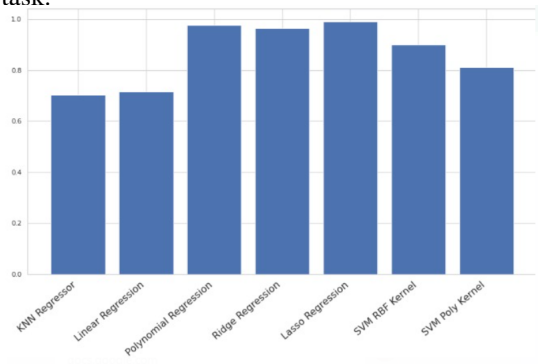
```
Cross-validation scores: [0.83990545 0.95554967 0.85246861 0.81775571
 0.47644656 0.95413367 0.79991031 0.8840575 ]
Average cross-validation score: 0.81
```

From the results of naive grid search and GridSearchCV, it can be said that the best parameters for this model are C = 100 and degree = 2.

IV. Best Regressor

As can be seen from the image average cross validation score for lasso regression is the highest and will be best model for regression task.

	Regressor	Average Cross Validation Score	Best Parameters
0	KNN Regressor	0.702636	{'n_neighbors': 6}
1	Linear Regression	0.714401	
2	Polynomial Regression	0.976232	{'N': 2}
3	Ridge Regression	0.964869	{'alpha': 0.001}
4	Lasso Regression	0.990546	{'alpha': 0.001}
5	SVM RBF Kernel	0.898714	{'gamma': 0.1, 'C': 100}
6	SVM Poly Kernel	0.810626	{'degree': 2, 'C': 100}

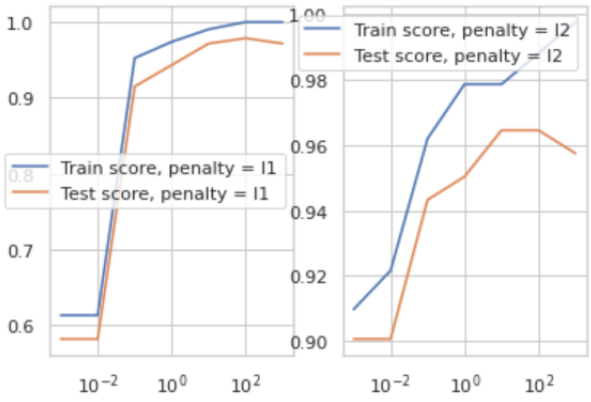


A. SVM Regression With Kernel Trick

V. Application of Classification Models

A. Logistic Regression

Logistic Regression is another form of regression analysis that focuses on modeling the probability of an outcome based on an input. The common outcome is to values such as true/false, anything that has only 2 values. Logistic Regression most useful analysis is when determining if a new sample fits into any category. With logistic regression, we see that using penalty l1 and parameter 100 give the best accuracy score of 0.98. We double check by using cross validation to ensure that this is the best parameter. We see that the average is 0.99 when performing cross validation. We finally use Grid Search again to check for the best parameter. The grid search returns the best parameter as 100 and the penalty as l1. We decide to stick with 100 and the penalty as l1.

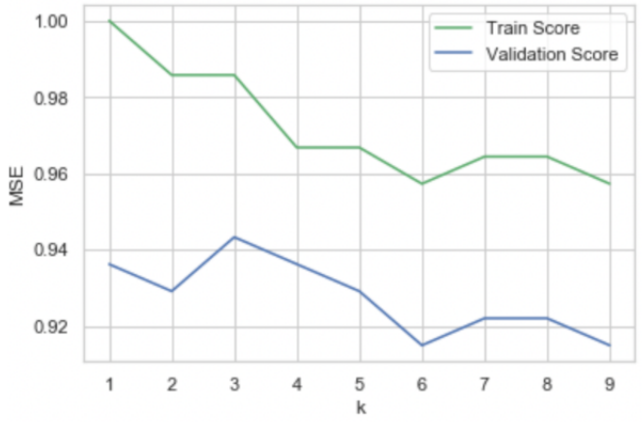


B. KNN Classification

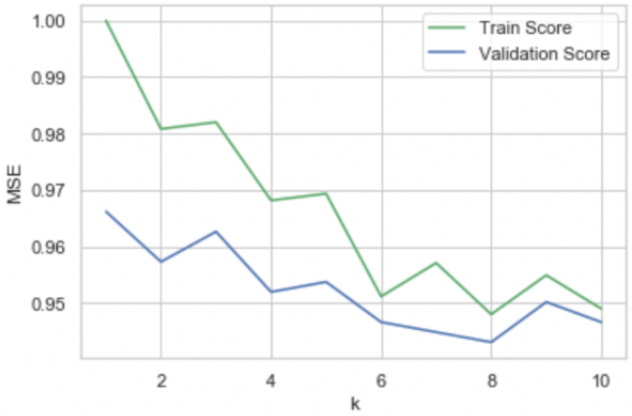
In statistics, the k-nearest neighbors algorithm (k-NN) could be a non-parametric supervised learning method first developed by Evelyn Fix and Joseph Hodges in 1951, and later expanded by Thomas Cover. it’s used for classification and regression. In both cases, the input consists of the k closest training examples in a very data set. The output depends on whether k-NN is employed for classification or regression: In k-NN classification, an object is classed by a plurality vote of its neighbors, with the article being assigned to the category commonest among its k nearest neighbors (k could be a positive integer, typically small). If k = 1, then the thing is solely assigned to the category of that single nearest neighbor.

k-NN may be a sort of classification where the function is just approximated locally and every one computation is deferred until function evaluation. Since this algorithm relies on distance for classification, if the features represent different physical units or are available at vastly different scales then normalizing the training data can improve its accuracy dramatically. Both for classification and regression, a useful technique will be to assign weights to the contributions of the neighbors, in order that the nearer neighbors contribute more to the common than the more distant ones. as an example, a standard weighting scheme consists in giving each neighbor a weight of 1/d, where d is that the distance to the neighbor. The neighbors are taken from a group of objects that the category (for k-NN classification) or the item property value (for k-NN regression) is understood. this may be thought of because the training set for the algorithm, though no explicit training step is required.

The idea in k-nearest-neighbors methods is to spot k records within the training dataset that are kind of like a replacement record that we wish to classify. We then use these similar (neighboring) records to classify the new record into a category, assigning the new record to the predominant class among these neighbors.



best parameter value of K for this model is 3 which gives an accuracy of 0.94 on the validation dataset. The average cross-validation score for the parameter K = 3 is 0.96.



The best parameter for this model using GridSearchCV is 1. Eventhough we found the best parameter as K = 3 using the naive grid search, but with parameter K = 1, the average cross validation is 0.97 which is better than the average cross validation of the naive grid search. But we know that for parameter K = 1 in KNN-Classifier, the model is overfitted eventhough it is giving a good average cross-validation score.

Confusion Matrix

	precision	recall	f1-score	support
0.0	0.98	1.00	0.99	118
1.0	1.00	0.96	0.98	70
micro avg	0.98	0.98	0.98	188
macro avg	0.99	0.98	0.98	188
weighted avg	0.98	0.98	0.98	188

Thus, the accuracy and recall of this model for the best parameter - K are 0.984 and 0.957.

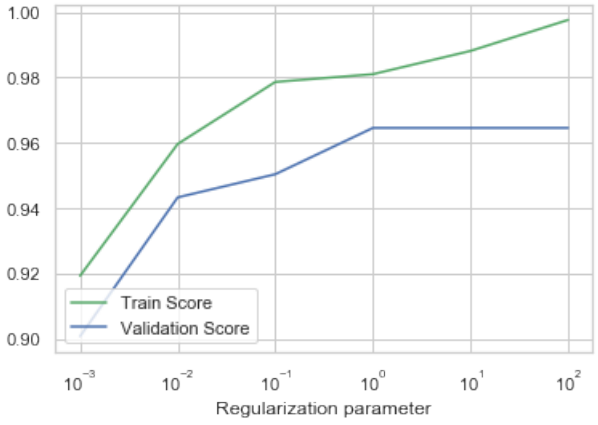
C. Support Vector Machine

SVMs are a class of supervised learning algorithms for classification, regression, and outlier identification. Support vector machines provide the following advantages: In high-dimensional spaces, it works well. When the number of dimensions exceeds the number of samples, the method is still successful. It is memory efficient because it uses a subset of training points (called support vectors) in the decision function.

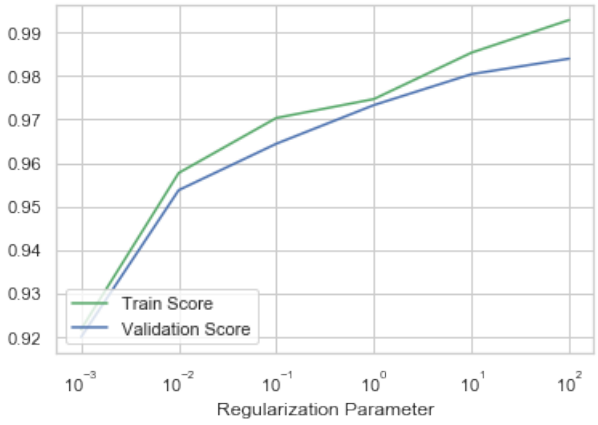
Different Kernel functions may be given for the decision function, making it versatile. Common kernels are included, however custom kernels can also be specified.

1) Linear SVC (Linear Support Vector Classifier) :

The objective of a Linear SVC is to fit to the data and, returning a ”best fit” hyperplane that divides, or categorizes, the data. From there, after getting the hyperplane, then feed some features to the classifier to see what the ”predicted” class is. The parameter that affects this model is the regularization term C. That controls the trade off between achieving a low training error and a low testing error that is the ability to generalize your classifier to unseen data. The penalty is a squared l2 penalty. The bigger this parameter, the less regularization is done and more features are added in the model.



Using LinearSVC, we get a score of 0.96 for the regularization parameter C = 1. We get an average cross-validation score of 0.96 for C = 1.



From the results of both the grid search methods - naive

grid search and GridSearchCV, we get the same average cross-validation score of 0.96 for regularization parameter c=100 Also, from the graph we see that for this model, with the increase in the value of regularization parameter i.e. when less regularization is done, the model has more features and performs better compared to when more regularization is done.

Thus, the accuracy and recall of this model for the best parameter - C=100 are 0.96

2) SVM RBF(Radial Basis Function):

RBF is the default kernel in sklearn's SVM classification method, and it can be stated as follows: gamma may be manually adjusted and must be greater than zero. A Kernel Trick is a basic way for projecting non-linear data into a higher-dimensional space to make it simpler to classify the data where it may be split linearly by a plane.

```
0.9787234042553191
Confusion matrix:
[[117   1]
 [  3  67]]
              precision    recall  f1-score   support

           0       0.97      0.99      0.98       118
           1       0.99      0.96      0.97        70

    accuracy                           0.98       188
   macro avg       0.98      0.97      0.98       188
weighted avg       0.98      0.98      0.98       188
```

The hyper-parameters for this model are gamma and regularization term C. The best paramaters for this kernel are gamma = 0.01 and C = 100 which gives a score of 0.96 on the validation set. The best parameters that we got from GridSearchCV are C = 100 and gamma = 0.1. Compared to the best parameters we got from naive grid search (C = 100 and gamma = 0.01), the average cross-validation score increased from 0.97 to 0.98.

Thus the best parameters for this model are C = 100 and gamma = 0.1.

3) SVM poly :

The polynomial kernel is a function that represents the similarity of vectors (training samples) over polynomials of the original variables, allowing learning of non-linear models.

```
0.9787234042553191
Confusion matrix:
[[117   1]
 [  3  67]]
              precision    recall  f1-score   support

           0       0.97      0.99      0.98       118
           1       0.99      0.96      0.97        70

    accuracy                           0.98       188
   macro avg       0.98      0.97      0.98       188
weighted avg       0.98      0.98      0.98       188
```

The additional hyperparameter in this model is degree along with regularization term C and gamma.The best paramaters for this kernel are degree = 1, gamma = 100 and C = 10 which gives a score of 0.98 on the validation set. The average cross-validation score for these parameters is: 0.96.

Now The best parameters that we got from GridSearchCV are degree = 1, C = 100 and gamma = 100. Which gives the cross validation score as 0.99.Compared to the best parameters we got from naive grid search (degree = 1, C = 10 and gamma = 100), the average cross-validation score increased from 0.96 to 0.99.

Thus the best parameters for this model are degree = 1, C = 100 and gamma = 100

D. Decision Tree Classification

A decision tree is a flowchart-like structure with an internal node representing features, the branch representing a decision rule, and each leaf node representing the outcome which helps in decision making usecases. The topmost node in a decision tree is called the root node. The Decision tree model learns to partition on the basis of the attribute value.

Since implementation does not use maxdepth in classifier, nodes are expanded until all leaves are pure resulting an accuracy of 1.00 on the training set as well as testing set. The model gave 0.99 cross validation score.GridSearchCV is not used there are no paramteres in this model.

```
Confusion matrix:
[[118   0]
 [  0  70]]
              precision    recall  f1-score   support

           0       1.00      1.00      1.00       118
           1       1.00      1.00      1.00        70

    accuracy                           1.00       188
   macro avg       1.00      1.00      1.00       188
weighted avg       1.00      1.00      1.00       188
```

The Accuracy and recall value of this model are 1.0 and 1.0 respectively.

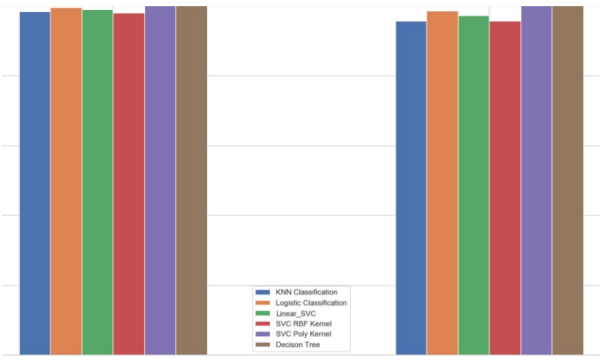
VI. Conclusions on Best Classification

On calculating the accuracy and recall value of each classification model The model has the best accuracy and the best recall value is chosen to avoid misclassification of the firms that are fraudulent but classified as not fraudulent. Also further it has been evaluated based on false negatives in the confusion matrix

	Classifier	Best Parameters	Accuracy_Score	Recall of 1
0	KNN Classification	{'n_neighbors': 1}	0.984043	0.957143
1	Logistic Classification	{'C': 100, 'penalty': 'l1', 'solver': 'libline...	0.994681	0.985714
2	Linear_SVC	{'C': 100}	0.989362	0.971429
3	SVC RBF Kernel	{'C': 100, 'gamma': 0.1}	0.978723	0.957143
4	SVC Poly Kernel	{'C': 100, 'gamma': 0.1}	0.978723	0.957143
5	Decision Tree		1.000000	1.000000

```
Classification_Scores=pd.DataFrame(Classification_Scores)
Classification_Scores.rename({0:'Accuracy_Score',1:'Recall'})
```

	KNN Classification	Logistic Classification	Linear_SVC	SVC RBF Kernel	SVC Poly Kernel	Decison Tree
Accuracy_Score	0.984043	0.994681	0.989362	0.978723	0.978723	1.0
Recall	0.957143	0.985714	0.971429	0.957143	0.957143	1.0



VII. Conclusions

From the above graph, it can be said that we are getting an accuracy of 1 and a recall score of 1 for two models:

SVM - Poly Decision Tree Thus, any of the above two classifiers can be used as a classifier for predicting the class of Risk.

SVM Poly uses the kernel trick to turn a linearly inseparable problem into a linear separable problem, while decision trees split the input into hyperrectangles according to the target.

However, for the classification problem, Decision Tree is more useful over SVM - Poly as it is nonparametric, and does not make any assumption on the distribution of data and the structure of the true model. Thuss requires less data cleaning and is not influenced by outliers .

References

[1] W. Pan, "Fraudulent Firm Classification Using Monotonic Classification Techniques," 2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), 2020, pp. 1773-1776, doi: 10.1109/ITAIC49862.2020.9338892.

[2] Y. Sun, Z. Ma, X. Zeng and Y. Guo, "A Predicting Model For Accounting Fraud Based On Ensemble Learning," 2021 IEEE 19th International Conference on Industrial Informatics (INDIN), 2021, pp. 1-5, doi: 10.1109/INDIN45523.2021.9557545.

[3] Chambers, J. M. 1977. Computational methods for data analysis. Technical report, New York. Cosserat, G. 2009. Accepting the engagement and planning the audit. In Modern auditing, ed.

[4] G. Cosserat and N. Rodda, 3rd ed., 734–36. John Wiley Sons.

[5] Couceiro, M. 2016. Particle swarm optimization. In Fractional order darwinian particle swarm optimization: Applications and evaluation of an evolutionary algorithm, 1–10. Boston, MA: Springer.

[6] Fanning, K. M., and K. O. Cogger. 1998. Neural network detection of management fraud using published financial data. International Journal of Intelligent Systems in Accounting, Finance Management 7 (1):21–41. doi:10.1002/(SICI)1099-1174(199803)7:1¡21::AIDISAF138¿3.0.CO;2-K.

[7] Fawcett, T. 2006. An introduction to roc analysis. Pattern Recognition Letters 27 (8):861–74. doi:10.1016/j.patrec.2005.10.010.

[8] Finney, D. J. 1992. Miscellaneous problems. Probit Analysi 4:140–45. JSTOR.

[9] Freund, Y., R. Schapire,and N. Abe, 1999. A short introduction to boosting. Journal of Japanese Society For Artificial Intelligence 14:771–80.

[10] Green, B. P., and J. H. Choi. 1997. Assessing the risk of management fraud through neural network technology. Auditing 16 (1):14–16.

[11] Houston, R. W., M. F. Peters, and J. H. Pratt. 1999. The audit risk model, business risk and audit-planning decisions. The Accounting Review 74 (3):281–98. doi:10.2308/ accr.1999.74.3.281.

[12] Iba, W., and P. Langley 1992. Induction of one-level decision trees. In Proceedings of the ninth international conference on machine learning, Moffett Field, California. 233–40.

[13] Keerthi, S. S., and E. G. Gilbert. 2002. Convergence of a generalized smo algorithm for svm classifier design. Machine Learning 46 (1–3):351–60. doi:10.1023/A:1012431217818.

[14] Kennedy, J. 2011. Particle Swarm Optimization. In Encyclopedia of Machine Learning, Boston, MA: Springer.

[15] Kothari, V. 2012. A survey on particle swarm optimization in feature selection, 192–201. Berlin, Heidelberg: Springer.

[16] Kotsiantis, S. 2006. Forecasting fraudulent financial statements using data mining. International Journal of Computational Intelligence 3 (2):104–10.

[17] Liaw, A., and M. Wiener. 2002. Classification and regression by randomforest. R News 2 (3):18–22.

[18] Maria, L., C. Murphy, and K. Tysiac. 2015. Data analytics helps auditors gain deep insight, 52– 54, New York: Journal of Accountancy.

[19] Neapolitan, R. E., et al. 2004. Introduction to Bayesian Networks. In Learning Bayesian Networks, 40. Chicago: Pearson

[20] Nikolovski, P., I. Zdravkoski, G. Menkinoski, S. Dicevska, and V. Karadjova. 2016. The concept of audit risk. International Journal of Sciences Basic and Applied Research (IJSBAR) 27 (3):22–31.