ANWESH: International Journal of Management and Information Technology
7 (2) September 2022, 10-20
http://www.publishingindia.com/anwesh

Predictive Modelling for Transportation Security Administration Claims Data

Lu Xiong

Department of Mathematical Sciences, Middle Tennessee State University,
Murfreesboro, Tennessee, USA. Email: Lu.Xiong@mtsu.edu

Abstract: Travelers can file claims against Transportation Security Administration (TSA) if their baggage are damaged or lost during screening. After reviewing the claim, TSA will make the decision either approve or deny the claim. The data is published by TSA each year. It is an important data to understand the baggage damageloss, but it's underused by both researchers and industry. This article explores the models with high accuracy and interpretability that can be used to predict whether a TSA claim will be approved or not. The data columns used in this research include claim type, site, claim amount, and disposition as well as airport code, airline name, etc. The clustering method is used to combine the levels in the factor variables such as airport, airline. We first used grid search and cross-validation methods to tune a single decision tree. Then a boosted tree is built. The generated linear models (GLM) with 3 different regularization methods are applied to predict the probability of claim approval: LASSO, Ridge and Elastic Net. The GLM with LASSO is chosen as the final model because of its great interpretability and high accuracy. The optimized cutoff probability to convert the GLM probability to claim approval/deny class is also discussed. This research is significant for insurance companies to develop travel insurance, for travelers to estimate their proper efforts to be invested in the claims, and for TSA to better understand the baggage loss and improve their management.

Keywords: GLM with regularization, Predictive modeling, Tree-based methods, TSA claims data.

I. Introduction

The Transportation Security Administration (TSA) was established by the Aviation and Transportation Security Act (United States Congress, 2001) [1] in response to the 9/11 terrorist attacks. The main duty of TSA is toprotect the safety of the nation's transportation systems. TSA oversees security and inspects items at 450+ airports across the USA. Just in 2015, the TSA screened over 2 billion bags and more than 708 million passengers (Lowe, 2016) [2]. However, it is unavoidable the bags could be damaged, lost, or stolen during the screening process. If this happens, the traveler may make a claim against the TSA for the losses for monetary reimbursement. After investigating the claim, which may take up to 6 months, the TSA will either approve the claim and reimburse the full amount, settle the claim for a lower sum, or deny the claim altogether. The Homeland Security reported data of every claim that happened between 2002 and 2017 on its website (Homeland Security, 2021) [3].

Little research has been done on this important data set. Kelly and Wang (Kelly and Wang, 2020) [4] mentioned this underused data set would be particularly useful to analyze insurance claim frequency and severity. They provided the summary statistics and correlation analysis of this data; however, no predictive analysis is provided. Ciullo (Ciullo, 2017) [5] studied how to choose the airline (JetBlue or Delta) with a higher probability of approving their travel claims. But they only used basic regression and data visualization, which didn't fully explore this valuable dataset. Correia and Wirasinghe (Correia and Wirasinghe, 2010) [6] used the regression method studied on the data from 62 passengers to evaluate the level of service. Its data is not as big as the TSA data we use in this paper. Franks (Franks 2007) [7] discussed the law related to the airline liability for loss, damage, or delay of passenger baggage, but no quantitative discussion is included. The model developed in its research is the psychometric scaling technique. Kyseľová (Kyseľová, 2010) [8] proposed to use insurance as a tool for risk management manage in civil air transport. The author listed risks in air transport and evaluated the identified risks using a risk matrix. He discussed how travel insurance is significant in protecting travelers against baggage loss and property damage during travel. This indicates a predictive model for such losses would also be helpful for the travel insurance industry. However, the research lacks the quantitative analysis to support its statements.

The goal of our research is to construct a model from several candidate models that will accurately predict whether a claim will be denied or approved by TSA. Compare with the few existing research on this data, our research scope is wider. First, our research is not limited to one or two airlines, but all

the airlines listed in the TSA claim data. Second, our methods such as boosted trees and LASSO are more advanced than the regression used in previous research. Third, our goal is not only to help select the better airline but also to understand the factors (including airline, airport, claim type, claim amount, etc.) that contribute to the probability of claim approval.

We will first explore the data to see which predictors would potentially be good predictors with strong predictive power. Then the predictors' such as Airline Name with lots of levels are reduced to a few levels to improve their predictive powers against the target variable. The levels are reduced either by using clustering or comparing the similarity of the target variable statistics among the levels. After the data preprocessing is accomplished, 3 different decision trees will be built or discussed: single decision tree, boosted tree, random forest. Then we consider the generalized linear model (GLM) and regularization to predict the probability of claim approval. We will also provide a demo of how to use this model.

II. Data Description, Visualization, and Pre-Processing

The data used in this paper range from year 2008 to 2012. 8 columns are contained in the working data. In original data, Disposition can be Approve in Full, Settle, or Deny. We combine Approve in Full and Settle to Approve, so the prediction task becomes a binary classification task.

Table I: Data Dictionary

Variable Name	Data Type and Definition	Values
Report.Lag	A non-negative integer, the days from Incident Date to Date Received.	Ranging from 0 to 583 (days).
Airport.Code	An airport code is a three- or four-letter code used to identify a particular airport.	"BNA", "NYL" etc. 349 different values in total.
Airline.Name	String, the name of the airline the claiming passenger took.	"Jet Blue", "American Airlines" etc. 141 different values in total.
Claim.Type	String. There are 5 different claim types.	Passenger Property Loss Property Damage Employee Loss (MPCECA) Passenger Theft Personal Injury.
Claim.Site	String. There are 3 different Claim sites.	Checked Baggage Checkpoint Others.
Item	String. The item claimed.	There are hundreds of different categories for the predictor Item, including currency, cell phones, etc.
Claim.Amount	Numerical. The amount of loss claimed in dollars.	Ranging from $1 to $5,500,000.
Disposition	String. The decision of TSA whether to approve or deny the claim.	A binary variable with value "Approve", or "Deny".

Few rows with missing values are removed from our data and only the positive claim amounts are kept. The data contains 7 independent variables (predictors) and 1 dependent variable (target).

The variable Disposition is the target variable, the other variables are in the data are the predictors. 1 is representing approve, and 0 for deny. There are a total of 24638 rows of data, where 7270 (30%) rows with the target value is 1 and 17368 (70%) rows with the target value of 0.

We explore the predictors to find out the variables with potentially high predictive power. The first predictor we will check is Report.Lag, which is defined as DateRecorded minus IncidentDate. The histogram of predictor Report.Lag is shown in Fig. 1. It follows a right-skewed distribution.

We suspect the longer Report.Lag could lead to a higher deny rate because intuitively the older claims are less likely to be approved. To investigate this, we plot the histogram and box plot by denied claims group and approved claims group. The results are listed in Fig. 2 and Fig. 3. It turns out there is no significant difference in the values of Report.Lag days between denied claims and approved claims. Hence the predictor Report.Lag may have weak predictive power.

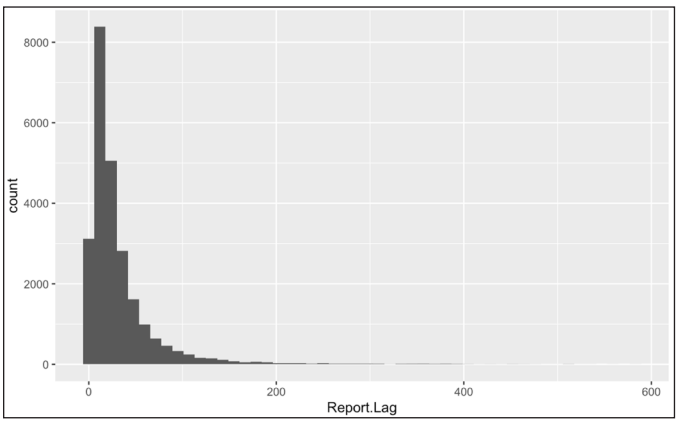


Fig. 1: The Histogram of Predictor Report.Lag

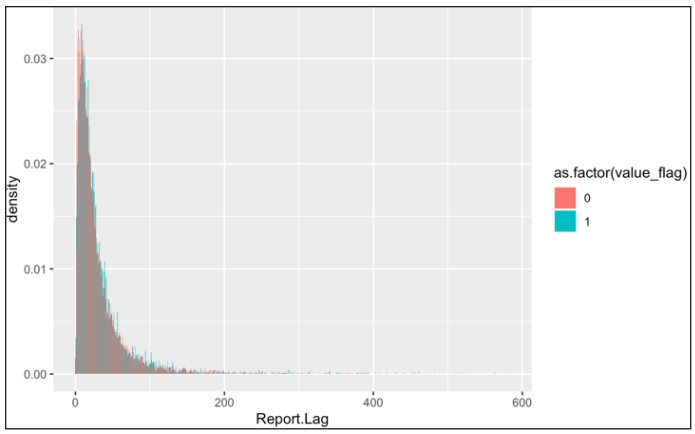


Fig. 2: The Histogram of Predictor Report.Lag for Approved Claims (Blue) and Denied Claims (Red)

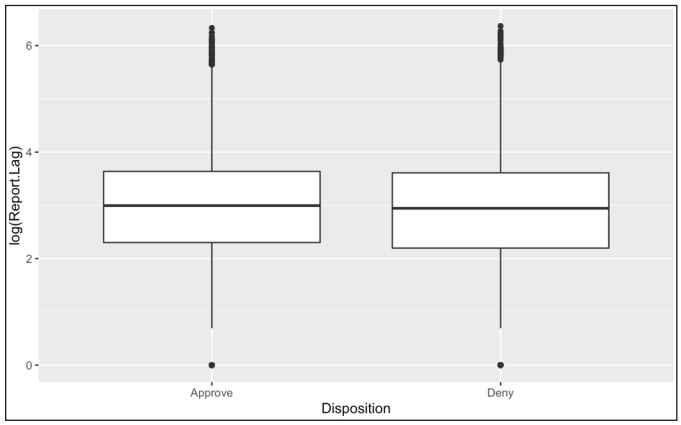


Fig. 3: The Box Plot of Predictor Report Lag for Approve Claims and Denied Claims

Then the predictor Airport.Code is explored. Its bar plot is shown in Fig. 4. Each bar on the x-axis stands for an airport. There are hundreds of different airports in the data. We need to reduce the number of attributes (levels) to improve its predictive power.

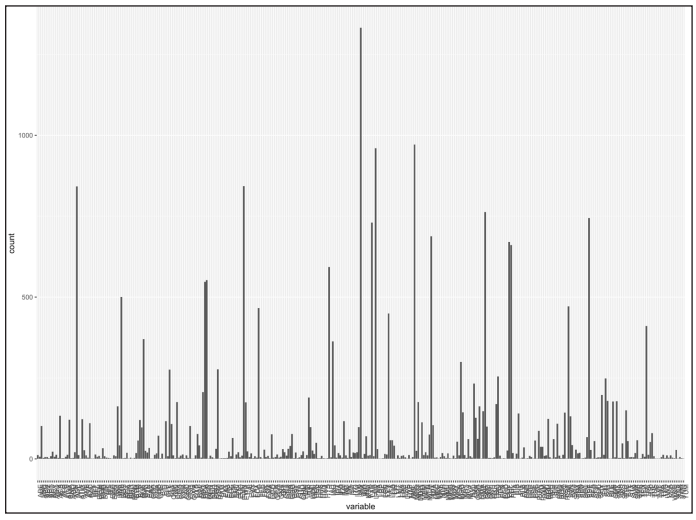


Fig. 4: The Bar Plot of Airport.Code

To reduce the levels of variable Airport.Code, we use *k*-means clustering. The inputs of the clustering are the mean and median of the target variable in each Airport.Code. We set *k*=5. Then the Airport.Code is clustered into 5 groups as shown in Fig. 5. In this way, its number of levels is reduced to 5.

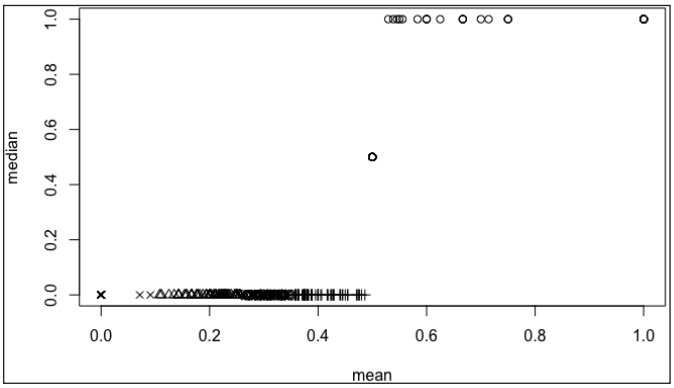


Fig. 5: The Result of k-Means (*k*=5) Clustering of Airport. Code using the Target Mean and Median at Each Airport. The Same Plot Characters (pch) belong to the Same Cluster

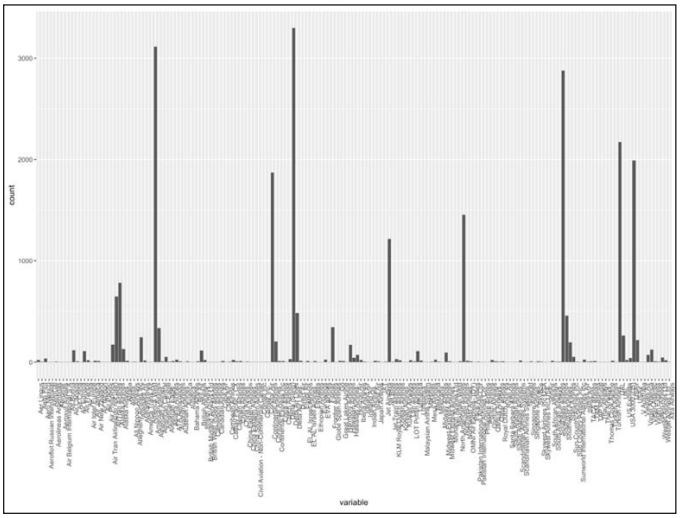


Fig. 6: The Bar Plot of the Airline.Name Shows the Number of Claims Associated with Each Airline

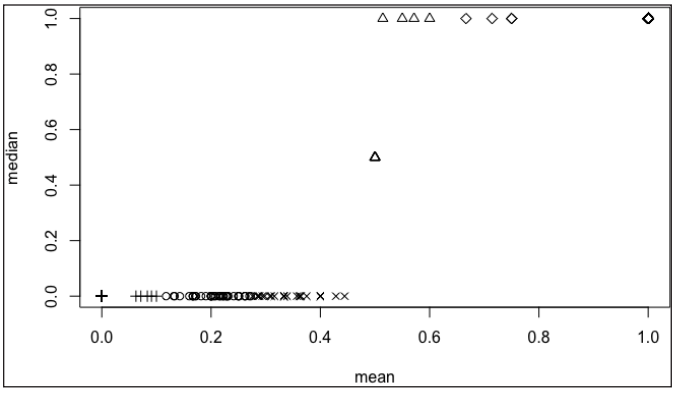


Fig. 7: The k-Means (*k*=5) Clustering Result of Predictor Airline.Name

We process the predictor Airline.Name in the same way. Fig. 6 is the bar plot of the number of claims for each airline. There are hundreds of different airlines in the date.

Similarly, k-means (k=5) clustering is applied to reduce the levels in predictor Airline.Name to 5.

There are 5 levels in the variable Claim.Type. Its bar plot (Fig. 8) below shows majority of the claims are in two levels: Passenger Property Loss, Property Damage. We consider merging the other 3 levels into one of the major levels because these 3 levels don't have enough claim counts.

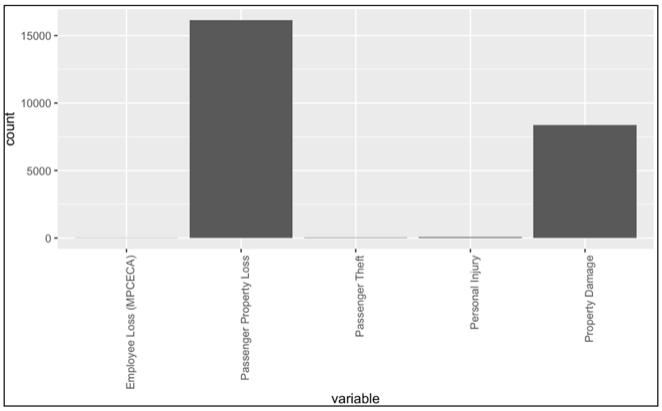


Fig. 8: The Bar Plot of Claims Counts in Each Level of Predictor Claim.Type

When looking at the proportion of approval and deny of each Claim.Type, we found there is a large difference between the two major levels (see Table II). The approve ratio is almost twice as higher in Property Damage claims (approve ratio 0.436) than in Passenger Property Loss claims (approve ratio 0.222). We merge the other 3 levels with very few claims to the 2 major levels based on their distance in approved proportion. Therefore, Passenger Theft, Person Injury are merged with Passenger Property Loss, since they also have a low approval proportion. The Employee Loss (MPCECA) is merged with Property Damage.

We set the Passenger Property Loss as the based level since it contains the most claim counts.

In the predictor Claim Site, the Checkpoint claims category (level) has a much higher proportion (approve ratio) than the other levels. Thus, we combine the level Other with the Checked Baggage because they both have low approval ratios. We set the Checked Baggage as the base level of the predictor Claim Site.

Table II: The Approved, Denied, Total Claim Counts, and Proportion of Approved Claims in Each Level of Predictor Claim Type

Claim.Type	Approve	Deny	Total	Approve Rate
Employee Loss (MPCECA)	11	7	18	0.3889
Passenger Property Loss	12569	3589	16158	0.2221
Passenger Theft	17	3	20	0.1500
Personal Injury	46	21	67	0.3134
Property Damage	4725	3650	8375	0.4358

Table III: The Approved, Denied, Total Claim Counts, and Proportion of Approved Claims in Each Level of Predictor Claim Site

Claim.Site	Approve	Deny	Total	Approve Rate
Checked Baggage	15355	4619	19974	0.2313
Checkpoint	1993	2650	4643	0.5708
Other	20	1	21	0.0476

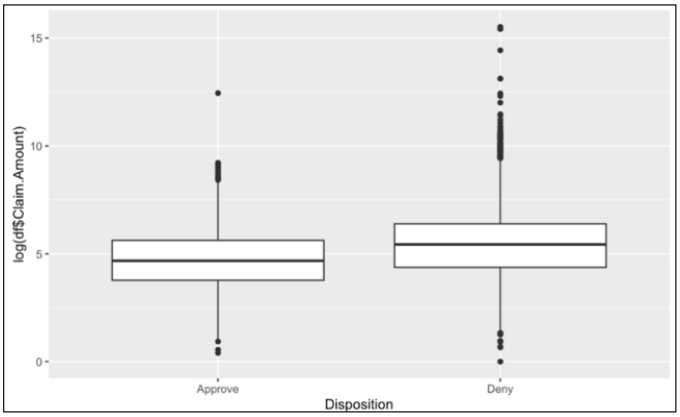


Fig. 9: The Box Plot Shows the Distribution of the Claim Amount in Approved Claims and Denied Claims

Table IV: The Approved, Denied, Total Claim Counts, and Proportion of Approved Claims in 5 Levels of Predictor Claim Site Clustered by K-Means Clustering

Item	Approve	Deny	Total	Approve Rate
I4	7110	1269	8379	0.1515
I2	2729	32	2761	0.0116
I5	7402	3331	10733	0.3104
I3	52	2563	2615	0.9801
I1	75	75	150	0.5000

Another speculation is the higher Claim Amount can easier lead to deny. To verify this, we plot the log (Claim.Amount) vs Disposition by a boxplot. The result validated our guess. From the boxplot box, the denied claims have significantly higher average claim amounts. This indicates the predictor Claim Amount has strong predictive power. There are 1873 different types of Items, including currency, cell phones, etc. It will produce poor prediction accuracy if we directly use them without combining them into a few major categories. We use k-means clustering similar to what is used on Airline. Name to combine the categories into 5 levels, as shown in Table IV. For instance, Item I4 contains 8379 records and 277 non-repeated values (levels), including Bicycles, Cameras, Clothing, etc.

III. Classification Tree

A. Single Decision Tree

We first use the trial-and-error method (Badriyah *et al.,* 2014; Shekar and Dagnew, 2019) [9] [10] to tune the single decision tree. Two parameters: maxdepth and complexity parameter (cp) (Therneau *et al.,* 2015) [11] will be selected in this process. The maxdepth defines the maximum depth of the decision tree. If the tree is too shallow, it would not be sophisticated enough to produce high accuracy. On the other hand, if the tree is too deep, it would be overfitting. So, we need to select the proper value for the maximum depth of the tree. The cp defines the minimum improvement of the model required to split a node (Myles *et al.,* 2004) [12]. A smaller cp tends to grow a tree with more branches. Again, the bias-variance tradeoff is happening here. In the following steps, we will search for maxdepth starting from 6, cp from 0.001, then try different values, see if the model is overfitting or under-fitting:

- First, at maxdepth = 6, cp = 0.001, the AUC is 0.7884 for the test, 0.7842 for the training, indicating overfitting may not be a problem.
- Then we try to reduce the complexity of the tree by setting maxdepth = 3, cp = 0.001, the AUC is 0.7834 for the test, 0.7876 for the training. The AUC almost doesn't change, but the tree is less complicated, which makes it more robust when predicting new test data.
- Then we try maxdepth = 2, cp = 0.001, the AUC is 0.6741 for the test, 0.6763 for the training. The AUC drops. Thus we will stay at maxdepth = 3.
- Adjust the parameter cp. We try maxdepth = 3, cp = 0.01, the AUC is 0.7834 for the test, 0.7876 for the training. Try maxdepth = 3, cp = 0.05, the AUC is 0.6741 for the test, 0.6763 for the training. Therefore, maxdepth = 3, cp = 0.01 is the best, since cp = 0.01 corresponds to the least complex model that has a high AUC. This model produced the following tree.

Next, we use another method: the cross-validation method to tune the parameters of the decision tree (Alawad, Zohdy and Debnath, 2018) [13]. In this method, the data is split into k folds. In each cross-validation (CV), k-1 folds are the training data, the other 1-fold is left as the validation data. Each time the validation fold is changed until all the k folds have been served as the validation data for once. The algorithm will try each value of parameter cp in the grid list to do a k fold CV. The best cp is the one with the highest average Accuracy in the k fold CV. We use the cp searching grid from 0 to 0.01 with step size 0.0005 (20 cp values) and try different maxdepth parameter values less than 6. If the tree depth is greater than 6, the tree has a high risk of overfitting. The following are the cross-validation results at different maxdepth values:

- When maxdepth = 6, AUC is 0.7842 for the training, 0.7884 for the test. The depth of the produced tree is 4, which has not reached the maxdepth.
- To reduce the model complexity, we try maxdepth = 3, its AUC is 0.7834 for the training, 0.7876 for the test. Since this tree is less complicated than maxdepth = 6 and AUC is almost the same, this tree is better.
- Then try maxdepth = 2, the AUC is 0.6741 for the training, 0.6763 for the test. Therefore, maxdepth = 3 is the best.

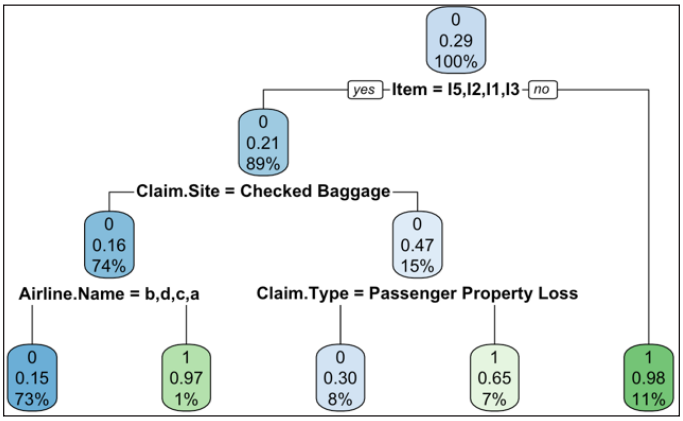


Fig. 10: The Decision Tree Whose Parameters are Tuned by the Trial-and-Error Method

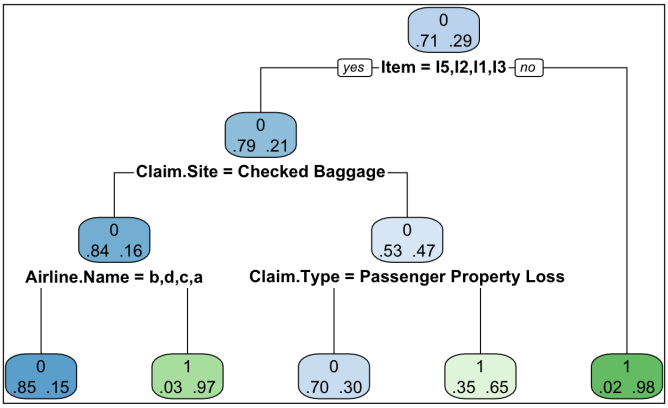


Fig. 11: The Decision Tree Whose Parameters are Tuned by the Cross-Validation Method

The best model selected by both methods is the same model. We recommend this model remain in consideration for the final model.

According to the nodes of the tree, the variables that are used in making splits in this model are:

- Item
- Claim.Site
- Airline.Name
- Claim.Type

We can interpret this tree as follows:

- If the Item is in group I4, then the model predicts the claim will be Approved.
- Otherwise, if Claim.Site is Checked Baggage and Airline. Name is in group e, then the model predicts the claim will be denied. If Airline.Name is in not in group e, then it will be approved.

- If Item is not in group 4, Claim.Site is not Checked Baggage, Claim.Type is Passenger Property Loss, then the model predicts the claim will be denied; if Claim. Type is not Passenger Property Loss, then it is predicted it will be approved.

Based on the tree, we recommend one interaction between the predictors to be used when constructing a generalized linear model (GLM) later. From the selected tree, we observe that when the value of Claim.Site is Checked Baggage, the Claim. Type doesn't matter in terms of predicting the target, while if Claim.Site is value Other, the variable Claim.Type matters. We recommend the interaction between Claim.Site and Claim.Type to be used later in GLM.

B. Boosting and Bagging Method

Boosted tree (Chen, 2014) [14] algorithm is a boosting method (Schapire, 1999) [15]. In this algorithm, a sequence of trees will be built. The latter tree is built to fit and reduce the error of the previous tree. This sequence of trees is dependent on each other. This iterative process will generate an ensemble model that is more accurate than the connected individual tree.

The random forest (Biau and Scornet, 2016) [16] algorithm is a bagging method. Unlike the dependencies of individual trees in a boosted tree method, in the bagging method such as random forest, the individual trees are independent. In this algorithm, a "forest" of trees will be generated, and the final result is the average of the result in each tree. In a random forest, each tree has an equal one "vote" to the final result. While in the boosted tree, the tree with more prediction accuracy has a higher weight in the voting for the final result. The boosted tree can reduce the bias. The random forest can reduce the variance of the prediction, thus reducing the overfitting risk.

The AUC of the boosted tree on the test data is 0.8694, which is promising.

The boosted tree algorithm also provides the variable importance, which is calculated based on how often a variable is selected to split the tree nodes (Kuhn, 2012; Greenwell, Boehmke and Gray, 2020) [17] [18]. The more often a variable is selected, the more important it is. The variable importance plot of our data according to the boosted tree is provided below.

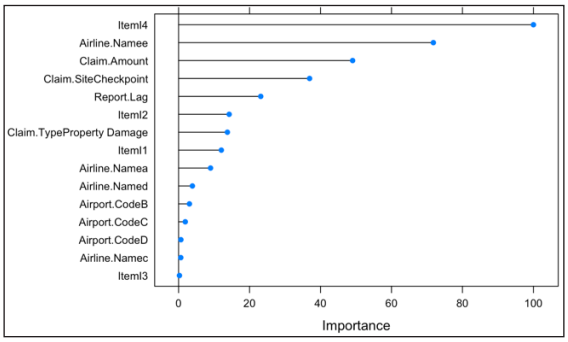


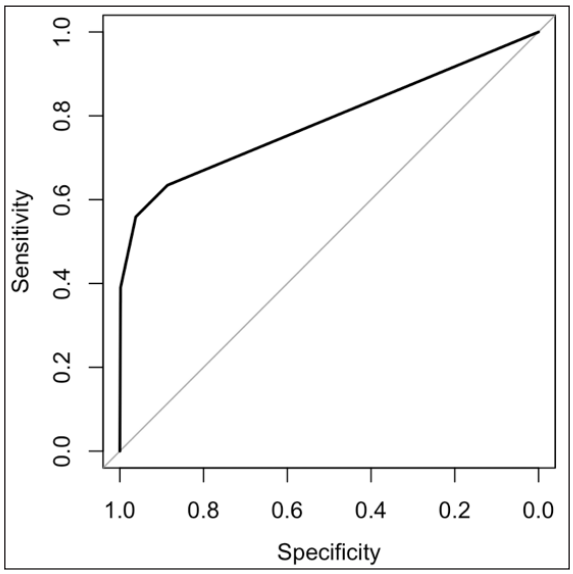
Fig. 12: The Variable Importance Plot of the Boosted Tree Algorithm

The variable importance provides more interpretability to the complicated machine learning algorithm and helps us understand what's going on in the data. In Fig. 12, the variable importance is scaled to 100, which measures the relative importance of each variable. The more important variables usually appear earlier or more frequently in the tree splitting. It is clear the variables Item = 4, Airline.Name = e, Claim.Site and Claim.Amount are more important than other variables (or levels). The majority of these variables of importance are also shown in the single decision tree we previously built. The Claim.Amount is important in this list but not shown in the single decision tree previously built. This may be due to the fact the boosting method allows the variables that are overshadowed in the single tree to be fit to the errors made by other variables.

C. Compare the Single Tree and Boosted Tree

We compare the single tree model and the boosted tree to recommend a tree model. The ROC and AUC are used as the metrics to measure the prediction accuracy. These two metrics are interpreted as follows. The tree method returns a probability value of each class before doing classification. For example, in the single tree selected in Fig. 11, the second leaf node from the right has a 0.65 probability of class 1. If the probability cut-off is 0.5, then the data in this bucket is classified as class 1. If the probability cut-off is 0.7, then they are classified as class 0. Therefore, the varying of probability cut-off changes the specificity and sensitivity. The ROC curve shows the relationship between Specificity vs Sensitivity when the cut-off probability changes from 0 to 1. The AUC is the area under the ROC curve, which measures the classification accuracy with a value ranging from 0 to 1, and larger AUC indicates better accuracy.

The AUC of the selected single tree on the test data is 0.7876. The AUC of the boosted tree on test data is 0.8694. The ROC curves are provided below, where the above figure is from the single tree, the below figure is for boosted tree.



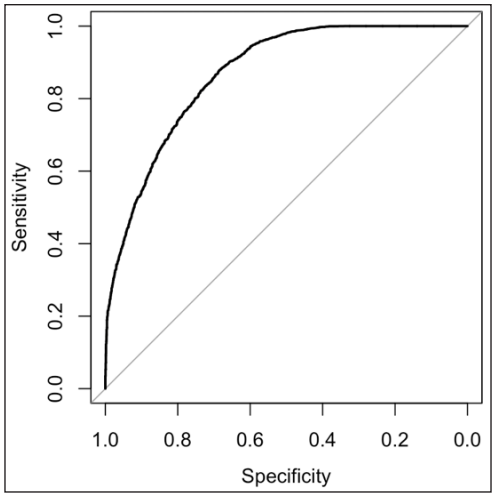


Fig. 13: The ROC Curve of the Single Tree Model (Above) and Boosted Tree Model (Below)

The advantage of a single tree is more interpretability and easier to understand, compared with the boosted tree. The splitting rule of the single tree is direct and clear to the decision-makers. The boosted tree is more of a "black box" model, where the variables are input, and the prediction is the output, the processes between are hard to explain.

In this research, we focus more on prediction accuracy rather than interpretation. The boosted tree has significantly higher AUC than the single tree algorithms considered above, therefore we recommend the boosted tree model.

IV. Generalized Linear Model

The next model we will consider is the generalized linear model (GLM) with regularization, because it can reveal more statistical structure of the data and its prediction accuracy is decent. The idea of GLM is to regress the probability that a claim will be approved using the predictors. The purpose of regularization is for variable selection: to remove the less important features while keeping the important features.

Before we do the regression, we convert Claim.Amount $500 to the level "lowAm". The level "mediumAm" is for Claim. Amount in ($500, $1500], and "highAm" are the claims with Claim.Amount > $1500.We do such discretization of the Claim. Amount is for 2 reasons:

- In linear models, $Y = \beta \cdot X$. This means there is a single constant β for all the values in one predictor to describe its constant effect on the target. Because the variable Claim. Amount is highly right-skewed as we can see from the boxplot below (Fig. 14), with mostly small values and some extremely large values. Thus, the existence of extremely large values, the effect of the lowest values would be ignored.
- By discretizing this variable, it gives us more flexibility. After binarization, the 3 levels will be treated as 3 variables. Thus, each of the levels will have a β coefficient associated with them. It can describe the different effects of this predictor at the low, medium, high level.

Therefore, discretize the continuous variable Claim.Amount to "lowAm", "mediumAm", "highAm" makes it more predictive and more interpretative than using its original numerical values. After converting, we get 18538 "lowAm" values, 3994 "mediumAm" and 2106 "highAm".

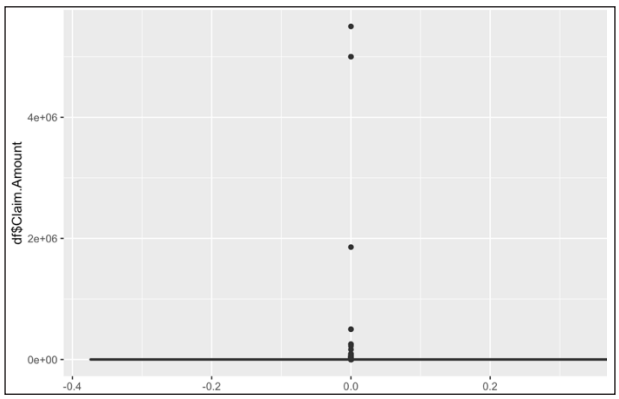


Fig. 14: The Boxplot of the Predictor Claim.Amount

In the GLM, we need to select a distribution of the target variable and the link function to map the nonlinear relationship to linear. Because the target is a binary variable, the only distribution that works here is binomial. We select the logit function as the link function, which is the default link for binomial GLM. We run three types of regularization with GLM, they are LASSO (Zou, 2006) [19], ridge (Segerstedt, 1992) [20], and Elastic Net (Zou and Hastie, 2005; Hastie *et al.*, 2021) [21] [22]. When running the model on the test data, we got AUC = 0.8575 for LASSO, 0.8694 for the ridge, 0.8682 for Elastic Net (alpha = 0.5). There is not much difference in AUC, but LASSO has the potential advantage of removing more features to build a simple model. So, we choose LASSO. Table V listed the GLM coefficients using LASSO.

4 variables have 0 coefficient and the variable Airport.CodeD has a small coefficient, so we can remove it. All the other variables are selected including the interaction term (in the last row of Table V).

The four variables selected in the session 3.1 single decision tree are Item, Claim.Site, Airline.Name, Claim.Type. They are all selected in the LASSO GLM here, and three extra variables are included: Airport, Claim.Amount, the interaction term between Claim.Type and Claim.Site. The variable Item is important in the single decision tree, which also have large coefficients here in the LASSO GLM. This confirms the correctness of both models. Because of the limited depth of the tree, it's no surprise that the single decision tree didn't include the variables like Airpot, Claim.Amount.

The boosted tree in session 3.2 listed ItemI4 as the most important feature, which confirms the largest coefficients of ItemI4 in the LASSO GLM. Other than ItemI4, the boosted tree selected Airline.Name, Claim.Site, Claim.Type, Claim.Amount as top important features and they are also selected as important

features in LASSO GLM. The Airport is not listed as the top important feature in the boosted tree, while in the LASSO GLM, it is more important than the Claim.Type.

The LASSO GLM requires binarization of the categorical variable. It treats each level of the categorical variable as a separate variable. For instance, variable Item has 5 levels: I1, I2, … I5. In LASOO GLM, the variable Item is binarized into 4 variables ItemI1, …, ItemI4. If they have 0 values, then it refers to the 5th level (the level with the least frequency in the data). Each variable corresponds to one level (or category) of the original variable. This process is also called making dummy variables. Doing so, there are advantages and disadvantages. The advantages of using binarized variables:

- It gives us more flexibility. We would like to have the opportunity to keep some levels of one variable while removing the other levels. Without binarization, the algorithm will remove or keep all levels entirely.
- Similarly, different levels of the same variable could have different effectson the target variable. If we treat it as one, this difference could not be resented. The binarization allows us to attach different regression coefficients to them.

The disadvantage of using binarized variables:

- It increases the computational burden. Because each level is treated as one variable after binarization, it could take more time to run the model.

Table V: The Regression Coefficients using GLM with Lasso

(Intercept)	.
Report.Lag	.
Airport.CodeE	.
Airport.CodeC	*0.1570*
Airport.CodeB	*-0.1066*
Airport.CodeD	*-0.0203*
Airline.Named	.
Airline.Namee	*2.1463*
Airline.Namec	.
Airline.Namea	*-0.0591*
Claim.TypeProperty Damage	*0.1067*
Claim.SiteCheckpoint	*0.5818*
ItemI2	*-0.9589*
ItemI1	*0.2795*
ItemI4	*2.3836*
ItemI3	*0.1097*
Claim.Amount_cutmediumAm	*-0.1529*
Claim.Amount_cuthighAm	*-0.1934*
Claim. TypePropertyDamage:Claim. SiteCheckpoint	*0.9846*

V. Final Model Selection

Now we select a final model based on the models we have built: single decision tree, boosted tree, and LASSO GLM. We recommend LASSO GLM as the final model. Because it is easier to interpret with straight forward statistical meaning with its regression coefficients. Its AUC of 0.8575 is also decent, higher than the single decision tree AUC of 0.7876, though slighter lower than boosted tree AUC of 0.8694. Also, the GLM can be implemented in a simple tool such as an Excel spreadsheet, while it is difficult to do so for the boosted tree.

The LASSO GLM by default, classifies the regressed probability greater than 0.5 as one class, lower than 0.5 as another class. That is, the cutoff probability is 0.5. The selection of cutoff values changes the classification results. The cutoff adjusts the tradeoff between the false positive and false negative. In practice, the profit/loss could be different in making each type of prediction in true positive (TP), true negative (TN), false positive (FP), false negative (FN). If our goal is to maximize the overall profit, we can optimize the cutoff probability. We will demonstrate how to do it on this data. Without losing generality, assuming for each TP prediction, we get a profit of $50. The actual profit loss number should be determined by the business. If it is FP prediction, we get a loss of $25. For the TN or FN prediction, we get a loss of $5. Therefore, the total profit is 50*TP-25*FP-5*(TN+FN). If the cutoff changes, the confusion matrix will change, resulting in total profit change, because the values of TP, FP, TN, FN will change with cutoff probability. We tried different cutoff values and calculated their profitsas listed in Table VI. We found out when cutoff = 0.23, the total profit can be maximized. Table VII lists the confusion matrix at cutoff = 0.23.

To demonstrate how to use the selected final model, we explain a demo. Table VIII contains 1 base case (the 1st row) and 7 other variation cases (the other 7 rows) from the base case. To do a prediction, we need 7 pieces of information in the input, including the Report.Lag, Aiport.Code, Claim.Amount etc. We use them to predict whether the TSA will deny or approve this case. The model used is the LASSO GLM with the cutoff = 0.23 to maximize the total profit. The algorithm returns the probability of approval, and the probability above 0.23 is classified as Approve, otherwise classified as Deny. The probability and prediction results are listed in the last two columns of the table. The probability is calculated using the coefficients in Table V to multiply the 7 variables in the input (after necessary preprocessing including binarization). This can be implemented in a spreadsheet tool that is easy for the related industry to use.

In the second row of Table VIII, the case has one variation from the base case, which is changing from the Report.Lag from 17 to 1. The classification probability doesn't change. This is because the coefficient is 0 for variable Report.Lag in the GLM with LASSO features selection in Table V. So, the value of Report. Lag doesn't matter. As another example, the coefficients of Claim.Site = Checkpoint is 0.5818. When the value of Claim.

Site changes from "Checked Baggage" to "Checkpoint", the regression coefficient is changed from 0 to 9.5818, resulting in the classification probability being increased from 0.1741 to 0.2738.

Table VI: The Overall Profit Benefiting from Our Algorithm by Selecting the Different Cutoff Probability of the Lasso Glm

Cutoff	Profit
0.20	25480
0.23	32195 (Best)
0.24	31480
0.25	30810
0.26	29765
0.30	29810
0.50	26850
0.60	14290

Table VII: Confusion Matrix at Cutoff = 0.23

		Reference	
		Deny	Approve
Prediction	Deny	4329	632
	Approve	860	1570

Table VIII: A Demo to Explain How to Use the Selected Model

Report.Lag	Airport. Code	Airline. Name	Claim.Type	Claim.Site	Item	Claim. Amount	Probability	Prediction
17	MDW	Delta Air Lines	Passenger Property Loss	Checked Baggage	Clothing - Shoes; belts; accessories; etc.	65	0.1741	Deny
1	MDW	Delta Air Lines	Passenger Property Loss	Checked Baggage	Clothing - Shoes; belts; accessories; etc.	65	0.1741	Deny
17	ABQ	Delta Air Lines	Passenger Property Loss	Checked Baggage	Clothing - Shoes; belts; accessories; etc.	65	0.1741	Deny
17	MDW	Air Canada	Passenger Property Loss	Checked Baggage	Clothing - Shoes; belts; accessories; etc.	65	0.1741	Deny
17	MDW	Delta Air Lines	Property Damage	Checked Baggage	Clothing - Shoes; belts; accessories; etc.	65	0.1899	Deny
17	MDW	Delta Air Lines	Passenger Property Loss	Checkpoint	Clothing - Shoes; belts; accessories; etc.	65	0.2738	Approve
17	MDW	Delta Air Lines	Passenger Property Loss	Checked Baggage	Locks	65	0.0747	Deny
17	MDW	Delta Air Lines	Passenger Property Loss	Checked Baggage	Clothing - Shoes; belts; accessories; etc.	3000	0.1480	Deny

VI. Summary and Discussion

In this paper, we used the TSA claim data to construct a model that will accurately predict if a claim will be denied or approved by TSA. We considered and compared different models including single decision tree, boosted tree, GLM with regulations including LASSO, Ridge regression, and Elastic Net. The raw data set has 7 columns of predictors and one column of the target variable, which is Deny or Approve. The predictor columns include the Report.Lag, Airport.Code,

Airline.Name, Claim.Amount etc. There are a total of 24628 cases of which 7270 are Approve cases and 17368 Deny cases. The Approve/Deny is replaced with a 0/1 target variable called value_flag. We examined each predictor variable on its own and with respect to value_flag. We tried 2 methods to construct and tune a single classification tree: the trial-and-error method, the cross-validation method. Both methods result in the same decision tree. The AUC of this tree is 0.78. Then a boosted tree is constructed. The AUC of the boosted tree for the test data is 0.8694. Then we considered the binomial Generalized Linear Model (GLM) with logit link function using regularization. The three types of regulation methods tested are LASSO, Ridge regression, Elastic Net. The three regulations are not much different in AUC (0.86~0.87), but LASSO has the potential advantage of removing more features to build a simple model. So we choose LASSO GLM. This is also the final model selected. Because it is easier to interpret and has a comparable AUC with the boosted tree.

We also demonstrated how to optimize the cutoff probability in the LASSO GLM to maximize the overall profit of the predictions. A spreadsheet demo is given to explain how the selected model can be easily used by the related industry professionals.

This research can be used by insurance companies to develop products such as luggage damage insurance, travel insurance. For instance, an actuary would use this research to first predict the probability that a luggage damage claim will be approved by TSA, then the denied claims will be covered by the insurance policy. Knowing this probability helps to price the proper premium rate. The passengers can also benefit from this research. An APP can be developed from our research to predict whether a claim made by the passenger will approve or not. Knowing this ahead of time helps the travelers decide how much time and effort they should spend on making the claim and following up on the claim status. The TSA can also benefit from this research to better manage their work. We know there is a delay from the time a claim is reported to the claim approval/deny decision. If the TSA use our research to early estimate the future amount of approved claim based on the current reported claims information, that helps the TSA plan ahead financially for how much money they need to prepare for the claims to be paid. This is beneficial for stable operating.

References

[1] United States, Congress (107th, 1st session: 2001), (2001). Aviation and Transportation Security Act: Conference Report (to accompany S. 1447). [Washington, D.C.] [U.S. Government Printing Office].

[2] K.-A. Lowe, "Safety in the sky: Will reforming and restructuring the TSA improve our security or merrily infringe on our rights," *J. Air L. & Com.*, vol. 81, p. 291, 2016.

[3] U. S. Department of Homeland Security, "TSA Claims Data," Nov. 2021. [Online]. Available: https://www.dhs.gov/tsa-claims-data

[4] M. Kelly, and Z. Wang, "A data set for modeling claims processes—TSA claims data," *Risk Management and Insurance Review*, vol. 23, no. 3, pp. 269-276, 2020.

[5] R. Ciullo, "JetBlue v. Delta: Battle of the Baggage," Poster Presented at University of Bridgeport Faculty Research Day, Bridgeport, CT, May 24, 2017.

[6] A.-R. Correia, and S.-C. Wirasinghe, "Level of service analysis for airport baggage claim with a case study of the Calgary International Airport," *Journal of Advanced Transportation*, vol. 4, no. 2, pp. 103-112, 2010.

[7] M.-R. Franks, "Airline liability for loss, damage, or delay of passenger baggage," *Fordham J. Corp. & Fin. L.,* vol. 12, p. 735, 2007.

[8] V. Kyseľová, "Risk management in air transport and insurance," *MEST Journal*, vol. 1, no. 1, pp. 113-125, 2013.

[9] T. Badriyah, J. S. Briggs, P. Meredith, S. W. Jarvis, P. E. Schmidt, ... and G. B. Smith, "Decision-tree early warning score (DTEWS) validates the design of the national early warning score (NEWS)," *Resuscitation*, vol. 85, no. 3, pp. 418-423, 2014.

[10] B.-H. Shekar,and G. Dagnew, "Grid search-based hyperparameter tuning and classification of microarray cancer data," *2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP),* IEEE, Feb. 2019, pp. 1-8.

[11] T. Therneau, B. Atkinson, B. Ripley, and M.-B. Ripley, "Package 'rpart'," 2015. [Online] Available: http://cran.ma.ic.ac.uk/web/packages/rpart/rpart.pdf

[12] A.-J. Myles, R.-N. Feudale, Y. Liu, N.-A. Woody, and S.-D. Brown, "An introduction to decision tree modeling," *Journal of Chemometrics: A Journal of the Chemometrics Society*, vol. 18, no. 6, pp. 275-285, 2004.

[13] W. Alawad, M. Zohdy, and D. Debnath, "Tuning hyperparameters of decision tree classifiers using computationally efficient schemes," in *2018 IEEE First International Conference on Artificial Intelligence and Knowledge Engineering (AIKE),* Sep. 2018, pp. 168-169.

[14] T. Chen, "Introduction to boosted trees," [PowerPoint slides], 2014. [Online]. Available: https://web.njit.edu/~usman/courses/cs675_fall16/BoostedTree.pdf

[15] R.-E. Schapire, "A brief introduction to boosting," in *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, vol. 99, pp. 1401-1406, 1999.

[16] G. Biau, and E. Scornet, "A random forest guided tour," *Test*, vol. 25, no. 2, pp. 197-227, 2016.

[17] M. Kuhn, "Variable importance using the caret package," *Journal of Statistical Software*, vol. 6, 2012.

[18] B.-M. Greenwell, B.-C. Boehmke, and B. Gray, "Variable importance plots - An introduction to the VIP package," *R Journal,* vol. 12, no. 1, p. 343, 2020.

[19] H. Zou, "The adaptive lasso and its oracle properties," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1418-1429, 2006.

[20] B. Segerstedt, "On ordinary ridge regression in generalized linear models," *Communications in Statistics-Theory and Methods*, vol. 21, no. 8, pp. 2227-2246, 1992.

[21] H. Zou, and T. Hasti, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 67, no. 2, pp. 301-320, 2005.

[22] J.-K. Tay, B. Narasimhan, and T. Hastie, "Elastic net regularization paths for all generalized linear models," 2021, arXiv:2103.03475.