Project Task 2 - BAN404

March 27, 2019

INTRODUCTION

The objective of this project is to develop a model that accurately predicts if a household within the San Francisco Bay area belongs to a high-income group. The annual income of a household should be equal to $50,000 or higher to fit this criterion. For this task we have a dataset called *marketing* which is an extract of 14 variables and 8,993 observations from a survey conducted in 1987 to a total of 9,409 people. The survey was part of a Market Basket Analysis which included 502 questions. The variables in our dataset are mainly characteristics of the inhabitants.

DATA EXPLORATION

MISSING VALUES. Our first step is to check the number of missing values. About 2% of the total observations (2,694 observations out of 125,902) correspond to missing values. To investigate those problematic predictors, we use the function *sapply()* to count the number of missing values in each column. We sort and round it to three digits to make it more convenient. The marketing dataset is pretty high quality as datasets go. As Table 1 shows, 9 out of 15 variables have missing values but the missing observations represent less than 5% of the observations within a variable. This exludes the "Lived" predictor which missing information increases up to 10%. To deal with this inconvenient, first we make a new version of the dataset in case we ever change our mind after implementing some changes. By removing missing values, we lose 24% of the total observations meaning that we end up with a dataset of 6,876 observations.

Table 1: Percentage of missing value in each variable

Income	Sex	Age	Dual_Income	Householdu18	Home_Type
0.00%	0.00%	0.00%	0.00%	0.00%	4.00%
Ethnic	Edu	Occupation	Marital	Status	Language
0.80%	1.00%	1.50%	1.80%	2.70%	4.00%
Household	Lived				
4.02%	10.2%				

Then, we transform our predictors to factors. A factor is how R deals with categorical variables. In adittion we create our response variable; high==1 if the household income equals $50,000 or more. This criterion corresponds to the two upper levels of income out of nine in total. We get that only 26% of the sample is part of the high-income group.

DESCRIPTIVE STATISTICS

Before running usual correlation and summary tables we should remind that these methods do not capture correctly the relationships within categorical variables. For this reason we use Goodman and Kruskal's tau measure of association between categorical variables. In contrast to other popular measures for categorical variables as the chi-square and Cramer's V, the tau statistic is asymmetric. This means that the method can be used to identify cases where one variable is highly predictive from another, but the reverse implication is not true. This method attempts to quantify the variability in a specific feature that can be explained by variations in a source variable (Pearson, 2016). In Figure 1 the variable Income explains some variability in the other variables, but for this exercise we are interested in the reverse associations

(first column). Ranging from zero association we find the predictors Sex, Lived and Language and the highest values correspond to Status (0.07), Occupation(0.08) and Age(0.09).

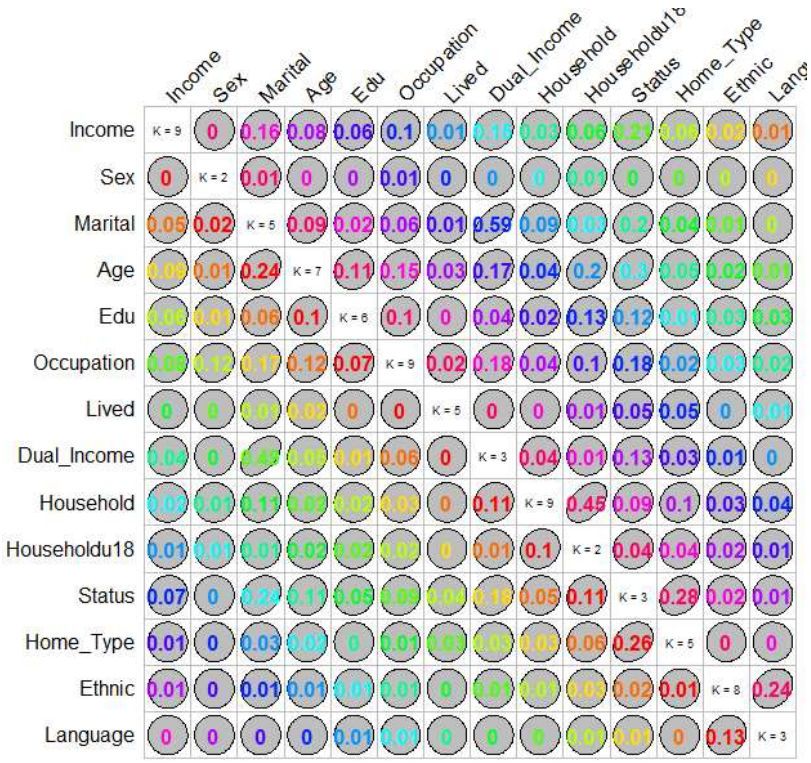

Figure 1: Goodman-Kruskal tau matrix for the marketing dataframe

Table 2: Average level of annual income for households

Occupation	Income	Age(years)	Income	Education	Income
Professional/managerial	6.611659	14-17	1.675425	Grade 8 or less	1.880682
Sales worker	4.690438	18-24	3.785093	Grade 8 to 11	2.479034
Factory/Laborer/Driver	4.540678	25-34	5.405543	High school grad.	4.521298
Clerical/Service worker	4.681055	35-44	6.483031	College (1-3 years)	5.228085
Homemaker	5.746032	45-54	6.741110	College grad.	6.219553
Student, HS or College	2.630319	55-64	6.264344	Grad. study	6.924390
Military	4.492857	+=65	5.071247		
Retired	5.274590				
Unemployed	3.181818				

Marital	Income	Dual Income	Income	Status	Income
Married	6.680995	Not married	4.011181	Own	6.787539
Living together	5.091418	Yes	6.845580	Rent	4.503123
Divorced/sep.	4.869242	No	6.187255	Living w/parents	2.993617
Widowed	4.242574				
Single	3.610736				

For features that show a slight ability to explain variations in the annual income, we describe the average level of annual income in Table 2. Notice that income here is not measured in a continuous variable as thousands of dollars, it registers nine levels or categories for annual earnings being 1 the lowest - Less than $10,000- and 9 the highest - $75,000 or more-. We call the attention to the fact that students are the

occupation group who earn the least on average ($10,000 to $14,999), their household income ranks after unemployed ($15,000 to $19,999). Household income peaks when its head members correspond to an age group between 45 to 54 years old. Income increases smoothly along with years of education. People who report living with their parents/family has on average lower income compared to people who rent or possess their own real state. On average, households with a married couple earn more than households where the partners are only living together. The difference is one level of income; married couples earn on average $30,000 to $39,999 per year compared to cohabitants who earn on average $25,000 to $29,999 per year. For married couples having dual income they do not earn, on average, a higher level of income compared to households without dual income (both cases are mostly classified in the sixth level). Although there is a slightly higher trend for dual income households to earn more.

We have a closer look to the annual income distribution. The higher concentration of households is grouped within the first category of income meaning that 18% of the sample earn less than $10,000 per year. The second group with highest concentration of households (earning $50,000 annually) represents 16% of the sample size. Figure 2 shows the Lorenz curve for income distribution and the perfect-equality is given by the 45-degree line. From the Lorenz curve we observe that in our sample 50% of the population earns a little more than 20% of the total income. The ratio of the area between the line and the curve gives the Gini coefficient which in this case is equal to 0.31. A larger Gini coefficient represents more income inequality.

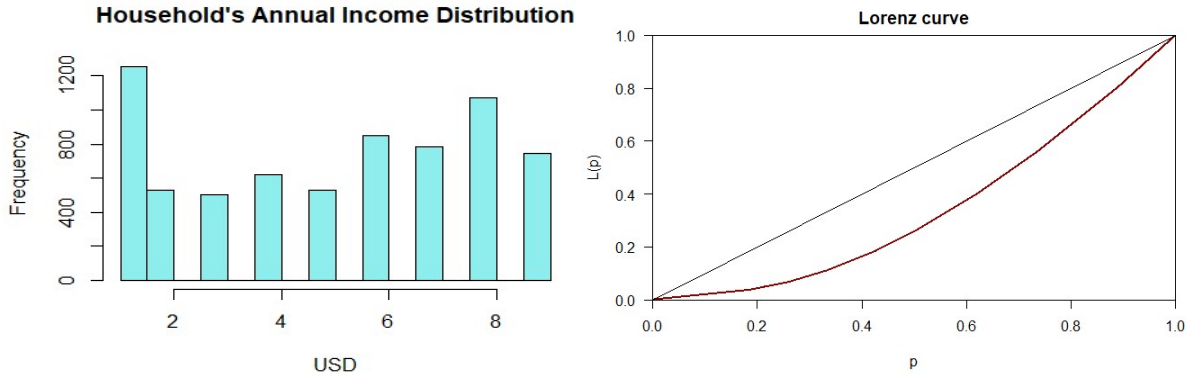

Figure 2: Graphical representations of the distribution of income

PRELIMINARY PROCESSING OF DATA

We combined the levels of the variable "Household". As the observations in some levels are too rare compared to those in major levels, and such levels can be treated as outliers. In addition, we made the variable "Householdu18" as a dummy variable, that is, according to whether within a household lives a person under 18 years old (Householdu18 ==1, zero otherwise). The reason why we do this is to avoid too little variation in some minor levels, which may cause error when we conduct some methods in R, especially in LDA part, and that will be explained in detail in LDA part.

PREDICTIVE MODELS

Before explaining the models we employ for predicting a household with high income, we would like to explain that for cross-validation we use K-folds where K=10. Unlike the first project where we only had 177 observations, now we have a larger dataset. Even if Leave-One-Out Cross-Validation(LOOCV) provides less estimation uncertainty relatively to k-fold cross-validation or the validation set method. Computational power can suppose a limitation for this case. We will discuss more about this topic when we explain the (Supervised Vector Machines) SVM approach on our data. Regarding the selection for the number of folds, if K = N, the cross-validation estimator would be approximately unbiased for the true

(expected) prediction error but can have high variance because close similarities among the N training sets, besides the computational burden would be also considerable. Breiman and Spector (1992); Kohavi (1995) have explore in more depth this trade off and they agree to recommend five or tenfold CV (Hastie, Tibshirani, & Friedman, 2009). We evaluate the accuracy of prediction by calculating the rate of misclassification, that is:

$$error = \frac{(predicted\ 0s\ that\ actually\ are\ 1s) + (predicted\ 1s\ that\ actually\ are\ 0s)}{num\ of\ observations\ in\ test\ data\ set} \quad (1)$$

LOGISTIC REGRESSION

Logistic regression is aimed at modelling the relationship between predictors and the probability of certain event occurring. It chooses parameters by the maximum likelihood method, selecting the one that makes the predicted probability of high income as closely as possible to the actual likelihood of high income happening. If we use standard linear regression, the probability could be greater than one or less than zero. Logistic regression overcomes this problem, it guarantees the probability to range between 0 to 1. After some algebra of the logistic regression, Equation 2 shows there is a linear relationship between log-odds and predictors. Because we just have two categories in the response variable, there would be only one linear relationship in our model, indicating it would be simple and easy to implement (Hastie, Tibshirani, & Friedman, 2009).

$$\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 x \quad (2)$$

First, we use *glm()* function to run a logistic regression. Then we use *summary()* to check the output, applying the Wald Test to decide the relevance of predictors. The null hypothesis of the Wald Test is H_0: $\beta_1 = 0$. When the Z score of a predictor is less than 2 we cannot reject the null hypothesis, indicating that the response variable is not dependent on this predictor.

Because 74% of the observations in our dataset belong to low-income category and 26% fall into the high-income group, we do not consider appropriate to just use the threshold of 0.5 to classify an observation with the predict *function()*. We use the ROC curve to decide the most suitable threshold. The receiver operating characteristic curve, abbreviated as ROC, is graphed by predicting and calculating the errors for both classes. Here we use 10-fold cross validation to calculate True Positive Rate and False Positive Rate for different thresholds. We observe in Figure 3 that we choose 0.7 as threshold since compared to 0.6, it improves True Positive Rate without increasing substantially the False Positive Rate. Compared to 0.8, it reduces substantially the False Positive Rate without affecting the True Positive Rate too much.

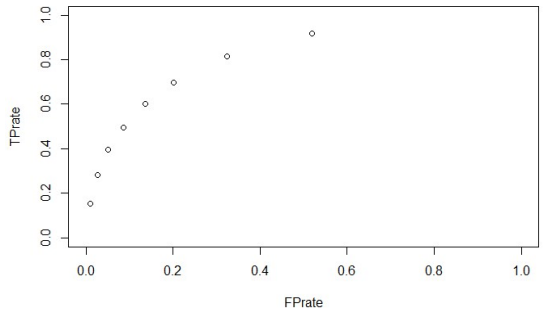

Figure 3: ROC curve of logistic regression

Next, we make predictions based on our coefficient estimation. Positive coefficients indicate a positive relationship between predictors and the response variable. Negative coefficients mean the probability of getting high level of income decreases when a household present these features. For instance, an estimated coefficient -0.33 for Female shows that compared to males with the same characteristics, a female interviewee is significantly less likely to be part of a high-income household. Being female leads to a decrease in the odds of high-income level of exp(-0.33)-1= 0.29 or 29%. Incorporating the standard error, we get an approximate 95% confidence interval of exp(-0.33+-2*0.07)-1= (0.17,0.37). When a customer, who is male, divorced, 46 years old, college graduate or factory worker, have dual income, his household is formed by 3 people, none of them under 18 and lives in a rented apartment, the probability of his household being classified as high-income is 0.054. But for a female who shares the same conditions, her probability of being assigned to the high-income group decreases to 0.038.

To sum up the relevant predictor of the logistic regression, a household tends to be classified as high-income if the interviewed is over 18 years old, has assisted to college or has graduated studies. In addition, if the person's household receive dual income and if the household is formed by two to seven people. At the same time is less likely to classify a household to the high-income group if the survey was filled by a single, divorced or widowed female that does not work in a managerial or professional position, whose household is conformed by people under 18, and pays rent instead of owing the real state and that lives in an apartment or mobile home. The logistic model provides more predictors against high-income households rather than providing features that raises the probability of classifying as high-income, this might be explained by the fact that more than 70% of our observations does not belong to the high-income group.

However, the significance levels that provides the Wald Test are not always reliable though it's easy to implement. When data produces large estimates, the St. Errors gets inflated, making the Z scores larger. As a result, some significant predictors can be dismissed (Bewick, Cheek, & Ball, 2005). As suggested by James, Witten, Hastie, & Tibshiran (2013) to improve the test error in the logistic regression we should apply first variable selection methods. The best subset selection method can also overcome the problem of Wald Test but it is time consuming and the variance would be high. Thus, we consider L1 Regularized Logistic Regression. L1 penalty helps us make a trade-off between variance and bias. For logistic regression, we need to find a combination of coefficient estimates that maximizes the quantity.

$$\max_{\beta_0,\beta}\left\{\sum_{i=1}^{N}\left[y_i(\beta_0+\beta^T x_i)-\log\left(1+e^{\beta_0+\beta^T x_i}\right)-\lambda\sum_{j=1}^{p}|\beta_j|\right]\right\} \tag{3}$$

$\lambda\sum_{j=1}^{p}|\beta_j|$, the second term in Equation 3, is the shrinkage penalty, which decreases to zero as $\beta_1,\beta_2,\ldots,\beta_p$ approaches zero. Lambda is tuning parameter which controls to what extent Eq. 2 would have influence on coefficient estimation. When $\lambda = 0$, there is no influence and we can run a standard logistic regression. As λ increases, we must shrink the coefficient estimates towards or exactly to zero. Each λ has an optimal model, whose coefficients are estimated under its corresponding λ value. We use *cv.glmnet()* to calculate the cross-validation error of models for different λ and select the one with lowest cross-validation error. Then we apply this best λ to our data. According to the ROC curve (Figure 4), we choose 0.6 as the classification threshold.

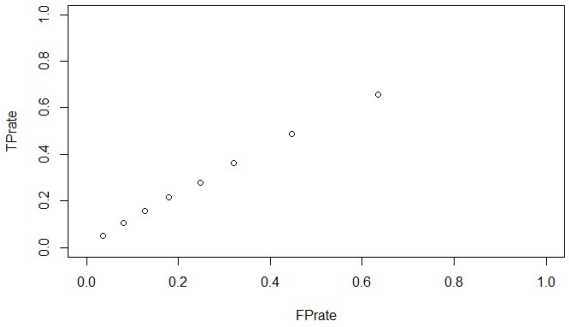

Figure 4: ROC curve for L1 Regularized Logistic Regression

The cross-validation error shows that L1 Regularized Logistic Regression outperforms standard logistic regression. It indicates that most high-income households consist of more than two white people who are between 35 to 64 years old, married or living together, living in a house or condominium which is owned by the householder and speaks English. In terms of education, the interviewee should have graduate studies or have assisted to college and work as professional or managerial.

LINEAR DISCRIMINAT ANALYSIS (LDA)

LDA is a supervised classification method of qualitative variables in which two or more groups are known a priori and new observations are classified in one of them according to their characteristics. Using the Bayes theorem, LDA estimates the probability that an observation, given a value for its predictors, belongs to a class of the response variable, P(Y=k | X=x). Finally, the observation is assigned to the class k for which the predicted probability is greater. It is an alternative to logistic regression when the qualitative variable has more than two levels. To classify a new observation, one must have a prior probability (πk) that the random observation belongs to class k. We define fk(X)≡P(X=x|Y=k) as the conditional probability density function of X for an observation belonging to class k. The posterior probability refers to the observation that belongs to the class k, where x is the value of the predictor P(Y=k|X=x). Applying the Bayes theorem, we can know the probability for each class:

$$P(belong\ to\ class\ k|observed\ x) = \frac{P(belong\ to\ class\ k\ and\ observe\ x)}{P(observe\ x)}$$

$$P(Y=k|X=x) = \frac{\pi_k P(X=x|Y=k)}{\sum_{j=1}^{K} \pi_j P(X=x|Y=j)} = \frac{\pi_k f_k(x)}{\sum_{j=1}^{K} \pi_j f_j(x)} \quad (4)$$

We get the classification with the lowest error by assigning an observation to the group that maximizes the posterior probability. For the Bayes classification, we need to know the population probability that any observation belongs to each class (πk) and the population probability that an observation that belongs to class k takes the value x in its predictor (fk(X)≡P(X=x|Y=k)). In practice, this information is not available, so we estimate those parameters from the sample.

According to the help document one caveat of LDA is that the function tries hard to detect if the within-class covariance matrix is singular. If any variable has a small variance within-group, it will stop and report the variable as constant. This happens to us, R could not solve the inverse matrix because the within-class covariance matrix was singular. There was a problem with the first two levels of Householdu18 which offers too little variation. Therefore, we did the level-combination as we discussed before about processing our data. Then, we run the regression and get that LDA and logistic regression

predictions are very similar. As shown in Table 3, LDA model produce predictions which are accurate about 80% of the time, even though half of the observations were not used to fit the model.

Table 3: Confusion table for LDA

	Predicted Group	
Actual Group	FALSE	TRUR
FALSE	2271	406
TRUE	263	497
Mean Accuracy	0.8054	
Mean Test Error	0.1946	

The LDA output indicates the prior probabilities of groups. Like mentioned before, 26.4% of the training observations are high-income households. It also provides the group means but due to extension limits we only show an extract of the whole outcome. These values correspond to the average of each predictor within each class. For example, we observe that there is a tendency for individuals between ages 45-54 (i.e. Age5) to be part of a high-income household and subjects with education from grade 9 up to grade 11 (Edu2) not to belong to a high-income household. The next section of coefficients of linear discriminants, provides the linear combination of all predictors that are used to form the LDA decision rule. Here, we present an extract of the 5 highest coefficients which can be understood as the most relevant predictors for the LDA model as those are the variables with a greater impact on defining the classification.

If the product of each variable and coefficient are large enough (e.g. 0.98×Edu6-1.09×Occupation7-0.97Occupation8+1.17xHousehold8-1.17xStatus […]), then the LDA classifier will predict high-income classification for an observation, and if it is smaller than the threshold, then the LDA classifier will predict a low-income classification. For the LDA model, predictors with the highest association with "high" are subjects with grad studies as their educational background, plus they are military or have retired. Other relevant characteristics are households with 8 members that rent their place to live. Overall the role of these variables is in line with our expectations. Table 4 demonstrates the results of LDA regression.

Table 4: LDA regression results

	Age3	Age4	Age5	Age6	Age7	Edu2
FALSE	0.2578	0.1483	0.0692	0.0589	0.0720	0.1534
TRUE	0.2277	0.3058	0.1760	0.0957	0.0484	0.0385
	Edu6	Occupation7	Occupation8	Household8	Status2	
Coefficient	0.9808	-1.0929	-0.9707	1.1696	-1.1749	

The ROC curve shows two types of errors for all possible thresholds, it is constructed by plotting Sensitivity (y-axis) versus specificity (x-axis). Sensitivity refers to subjects "positivity in disease", those cases where high-income households were correctly identified, using a given threshold value. The false positive rate is 1-specificity: the fraction of low-income households classified incorrectly as having high income, using that same threshold value. The ideal ROC curve "hugs the top left corner, indicating a high true positive rate and a low false positive rate" (James, Witten, Hastie, & Tibshiran, 2013). The 45-degree line represents the "no information" classifier which we would expect if the predictors were not related to the probability of being a high-income household. The performance of the classifier summarized over all possible thresholds is given by the area under the ROC curve, AUC (Figure 5). The larger the area, the better the classifier. The AUC in this case is 0.83, which is not far from the maximum of one. We expect a classifier evaluated on an independent test set to perform better than chance (AUC = 0.5).

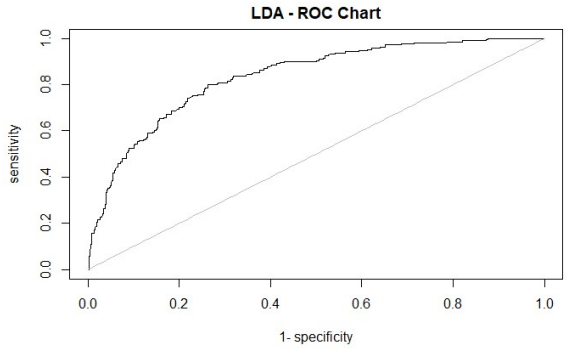

Figure 5: ROC curve of LDA

A different way of visualizing the prediction accuracy, often used in linear discriminant analysis, are confusion tables. The confusion matrix shows how many observations were correctly or incorrectly classified. The diagonal elements of the confusion matrix indicate correct predictions, while the off-diagonals represent incorrect predictions. Hence our model correctly predicted that 497 households earn more than $50,000 per year and that 2,271 households earn less than this threshold, giving a total of 2,271 + 497 = 2,768 correct predictions. Then we compute the fraction of households for which the prediction was correct. In this case, LDA correctly classifies a household 80% of the times. The confusion matrix also suggests that for households earning less than $50,000 annually, the model predicts correctly their income group 84% of the time but for households earning at least $50,000, it has only a 65% accuracy rate. The test error rate is almost 20% which is better than random guessing.

CLASSIFICATION TREES

The tree-based method partitions our predictors into a set of simple regions. The predicted value of an observation equals to the mode or mean of the training observations that belong to the same region as this predicted observation. Tree-based method is easy to implement, interpret and visualize.

First, we implement the *tree()* function on our data. Figure 6 shows, the top predictor classifies the household in which people live in rented houses or live with relatives to the left branch. If true, the households will be immediately discarded from the high-income group. Observations that do not fit the criteria of the initial node will follow the right branch and then they will be filtered by occupation. If a member of the household who reply the survey has a professional or managerial job, the household would be classify as high-income only if they are married or lives together with his/her partner.

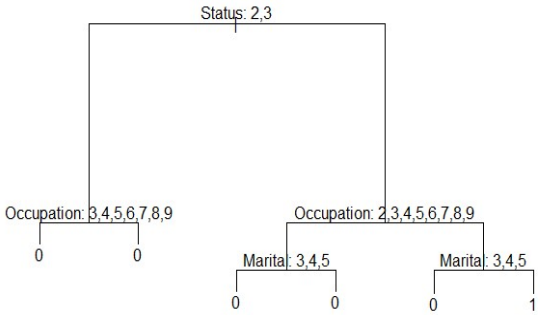

Figure 6: Basic Decision Tree

Although a basic tree can capture information in training observations more accurately, it may have high variance and produce overfitting. Hence in order to reduce variance, we prune the tree at the expense of increasing bias. Instead of exhausting all possible subtrees, which is time-consuming, we use the weakest

link pruning to do this job. The principle behind this method is shown in Equation 5 . The first part is the squared error of a subtree fitting training observations and the second part works as a penalty on the number of terminal nodes for the subtree. When $\alpha = 0$, there is no penalty so we can choose a complete tree. As α increases, selecting a subtree with more terminal nodes can fit training data more accurately, it will increase the value of penalty. Thus we will select instead a subtree with less terminal nodes that minimizes the value of this formula. In other words, the penalty parameter α helps us make a trade-off between variance and bias.

$$\sum_{m=1}^{|T|} \sum_{i:x_i \epsilon R_m} \left(y_i - \hat{y}_{R_m}\right)^2 + \alpha|T| \tag{5}$$

First, we use *tree()* function to create a complete tree and then implement the weakest linking pruning method to find optimal subtree corresponds to each α value. We employ *cv.tree()* function to calculate the cross-validation error of subtrees produced as a function of different α values. Next we show the relationship between the number of terminal nodes and its cross-validation error In Figure 7. Because number of terminal nodes is a function of different α and cross-validation error is a function of each α value, we can draw a graph to show the relationship between number of terminal nodes and cross-validation error, which is more convenient than to display the relationship between α and cross-validation error. Figure 7 suggests when the number of terminal nodes is 4 the lowest cross-validation error occurs. It is important to mention that the *cv.tree()* consider the training data that we specified when fitting the tree model.

The pruned tree is shown in Figure 8. Householder status (renting or living with relatives) is still a key factor to determine the income level. If in a household this does not happen and if a member works as professional or in a managerial job, is married or is living with a partner, the household will then be classified as high-income. The cross-validation error for the pruned tree and the basic tree are virtually the same, but the display of the pruned tree is simpler. Thus, due to parsimony we prefer the prune tree over its basic version.

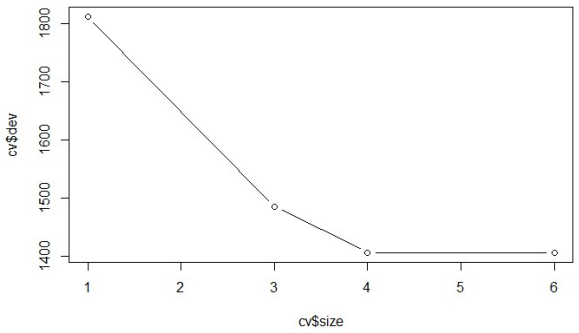

Figure 7: Cross-Validation error as a function of the number of terminal nodes

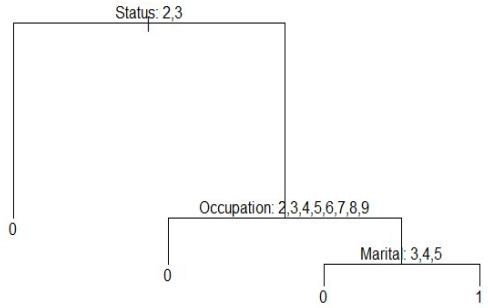

Figure 8: Pruned tree

Then we use the *randomForest()* function to apply the bagging procedure to reduce the variance in our pruned tree model. We set mtry=13 which refers to the number of variables randomly sampled as candidates at each split (Liaw, 2018). The tree we made before suffer a problem that a small change in data may result in a change in the pattern of the tree. In other words, our previous tree has high variance and may perform badly in prediction. Bagging takes repeatedly a sample from the training observations, and for each sample, it models a new tree. Ultimately for the observation we want to predict, we use all complete trees to predict its value. The procedure consists on taking the average predictions of all models. Given the limitations of computing means in a qualitive dataset, we choose the most frequent values produced by all the trees as the predicted value.

However, the improvement in prediction power is at the cost of reducing interpretability. We can't find a single tree to clearly describe the statistical procedure because we use a collection of bagged trees to get the final prediction. But we can use *varImpPlot()* to check the importance of each variable in predicting a target feature. As Figure 9 shows, Status is most crucial and Occupation on second place, the same as what we get in the basic tree and in the prune tree. Nevertheless, education level becomes more important and marital situation matters less.

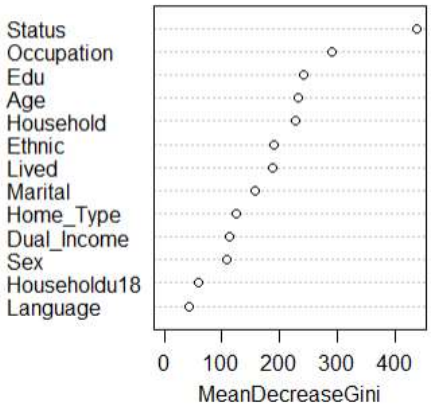

Figure 9: Variable importance plot

Next, we use *randomForest()* again to make random forests. Like bagging, it bootstraps training data to build a set of decision trees. But it doesn't randomly sample all predictors at each split, which is an advantage over bagging method. Because when one of our predictors is very strong, each single tree in bagging process would has the same top split, resulting in trees that look very alike, and predictions would be highly correlated to each other. Bagging cannot overcome the overfitting problem in this situation since averaging highly correlated predictions do not reduce variance substantially. Typically, in classification problems we choose square root of predictors as the number of sample candidate (James, Witten, Hastie, & Tibshiran, 2013). Thus, we set *mtry* argument in *randomForest()* as 4.

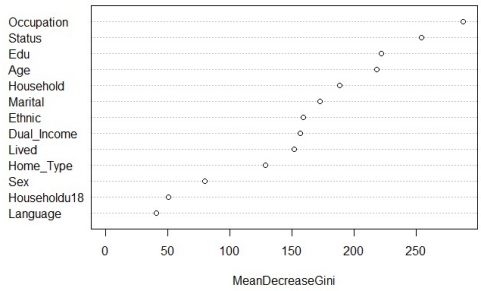

Figure 10: Variable importance plot

For random forests, Figure 10 shows occupation is the key variable in influencing income level. Lower cross-validation error of random forest indicates it outperforms bagging in prediction. This may because the strong predictor, Status, influences the performance of bagging. Thus, among all tree-based method, in terms of accuracy in prediction, random forest is the best one and in terms of interpretability, pruned tree better.

Support Vector Machines (SVM)

The second additional method we choose to address our task is Support Vector Machines, abbreviated as "SVM". Firstly, we naturally consider Maximal Margin Hyperplane, which determines a separating hyperplane that is farthest from the training observations. However, maximal margin hyperplane cannot deal with non-separable cases, so the method of Support Vector Classifiers or also known as soft margin classifier was developed. It allows some observations to be on the incorrect side of the margin or even the wrong side of hyperplane. Hence, this method tolerates bias to some extent for lower variance. The extent of tolerance of such bias is controlled by a coefficient cost(C). C controls the bias-variance trade-off of support vector classifier. Larger C means more support vectors, wider margin, and higher bias but low variance. In contrast, smaller C means less support vectors, narrower margin, and high variance but low bias.

However, both Maximal Margin Hyperplane and Support Vector Classifiers make classification with linear boundary, but in practice we often face non-linear boundaries. For this reason, the method of Support Vector Machines is introduced, as it can solve the problem of non-linear boundaries by enlarging the feature space using non-linear functions K referred in many cases as a kernel. A kernel is a function that quantifies the similarity between two observations. There are three commonly used kernels: linear, polynomial and radial.

Linear: $$K(x_i, x_{i'j}) = \sum_{j=1}^{p} x_{ij} x_{i'j} \tag{6}$$

Linear kernel can generate a support vector classifier as it creates linear boundary.

Polynomial: $$K(x_i, x_{i'}) = \left(1 + \sum_{j=1}^{p} x_{ij} x_{i'j}\right)^d \tag{7}$$

Polynomial kernel includes a parameter d, which is the degree of polynomial function used, which can generate a more flexible classification boundary. Specifically, when d=1 the SVM is identical to a support vector classifier.

Radial: $$K(x_i, x_{i'}) = \exp\left(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2\right) \tag{8}$$

Radial kernel includes a positive constant γ. Using radial, only the training observations that are close to boundary have an effect on classification of test observations.

The reason of using a kernel instead of using the function of original features to enlarge feature space is because kernel is computational. However, the computation is also a problem for some kernels, such as radial.We adopt the method of SVM, and we tried out three common function arguments, and we also conduct a 10-fold cross validation to calculate the test error, and the motivation of using this cross-validation method is stated in previous context. Here we made a customized function to fit a SVM model, where a range of gamma and cost (C) is provided for fitting a best model.

The 10-fold cross validation results are shown in Table 5. The mean errors of radial, linear and polynomial SVM are 0.1868, 0.1943 and 0.1902 respectively. Therefore, we believe that based on our data, radial SVM performs slightly better than the other two methods from a prediction perspective, and the linear kernel performs the worst. Hence, adopting a non-linear kernel function to fit a SVM model can better classify the observations in our data into two classes of income level.

Table 5: Errors and key coefficients of SVM

	radial SVM					linear SVM				polynomial SVM				
	γ	cost	nsv	error		cost	nsv	error		cost	d	nsv	error
1	0.01	100	2675	0.2111	1	0.1	2768	0.2169	1	0.001	3	2994	0.2067
2	0.01	10	2812	0.1645	2	0.1	2837	0.1703	2	0.001	3	3034	0.1499
3	0.01	10	2820	0.1616	3	100	2759	0.1761	3	0.001	3	3036	0.1645
4	0.01	10	2764	0.1936	4	10	2715	0.2082	4	0.001	3	3000	0.2023
5	0.01	10	2807	0.1776	5	0.1	2829	0.1834	5	1	3	3038	0.1907
6	0.01	10	2759	0.2096	6	1	2700	0.2154	6	0.001	3	2988	0.2082
7	0.01	100	2732	0.1689	7	10	2745	0.1834	7	0.001	3	3020	0.1790
8	0.01	10	2795	0.1776	8	0.1	2805	0.1892	8	0.001	3	3020	0.1805
9	0.01	10	2784	0.1849	9	10	2726	0.1921	9	0.001	3	3025	0.1907
10	0.01	10	2738	0.2183	10	100	2704	0.2082	10	0.001	3	2969	0.2300

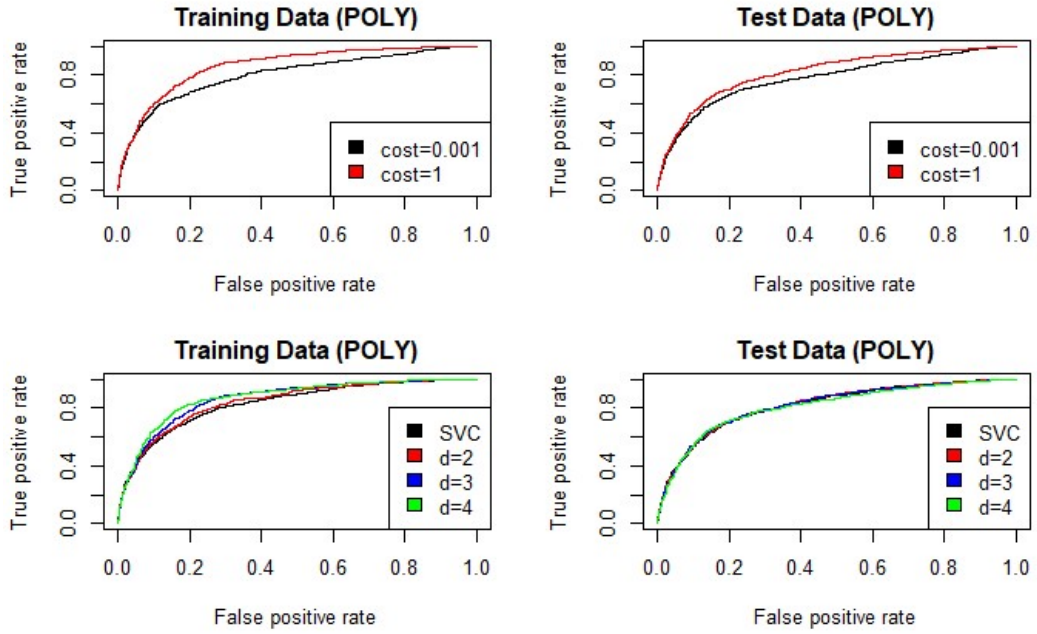

Figure 11: ROC Curve of Polynomial kernel

To better compare the performance of different SVM models with different coefficients on our training and test data, we provide visual representation of the results through the *ROC curve*. The choice of cost is based on the data in Table 5.

Regarding ROC curves for polynomial kernels (in Figure 11), we observe that setting C as 1 performs better than 0.001 both in training and test data. Then, we fix C to 1 and change the degree (d). The higher the degree, the better the fitting in training data, however, functions with different degrees seems to perform identical in the test data.

As shown in Figure 12, the ROC curves for radial kernels show that a higher cost of 100 performs better than 10 in the training data, however, in test data, a cost of 10 outperforms C=100. Therefore, a more flexible model with larger C can always perform better in fitting the training data, but not necessarily have a better prediction power on test data. Next, we fix C=10 fixed and compare different values for gamma. We can see that γ=1 fits perfectly the training data, however, performs worst in test data (overfitting), while γ=0.01 performs best in from an out-of-sample perspective.

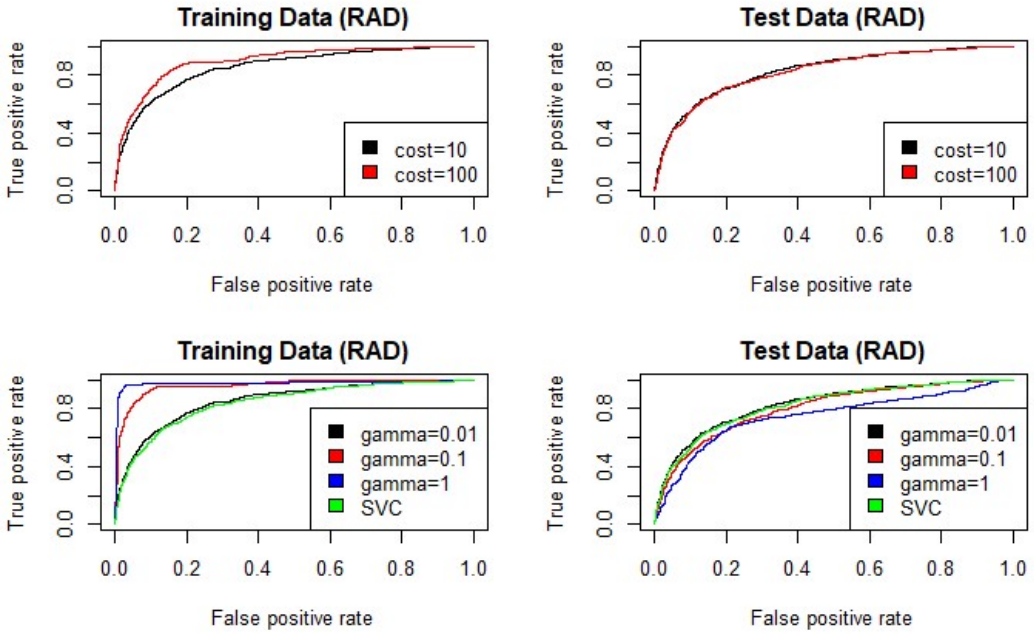

Figure 12: ROC Curve of Radial kernel

To conclude SVM analysis, based on our data, we would recommend fitting a SVM model using a radial function, with γ=0.01 and C=10. The accuracy of predictions using SVM is definitely satisfactory, however, it is difficult to interpret, because the determination of the classification boundary is too technical to explain and interpret using kernel functions rather than functions on the original data features. Just as previously discussed, SVM may have a computational disadvantage, especially for radial kernel. It took us a considerable amount of time to loop over all candidate γs and Cs for each method, using 10-fold cross-validation. Moreover, tune() function conducts a 10-fold cross-validation by default in the training set to fit a best model, however, best fitting in training set not necessarily leads to a best fit in test data, and that is the reason why we chose a 10-fold cross-validation on the whole data, to pick the best model on each fold, calculate the prediction test error on each fold, and then calculate the mean error.

CONCLUSION

To predict the two-dimensional qualitative variable "high", which represents the level of annual income a household earns, we tried five classification prediction methods, including logistic regression, linear discriminant analysis, classification tree with pruning, random forests including bagging procedure and support vector machines. We use 10-fold cross-validation for each method to calculate the test error of prediction. From Table 6 we can say that the method of SVM has the smallest test error rate. From a prediction perspective, using SVM is most accurate based on our data. The remaining four methods perform similarly being LDA the second-best option and the logistic regression the one least accurate in predicting.

Table 6: Summary of errors of different methods

10-fold cross-validation test errors	
Logistic regression	0.2067
Linear discriminant analysis	0.1987
Classification tree with pruning	0.2029
Random forest	0.1992
Support vector machines	0.1868

Nevertheless, from the five methods proposed, only logistic regression and classification tree with pruning can be easily interpreted. According to logistic regression, a household earning an annual income equal or higher than $50,000 most likely possess the following features: more than 2 white adults, living in San Francisco Bay area for 1 to 6 years, living in own house or living with families, English speaker, graduated from college, working professional or managerial jobs, age between 35 to 64 years old, married or living with partner. The features of high-income households suggested by pruned tree classification are, with members working as professional or managerial, married or living with partner, living in own house. However, using linear discriminant analysis, in linear regression there are some variables with significant coefficients, which play a more important role in prediction, like educational background, whether members are military or have retired, whether the household has over 8 members. However, the prediction is based on a linear combination of all variables, and thus cannot be interpreted very specifically by only few key variables. For the method of random forests, it sacrifices the interpretability for better prediction, as it uses a collection of bagged trees to get prediction results, therefore we cannot extract interpretations from individual key predictors.

Table 7: Some characteristics of different learning methods

green= good,yellow=fair, and red=poor					
Characteristic	Logistic regression	Linear discriminant analysis	Classification tree with pruning	Random forest	Support vector machines
Robustness of outliers	▼	▼	▲	▲	▼
Computational scalability	▲	▲	▲	◆	▼
Ability to deal with irrelevant inputs	▼	◆	▲	▲	▼
Ability to extract linear combination of features	▲	▲	▼	▼	▲
Interpretability	▲	◆	▲	▼	▼
Prediction power	▼	◆	▼	▲	▲

Based on our working experience on this project, we also summarize in Table 7 the performance of each method in different characteristics, including robustness of outliers, computational scalability, ability to deal with irrelevant inputs, ability to extract linear combination of features, interpretability and prediction power. From the Table 7, all the methods we tried have both pros and cons, so we should wisely select a method based on our aim. If our aim is to predict precisely, we should choose methods with strongest prediction power, like random forest or support vector machines. If our aim is to clearly and specifically interpret the prediction features of high-income households, we should instead choose methods with superior interpretability, like logistic regression or classification tree with pruning. Also, if our data is high-dimensional, a method that can reduce dimension will be a good choice, just as what we did with the logistic model; we used lasso to shrink the dimension in our data. In addition, if we have a data set with large N, we should also take computational scalability into consideration, and choose a more computational feasible method, for instance, logistic regression, LDA/QDA or classification tree with pruning.

Finally, we exclude from our analysis unsupervised learning methods as we have information regarding the household income. If that were not the case, we would have implemented clustering techniques on our data but for this study we did not consider appropriate to discard valuable information of income. In addition, if we were interested in reduction of dimensions that PCA offers, we can also achieve that through LDA/QDA models. Therefore, our additional methods to address the classification problem are SVM and the extensions for the tree model with random forest application including bagging.

REFERENCES

Bewick, V., Cheek, L., & Ball, J. (2005). Statistics review 14: Logistic regression. *Critical care, 9(1)*, 112–118.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning.* Standford: Springer.

James, G., Witten, D., Hastie, T., & Tibshiran, R. (2013). *An Introduction to Statistical Learning.* Los Angeles: Springer.

Liaw, A. (2018, March 22). *The Comprehensive R Archive Network*. Retrieved from Classification And Regression With Random Forest: https://www.rdocumentation.org/packages/randomForest/versions/4.6-14/topics/randomForest

Pearson, R. (2016, 04 12). *The GoodmanKruskal package: Measuring association between categorical variables*. Retrieved from The Comprehensive R Archive Network: https://cran.r-project.org/web/packages/GoodmanKruskal/vignettes/GoodmanKruskal.html