|  |  |
| --- | --- |
| For taking only some columns from a data frame df, say variables x and z, but with z first, we write: | df[c("z","x")] |
| To save df we enter | write.csv(df,"df.csv") |
| Box and whisker plot useful for | getting an idea of the range of values of a variable |
| To create a box and whisker plot for x in R | boxplot(x) |
| \*If we have a factor variable f also in the data set and want to create a boxplot for x broken up by the different factors | boxplot(x~f) |
| Histogram useful for | showing the distribution of a variable |
| If we want to compute max of a variable x but at least one of its slots is NA, then it will output NA. How can we work around this? What else needs this? | max(x, na.rm=TRUE)  mean(x,na.rm=TRUE)  median(x,na.rm=TRUE) |
| In script file how to modify a line to communicate to R it is a comment and not a command when we highlight and ctrl+r everything: | Put # in front |
| How do we look at the current directory being used? | getwd() |
| Note about a similarity between variables and data frames | can also use subset and cbind on variables |
| If we want to know the variable names of a data frame df | colnames(df) |
| If we know an observation value p of variable x but not its slot number use | match(p,x) |
| Limit the range of variable x in a plot to a<=x<=b | xlim=c(a,b) |
| \*To break the x-axis in a plot into p parts use | breaks=p |
| TRUE/FALSE called | logicals |
| To convert TRUE/FALSE to 1/0 | as.numeric |
| When we use tapply(arg1, arg2, arg3) what does the output look like? | There is a column whose headings are the values of arg2, and the value underneath each heading is arg3 applied to arg1 restricted to the observations where that value of arg2 applies. |
| \*If we have a data frame df with a factor variable f but there are some factor levels which are not being used by any of the observations, we can get rid of this how? | df$f = factor(df$f) |
| How do we check the above and what happens? | By looking at str(df); the factor variable now has less levels and the numbers associated with each factor are now adjusted so that they start back at 1. |
| If we have a string (chr or factor variable called s) for date and want to convert it to a date format that R recognizes: | If s = "m/d/yyyy" then as.Date(s, "%m/%d/%Y")  If s = "d/yyyy/m" then as.Date(s, "%d/%Y/%m")  etc. Notice that the Y is capital in as.Date |
| The date format that R recognizes | "yyyy-mm-dd" |
| See p.26 to see why Y is capitalized above but lowercase sometimes. |  |
| Another way dates may be written in strings and how we handle this when using as.Date | instead of / we have - |
| How are dates in the as.Date format ordered? | Earlier dates come first |
| For a plot in R how are the data points plotted by default? | As open circles |
| \*What if we want to instead join the points by a line? | Add the argument type = "l" |
| What if we want to add another line to our plot using new variables x and y? | Add a second command lines(x,y) |
| What happens to the plot when we do this? | The axis names/scales of the plot don't change; so it displays the line only in the region of the plane shown by the original plot. |
| Example of something we can add to an existing plot that isn't a line? | points(x,y) |
| How to obtain the 3rd through 6th observations of a variable x: | x[3:6] |
| For a data frame df how do we obtain the observations in rows 3 to 5 and columns 4 to 8? | df[3:5,4:8] |
| What if we want all rows but just columns 4 to 8? | df[,4:8] or just df[4:8] |
| What if we want all columns but just rows 3 to 5? | df[3:5,] |
| When a variable is in "yyyy-mm-dd" format what does months() do? What does it require to work? | It outputs the month as a word, surrounded by quotes. The variable needs to have been converted using as.Date; this would not work on a chr or factor variable before conversion. |
| What does is.na(x) do? | It outputs TRUE or FALSE for each of the slots of x, TRUE for the NA values and FALSE for the others. |
| \*If we have a variable x which outputs TRUE and FALSE for each slot and we want to calculate the proportion of true: | mean(x) |
| Same as above but if we want to calculate total number of TRUE: | sum(x) |
| How do we use jitter? | If x and y are variables we can do for example jitter(x) or jitter(c(x,y)). The second outputs two numbers (no spaces or quotes or anything). |
| What does ls() output? | All user-defined objects (vectors, matrices, data frames, lists, functions) |
| For linear regression model y = B0 + B1x what are B0 and B1 called? | The intercept coefficient and regression coefficient. |
| SSE, what is it? | Sum of squared errors; the sum of the squares of the differences between true outcome values and outcome values predicted by a linear regression model. |
| 2 weaknesses of SSE | It scales proportionally to the number of data points and its unit are non-interpretable. |
| Solution to above? | Calculate root mean squared error instead (RMSE) = sqrt(SSE/n); square residuals, then sum, then divide by number of residuals, then sqrt. |
| Baseline model for linear regression | Predicts the mean outcome value |
| What is SST? | The SSE for the baseline model, called the total sum of squares. |
| How is R­­2 calculated? | 1-SSE/SST, where SSE is the minimal SSE possible (the best model possible) |
| What does adding more variables do to R­2? | It increases but with diminishing returns |
| \*What happens if we add too many variables to our model? | We risk overfitting, which means adding too many variables relative to the amount of data and so the model won't extrapolate well to unseen data despite having a high R2. |
| What does R2 do? | It captures model accuracy on the training set. |
| What does adjusted R2 do? | It is similar to R2 but takes into account how many variables are used to build the model (model complexity). |
| When is adjusted R2 decreased? | When we add an independent variable that doesn't help the model. |
| What do we want between R2 and adjusted R2? | Both to be high |
| How to obtain the residuals of a linear model called lm: | lm$residuals |
| What arguments can cor() take? | Two variables or a data frame whose variables are all numerical for example. |
| Multicollinearity means | Two variables are highly correlated, which we don't want between independent variables. |
| How do we deal with multicollinearity? | Between pairs of highly correlated variables we can try reproducing the model removing one of the variables at a time. Which should be removed can be judged on the context. |
| Examples of when we might need to remove a variable? | When its coefficient is positive but we would think the variable should be positively correlated with the dependent variable, or vice versa. When two highly correlated variables have high significance codes but one of them probably has no relation to the outcome variable. When its significance code is low. If we don't know where to start we can start by removing the ones with the highest values of Pr(|t|). |
| Approximately how large does the correlation between two independent variables have to be before there is cause for concern? | Nothing definitive, but as a rule of thumb, say 0.7. |
| \*The accuracy of a model on a test set is called | Out-of-sample accuracy (double check that this is indeed accuracy; I thought accuracy was for categorical outcomes) |
| \*How to manually compute out-of-sample accuracy of a model on a test set? | R2 = 1-SSE/SST where to compute SSE we use the line from the model compared to the values from the test set, and for SST we use the mean from the training data to compare to the values from the test set. |
| What else should be taken into account when out-of-sample accuracy has been calculated? | How many test points was it computed on? If not a lot then it is not very reliable. |
| When selecting our model what do we want? | Both high R2 and high test set R2 |
| If we can remove an independent variable from a model without changing the R2 values much, what does this mean? | We have a simpler and therefore better model. |
| Summarize how the Moneyball example works | * The goal is to make it to the playoffs * By looking at historical data it was realized that if a team wins 95 of the 162 season games they almost always make it to the playoffs, so this is what we aim for * It was noticed that W can be predicted by the difference (RD) in runs scored (RS) and runs allowed (RA) throughout the entire season by creating a linear regression model and seeing there is a high R2. Inserting W=95 into this equation we find RD is 133.5 will predict 95 wins. * To predict RS we look at the baseball data frame restricted to before 2002 (our historical data) and see there are 3 relevant stats, OBP, SLG and BA, so we create a model but find that the coefficient for BA is negative, which is counterintuitive, so we remove this stat and find that OBP and SLG alone do a good job. * To predict RA we do similarly as above (actually it doesn't include OBA in the data set). * If we want to make future predictions using the models then the teams change, so we cannot use past values for OBP, SLG, OOBP and OSLG. Instead we can estimate these by averaging the OBP and SLG of each player on a team from the previous season and the OOBP and OSLG of each pitcher on a team from the previous season. * Therefore a team can be built by choose averages of OBP, SLG, OOBP and OSLG appropriately. |
| How we can quickly check if there is a linear relationship between a dependent variable and an independent variable? | plot them or look at R2 |
| Difference between subset of a data frame df and for example df[c("x","y")] | The first takes a subset of the rows (observations) whereas the second takes a subset of the columns (variables). |
| \*For a linear regression model lm what does steps(lm) do? | It automates the process of trying out different combinations of independent variables to try to predict the outcome variable and outputs the model it judges to be most appropriate, based on the number of independent variables and R2 value. |
| A weakness I see with steps(lm): | It doesn't factor in context. For example it won't realize that a given coefficient being negative might not make sense in the given application. |
| If we have a data frame df which has NA values but we don't want them: | df = na.omit(df) |
| What exactly does this do? | It removes all observations where at least one variable has NA. |
| What is AIC? | Akaike Information Criterion, which formalizes the tradeoff between having more independent variables (which increases model complexity and threatens over-fitting) and a higher R2. |
| What happens when we build a linear regression model where one of the independent variables is a factor, call it f with levels A, B, C, D. | Suppose A is the reference level for f in its data frame. Then the model will have a variable for each of the other levels, called fB, fC, fD and there will be no variable called f. |
| What are their values? | They are binary. When talking about a specific level that isn't the reference then its variable (e.g. fC) is 1 and the others are 0. If talking about the reference level then all of them are 0. |
| \*How can we change the reference level of a factor variable? | df$f = relevel(df$f, ref="C") where df is the data frame where f is found and C is the new reference level. It seems we don't actually need to put ref= |
| When we have data that is skewed left or skewed right what is a good idea? | To trying graph the dependent variable versus the log of the independent variable instead. |
| What can we use that is identical to match(p,x)? | which(x==p) |
| Which package do we use for time series models? | zoo |
| What is a time series model? | One in which consecutive weekly (or some unit of time) measurements of variables are taken. |
| What is a categorical variable? | Takes a limited number of possible values (like binary 0/1 or FALSE/TRUE), usually qualitative. |
| What is the other possibility? | Numerical (quantitative) |
| What do we need for logistic regression to have a chance of being appropriate? | A binary outcome variable. |
| Where does the definition of logit come from? | In a logistic regression we start with the probability of the outcome being 1 and define P(y=1) = 1/1+e^-(logit) where logit = log(odds) and odds=P(y=1)/P(y=0). |
| What is the relationship between a logistic regression model with a binary outcome variable predicting 1 and its logit? | Bigger logit -> larger probability for outcome=1.  Smaller logit -> larger probability for outcome=0. |
| How do we determine a baseline model for a categorical situation? | By taking the mode outcome. |
| If we know the independent variables in a logistic regression are non-negative, then what does the sign of a coefficient say about its variable? | If negative then it tells us that increasing its variable is predictive of increasing the probability of outcome 0 and if positive then increasing its variable is predictive of increasing the probability of outcome 1. |
| What is a problem called when its outcome variable is categorical? | classification |
| Syntax of sample.split | If splitting data frame df and want it to be balanced over variable x with 70% of the data in the training set then we have split=sample.split(df$x, SplitRatio=0.7). It is more useful when x is categorical I think. |
| When we have subset(df, arg) what does this do? | arg needs to be a logical and this outputs a data frame that consists of the subset of all observations of df where arg is TRUE. |
| What might we want to do before splitting? | set.seed() |
| What else does split = sample.split() do? | If using a SplitRatio of 0.7 for example then the training set will have 70% of the data and the test set will have 30% of the data. However if we input split we see that it doesn't merely label the first 70% data points as TRUE and the rest as FALSE. Instead it makes it so that the proportions of the variable values are the same in each of the test and training sets as they are in the original |
| Logistic regression models don't have R2; what do they have instead? | AIC, comparable to adjusted R2, compares accuracy of model on training set relative to model complexity (number of independent variables used). |
| Interpretation of it? | Lower is better; however, only comparable amongst models on the same data set. |
| Output of a logistic regression model is: | A probability |
| When we have a logistic regression model m and we use predict(), what is the output? | If we have predict(m, type="response") then the output consists of the probability of the outcome variable being 1 for each observation in the data set that defined m. If we want to use this model to make predictions on a new set then we add the argument newdata=test for example. (It is probably 1 rather than 0 because our convention where logit was defined was in terms of the outcome variable being 1, namely P(y=1) = 1/(1+e^-(logit)).  If we omit type="response" then the output is the values of the logit for each observation. |
| What is a threshold value? | We often want to make a binary prediction based on the probability of the outcome being 1 - if this probability is larger than the threshold then we predict 1, if not then we predict 0. |
| How do we choose t? | Depends on what we want to prioritize. If we want to capture more TP's then we set a low threshold for example. |
| When do we choose t=0.5 | When there is no preference between the two outcomes. |
| What is 1-specificity called in a confusion matrix? | false-positive rate |
| How can we choose a good threshold value? | ROC curve |
| What goes along x-axis? | 1-specificity = FPR |
| Syntax of using it | * library(ROCR) * We have logistic regression predictions on a set df saved as pred = predict(model,type="response"), where the outcome variable in the model is Y. * ROCRpred = prediction(pred, df$Y) * ROCRperf = performance(ROCRpred,"tpr","fpr") * plot(ROCRperf) |
| What if we want to add color to our ROC curve? Or specify specific values of the threshold along the way? | in plot() add colorize=TRUE and print.cutoffs.at=seq(0,1,0.1) for example. |
| What does the AUC tell us? | How good our model is; higher is better. |
| Note about the newdata argument | It's not only for logistic regression models. Rather whenever we build a model with one data set then want to apply that model to make predictions on a new data set (for example a test set) we use this argument. |
| When can we choose a smaller SplitRatio? | When we have a large data set to split over |
| Why would we want to? | It leaves us a larger test set and therefore a more reliable out of sample accuracy. |
| For a logistic regression model with a given threshold how do we compute its accuracy on a given set? | In the confusion matrix we sum the diagonals and divide by the sum of the table |
| What might we want to compare this accuracy to? | The accuracy of the baseline model, found by taking the largest row sum in the confusion matrix and dividing by the sum of the table. |
| What should happen? | If our logistic regression model is any good it should have an accuracy at least that of the baseline, but ideally significantly larger. |
| How do we calculate auc? | as.numeric(performance(ROCRpred,"auc")@y.values) |
| How do we interpret auc? | If we have two outputs chosen at random such that one has one binary outcome value and the other has the other then the auc is the probability that we can successfully differentiate between the two. |
| Internal validation | When a model is tested on a set of observations who have similar qualities to the training data (e.g. demographics) |
| External validation | When a model that was built on one set of observations is used to test on a set of observations that are quite different (for e.g. if the original was built on white, middle-aged men and then we want to test it on other ethnicities, ages and women). |
| Suppose we are trying to build a model from a data set df that has a lot of NAs in some of the variables (x and y) we are trying to use to build our model and there are other variables that are related to x and y, say z. What do we do? | We could remove all observations that have an NA value, but this might get rid of a lot of observations. We could get rid of all variables that have an NA value but this could get rid of some valuable variables. We could fill in those NA values using multiple imputation by chained equations.   * library(mice) * simple = df[c("x","y","z")] * imputed = complete(mice(simple)) * df$x = imputed$x * df$y = imputed$y |
| Which variables should be included in the imputation? | Aside from the ones with NA values we want the ones that are related to the variables whose NA values we are trying to compute (so probably we should keep only numerical variables?). However, we \*don't\* want to add in the dependent variable as this would create a circular argument when using these independent variables to model the dependent variables. |
| What might we want to do when doing multiple imputation? | set.seed() |
| Is imputation easy? | In R it is easy to use but computationally it is complex. |
| How to obtain whether a variable x is +,- or 0? | sign(x) |
| The possible outputs? | -1,0,1 |
| \*How does the polling example work? | * We have training set (data from 2004 and 2008) and a test set (from 2012) * To obtain our initial training set we predict the most common outcome: 53R and 47D -> predict R, which has only a 53% accuracy. * So we use an improved baseline model which instead uses the sign of the Rasmussen variable to predict the outcome; if 1 then predict R and if -1 then predict D. * By creating table(train$Republican, sign(train$Rasmussen)) we find an accuracy of 94%, which is much better. * Now we use logistic regression to try to build an even better model. First we find the correlations between each of the numerical variables, including Republican, and find the single dependent variable which is most highly correlated, which is PropR and build a model with just this independent variable. * When making predictions using this model we use a threshold of 0.5 because this means when the proportion of votes who voted R is greater than half then we predict R. * table(train$Republican, pred1>0.5) tells us the accuracy has now improved to 96%. * Now we try building a logistic regression model using two independent variables. However we see that PropR is highly correlated with all of the other independent variables, so we can't use this one. Instead we find two which have a low correlation, such as SurveyUSA and DiffCount. * We repeat the above making predictions again and find an accuracy of 97%, which is good, and the coefficients are positive, as we would expect, and the AIC has decreased which is also good. However the significance codes are low. * Then we test the improved baseline model and this second lrm on the test set. We find the improved baseline model has an accuracy of 87% whereas the second lrm has 98%. |
| If we have a dependent variable Y as our outcome variable in a model and the data set has many independent variables and we want all except for x, how do we write this? (Say for logistic regression) | glm(Y~.-x, data=..., family=binomial) |
| How do we look at the observations of data frame df corresponding only to observations where the values of its variable x is NA? | subset(df, is.na(x)) |
| How do we look at the observations of data frame df corresponding only to observations where the values of its variables x or y are NA? | subset(df, is.na(x)|is.na(y)) |
| For numerical variable x how do we find for example the 67th smallest observation? | sort(x)[67] |
| How does table(x) arrange its values for numerical variable x? What is another way? | * It lists all possible values of x in increasing order, with its frequencies in the second row. * Using sort(table(x)) we could instead make the columns so that the frequencies are instead in increasing order. |
| What if we instead want to look at x in decreasing order? table(x)? | sort(x,decreasing=TRUE)  sort(table(x),decreasing=TRUE) |
| A downfall of logistic regression models: | They are not easily interpretable. Specifically, the coefficients are difficult to interpret directly because of the definition P(y=1) = 1/1+e^-(logit), where logit = log(odds) and odds=P(y=1)/P(y=0). Logistic regression is defined in terms of log(odds), and odds do not have an intuitive interpretation the same way that probabilities do. For example if we have a coefficient of 0.3 and increase the corresponding variable by 1 then we have increased the odds by a factor of e^0.3=1.35 or 35%, but again, this has no intuitive interpretation. |
| How are odds of event A defined? | P(A)/(1-P(A)) |
| Note that logistic regression is not a linear model but a generalized linear model. What does this mean? | The decision boundary, as defined by the logit, is linear, as we can see from the S-shaped graph when P(y=1) = 1/1+e^-(logit) is defined and logit is along the x-axis. The decision boundary is the vertical line at logit=0. Namely, if the value of the logit is to the right of this boundary (that is, if logit>0) then P(y=1)>0.5 and so we predict class 1; if the value of the logit is to the left of this boundary (that is, if logit<0) then P(y=1)<0.5 and so we predict class 0. |
| What is another way to model binary outcomes? | Classification trees (CART models) |
| Can we use CART with an outcome variable that has more than 2 possible values? | It appears we can use classification trees for factor/categorical dependent variables with more than 2 possible outcomes but we focus more on binary classification (we do one example with 5 outcomes, using a penalty matrix). Regression trees are for numerical outcome variables. |
| What is one main difference between a binary classification tree and logistic regression? | We are no longer dealing with a generalized linear model. That is, the decision boundary is no longer defined by a single line which divides the plane into two cases, each of which predicts one of the two outcomes, but rather by many lines that split the plane (or space) into many regions (for e.g. rectangles in the case of a plane) each of which predicts one of the two cases. |
| How does it work? | R splits some of its independent variables into binary cases for example using < and > (if numerical) or if the independent variable has one value go left in the tree, otherwise go right (even if the independent variable is not binary). If creating split X<60 for example then along the LHS of the split X<60 and along the RHS X>=60. After all splits have been made, each region is classified by the outcome that occurs most frequently in that region, and R tries to create splits so that each region is as homogeneous as possible in its outcomes. When making predictions we find which region an observation falls in and use that classification for its predicted outcome. |
| One way to control the number of splits, and what does it do: | Use the minbucket argument. It controls the minimum number of observations from the training set that must occur in each created region, called 'buckets'. |
| What if it is too low? Too high? | If too low then the model will fit the training set almost perfectly but we risk overfitting and therefore poor extrapolation to new data. If too high then there will be few buckets and the model will be too simple and inaccurate. |
| Overfitting means | A model is excessively complex, for example if having too many variables relative to the number of observations or if minbucket is too low and therefore the space is divided into too many regions. This second case may be a problem because if you have divided your space into too many regions then each region will receive less observations and therefore the prediction made by that region will be less reliable as it is caused by a smaller sample size. |
| What do we need to build CART models? | library(rpart), library(rpart.plot) |
| Syntax for building a classification tree: | * rpart(Y~X1+...+Xk, data = ..., method="class", minbucket=...) * method="class" specifies that we are making binary/categorical predictions (so using classification trees, not regression trees). |
| How to display a classification tree: | prp() |
| Syntax of making predictions from a CART model called CART (two ways): | * predict(CART, newdata=test)   This outputs probabilities for each of the two possible outcomes (0 or 1) for each of the observations.   * predict(CART, newdata=test, type="class")   type="class" tells us that predictions are of type classification, so 0 or 1 for each data point, determined by which has a higher probability. When creating a confusion matrix this is like using a threshold value of 0.5. |
| If we use the first syntax above, what would we do in creating a confusion matrix with a threshold of 0.5? | The observations are the rows, the binary outcomes 0 and 1 are the columns, with probabilities being the entries, so we need to extract the probabilities in the second column, pred[,2] and set >0.5 in the table used to create the confusion matrix. We use pred[,2] to maintain our usual convention that the right-most column in the confusion matrix corresponds to outcome value 1 and the left to 0. |
| How are the probabilities decided for observations in a CART model? | Each observation in a given bucket has the same probabilities for both outcome 0 and 1, determined by the proportions of 0 and 1 outcomes of the training set in that bucket. |
| How do we create the ROC curve for a CART model used on test and outcome variable y | Similar to logistic regression with a couple differences:   * pred = predict(model, newdata=test) * ROCRpred = prediction(pred[,2], test$y) * ROCRperf = performance(ROCRpred, "tpr", "fpr") * Here we don't want pred to have the argument type="class" because this corresponds to just t=0.5. * In ROCRpred notice we are taking just the second column of pred, so the probabilities of outcome being 1 over all observations. |
| When might we use random forests? | Classification |
| \*What must be true about the outcome variable when creating a random forest? | It must be factor/categorical, so we may need to change the variable if it is for example of type int with values 0 and 1. |
| What is random about a random forest? | * Which observations are used to grow each tree (we take a set as large as the training set but with possible repetitions; bootstrap) * Which variables are split over. |
| What is the gist of how random forests work? | A random forest produces many CART trees, each of which votes on the outcome (0 or 1) for each observation just as a CART tree would, and the final outcome for that observation is determined by the majority vote. |
| What is the tradeoff between a random forest and a CART model? | Random forests offer a higher accuracy, but they are less interpretable and more computationally intensive. |
| What do we need to build a random forest model? | library(randomForest) |
| The syntax for a random forest | * randomForest(Y~X1+...+Xk, data=..., nodesize=..., ntree=...) * nodesize is like minbucket; smaller nodesize means more splits and therefore larger trees * ntree is the number of trees in the forest |
| k-fold cross-validation: what is it for? | When building a CART tree rather than using minbucket we may want to use another parameter called the complexity parameter(cp). K-fold cross-validation allows us to compute the optimal cp value (optimal mean it offers higher accuracies on the training data and test sets for e.g.), so a good parameter for our CART model. |
| How does it work? | Our training data is divided into k equal-sized subsets ("folds"). For each possible cp value, each subset takes a turn being the "validation set", which is a test set, while the other k-1 sets are used as training data to build a CART model. This CART model then computes the accuracy on the validation set. For the given cp value we take the average accuracy over the k accuracies. This is repeated for each possible cp value and the chosen cp value is the one which offers the highest average accuracy. |
| The range of possible values for cp | It seems to be 0 to 1. |
| What is cp comparable to? | Adjusted R2 for linear regression or AIC for logistic regression; it captures model accuracy on the training set compared to model complexity. |
| Syntax of computing the optimal cp value | * library(caret) and maybe library(e1071) * Decide k (e.g. 10) and the range of possible cp values you want to check (e.g. seq(0.01,0.5,0.01). * folds = trainControl(method="cv", number=10 * cpGrid = expand.grid(.cp = seq(0.01,0.5,0.01)) * train(Y~X1+...+Xn, data=..., method="rpart", trControl=folds, tuneGrid=cpGrid) |
| What does this last command output? | A list of cp values with their accuracies (for a categorical variable) or RMSEs (for a numerical outcome variable); we choose the cp value corresponding to the highest accuracy, which is displayed at the bottom of the output. |
| So what is train() doing? | For a fixed cp value in seq(0.01,0.5,0.01) an accuracy/RMSE is computed for each of the 10 models on their corresponding validation sets and the accuracy/RMSE corresponding to a given cp value is the average of these 10 accuracies/RMSEs. Train does this for each cp value and we choose the cp value corresponding to the largest accuracy/lowest RMSE. |
| What do we do once we have found the optimal cp value? | Create a cart model using that cp value. |
| If we want to divide all the entries in a table() by a number c: | table()/c |
| An example of when would this be useful: | For example if we want to find relative frequencies of certain outcomes by dividing by the number of observations in the table. |
| D2Hawkeye example, how do we convert to classification? | Each patient has a medical costs ($) variable over the year. However, we can group these costs in 5 buckets by creating cost ranges in such a way that each bucket receives 20% of the total costs. For example the lowest cost patients, <3000$, would be in bucket 1, etc. |
| What is a penalty matrix? | In classification problems this specifies the value of the error made between each outcome/prediction pair. In D2Hawkeye the values range from 1 to 5 and so predicting 3 while the actual outcome is 5 would have error 4 according to the matrix. |
| What can we say about classification trees that have more than 2 outcomes? | It seems that the splits themselves are still binary. |
| How to create a matrix: | matrix(c(5,3,5,9,2,9), byrow=TRUE, nrow=2) |
| If M1 and M2 are matrices what does M1\*M2 do? | Multiplies them entry-wise, assuming they are the same dimensions. |
| What is the penalty error and how is it found? | It is the average size of error made by a prediction. We multiply the penalty error matrix by the outcomes vs. predictions table (converted to a matrix), sum the values of this new matrix and divide by the number of observations. |
| What often happens when we find a solution that has a lower penalty error? | Accuracy goes down as well. This is because unlike accuracy, which gives each error a weight of 1, this method prioritizes avoiding higher-error cases even if that means making more mistakes overall. |
| We can add the penalty matrix into our CART model so that the model uses these weighted errors instead of assigning each error a weight of 1 when deciding its splits. How and what happens in the end result? | * Add argument parms=list(loss=penaltyMatrix) * The penalty error goes down but so does accuracy. |
| When do we use regression trees? What is the value at each leaf? | When our outcome variable is continuous rather than categorical. The value at each leaf, rather than being a category (for example 0 or 1 in binary, determined by the mode) is instead the average of the points that occur at that leaf. |
| What do regression trees do? | Like classification trees they can capture non-linearities that linear regression can't. It groups the data into subsets. Each subset will have an average, and this average is used for the model's outcome value of the data points in that subset. (Think of a horizontal linear regression line for each of the subsets.) |
| If we have x=c(4,3,9) and we want to obtain a vector with just the first and third coordinates, how do we do this? | x[c(1,3)] |
| Suppose we have an existing plot(x,y) and we want to manipulate a subset of the points, say the second and fourth observations, for example to add colors to this subset. How do we do this? | points(x[c(2,4)],y[c(2,4)], col="red") |
| What if we want to fill these points in? | add pch=19 |
| If we have x[logical] what does this return? | This runs through all values of x as well as the value of the logical at each step and outputs a vector that consists of the values of x where the logical is TRUE. |
| What if we want to add a specific symbol for our points in the plot rather than the previously-defined ones? | e.g. pch ="$" or any symbol in place of $ |
| On a plot how might we compare a subset of points to another subset of points. | Write a points() argument for each, giving them different colors/symbols to see how they overlap. |
| An example when we might use this | When we want to look at which points correspond to a numerical variable that is at least a certain value as well as which points a model predicts for this variable to be above that same certain value. (See latlontree example) |
| \*How do we know when to consider using a regression tree? How do we check if the model we build is appropriate? | When the outcome variable is numerical. We can first start by looking at a linear regression and see if it is appropriate by looking at its R2 or a plot for example. Then we can create a regression tree and for example plot by converting to a binary case using > or < to see how actual values and predicted values overlap. We could also look at the RMSE and compare it to the linear regression case. If the linear regression has a low R2 yet the RMSE for the regression tree is still worse, this is not a good sign for the regression tree; if the linear regression has a high R2 and the regression tree has a better RMSE then this is a good sign for the regression tree. |
| Syntax of creating regression tree | Same as classification tree except omit method="class" |
| When we say that a plot with Var1 along the bottom, Var2 along the side and the points colored according to whether or not the outcome variable is greater than a given value, what does it mean that there is not a linear relationship? | The colored points are scattered. Recall the linear regression equation; if both coefficients were positive and linear regression were appropriate then this would mean the colored points are more concentrated the higher Var1 and Var2 are; if both coefficients are negative then the colored points are more concentrated the lower Var1 and Var2 are, etc. It's probably easier to just look at R2. |
| \*How can we check if there is over-fitting in a regression tree? | * plot it using prp() and if there are too many splits then it might be over-fitted. * Check the SSE or RMSE on a test set; it might be higher than in the linear regression case (even if we have already concluded linear regression is not a good fit). |
| \*What can we do if this is the case? | Increase minbucket or cp; particularly we can find the best cp value using train(). |
| When train() is used to find the optimal cp value, what is it based on in the classification and regression tree cases (different in both)? | For classification trees on best accuracy, on regression trees best RMSE. |
| If we plot it using prp() but the name of the splits are not ideal (because they don't display decimals), what is another way? | * plot(model)   text(model)   * Or in prp() add the argument digits = (some number) |
| Add a horizontal/vertical line to an existing plot with x and y axes: | abline(h=72.2)  abline(v=43.2) |
| Difference between computing a measure of errors in categorical case and numerical case | For categorical we use a confusion matrix to get an accuracy; for numerical we use RMSE for example. |
| One way of seeing over-fitting of a regression tree: | Calculate the RMSE on the test set |
| Another way of writing the sequence 0.3, 0.6, 0.9, ..., 3 that isn't seq(0.3,3,0.3) | (1:10)\*0.3 |
| Ask why MEDV~.-TRACT-TOWN didn't work p.51 |  |
| What is it called when dealing with preprocessing data/words, etc. | natural language processing |
| What are the related models called? | text analytics models (but still CART, random forests, etc.) |
| What is Amazon Mechanical Turk? Give an example | Owned by Amazon, it allows businesses to coordinate human intelligence to complete tasks that computers are currently unable to do. It connects the business to the humans who can complete the tasks; the humans receive payment from the business and Amazon takes a cut. E.g. Apple may want to rate certain tweets as positive, negative or neutral |
| What is the generic text analytics model we use called and what is it? | Bag of words, where each word is treated as a variable whose value in a given observation (for example, in one tweet) is the number of occurrences of the word in that observation. |
| \*What should we remember when dealing with a text analytics problem? | Use the argument stringsAsFactors=FALSE when loading the data set because we want our words read properly (for example as chr, which means character) |
| What library do we need for text analytics/pre-processing? | library(tm) |
| If we have a data frame, what is it we are preprocessing? | One of its variables, e.g. tweets$Tweet |
| How do we do it? | * corpus = VCorpus(VectorSource(tweets$Tweet)) * corpus = tm\_map(corpus, content\_transformer(tolower)) * corpus = tm\_map(corpus, removePunctuation)) * corpus = tm\_map(corpus, removeWords, stopwords("english")) * corpus = tm\_map(corpus, stemDocument) * frequencies = DocumentTermMatrix(corpus) * sparse = removeSparseTerms(frequencies, 0.995) * final = as.data.frame(as.matrix(sparse)) |
| If we enter the command corpus, what happens? | Nothing interesting |
| If we enter the command frequencies, what happens? | It tells us the number of documents (which is the number of observations in the original data frame) and the number of terms (the words are the new variables, called terms). |
| If we want to find all terms which have a total frequency of at least 20: | findFreqTerms(frequencies, lowfreq=20) |
| difference between sum and colSums | colSums is for adding columns in matrices, sum is for adding a column in a data frame |
| Why might we use findFreqTerms? | To get an idea of how many words actually appear with high frequency; the others are likely not very valuable for our predictive model. |
| What does sparse as defined above do? | It creates a new document term matrix that keeps only the terms (words) in frequencies which occur in at least 0.5% of the documents (observations). |
| One important detail when building text analytic models: | colnames(final) = make.names(colnames(final))  This is because R struggles with variable names that start with a number and this command fixes that. |
| What does a lower cp mean? The tradeoff? | It means more splits and therefore a higher accuracy/lower RMSE on the training set but at the risk of over-fitting and therefore lower accuracy/higher RMSE on a test set. |
| It seems we may want to consider set.seed() whenever | Something random is about to happen, such as a split or a random forest. |
| For a CART model where the outcome variable is categorical, what happens if we omit the argument type="class" in the prediction? | It instead makes predictions as probabilities for each of the categories rather than choosing the mode outcome. |
| What if you create a random forest and it outputs a bunch of decimal numbers instead of class predictions? | Remember that the outcome variable should be converted to a factor variable. |
| What do we do after we have finished preprocessing and have our desired data frame? | We can start building text analytics models and computing their accuracies, for example classification trees or random forests. |
| Random forests typically offer higher accuracy than classification trees, so why might we choose the latter? | Increased interpretability/ease of computation. |
| What is the eDiscovery problem? How did it used to work, and how does it work now? | In law it refers to being able to find documents relevant to a given case, called responsive documents. This was solved using keyword search. Now predictive coding is done, whereby lawyers manually label some documents as responsive/non-responsive and use these labeled documents as training data which can used to create a text analytics model that predicts whether or not the remaining documents are responsive or not. |
| What predictive coding problem do we focus on in the course? What are we predicting and how? | Enron, because FERC has released 600k+ emails from Enron. We are building a model based on 855 of those emails that can predict whether or not each email (1 per observation) is related to energy schedules. Lawyers manually labeled these 855 emails as responsive/not responsive, and we build a predictive model from this data set (energy\_bids.csv.) |
| What function do we use if some text is too long and continues on to the right of the R console but we would instead rather it broken into multiple lines? | strwrap() |
| Which predictive model did we use and how did we assess its predictions? What will the lawyers do after predictions have been made on the 600k+ emails (if I understood correctly)? | A classification tree. We look at its confusion matrix on a test set built from the 855 observations as well as its accuracy. They will manually look through all of the ones predicted as responsive in the 600k set; so FP's cause extra work but will be correctly identified. However the FN's will be missed, so they are penalized higher. |
| What should we do after we have computed the accuracy/RMSE of a model? | Compare it to the accuracy/RMSE of a baseline; it should be at least as good as that, but ideally substantially better. |
| How do we use the ROC curve to help determine our ideal threshold value t? | It depends on which types of errors we prioritize. The left of the graph corresponds to a lower FPR but higher FNR; the right is the reverse. |
| How to calculate auc: | auc = as.numeric(performance(ROCRpred, "auc")@y.values) |
| What does a higher FPR rate mean? A higher FNR? Why? | Higher FPR means lower TNR and higher FNR means lower TPR. This is because FPR+TNR=1 and FNR+TPR=1 |
| What is the range of possible values for auc and what are the interpretations of the extremities? | 0.5 (worthless test, no better than flipping a coin) and 1 (a perfect test) |
| What is the interpretation of auc? | (I think, double check if this is correct): Suppose we have two random data points, one with actual outcome value 0 and one with actual outcome value 1, but we don't know which is which. For both of these data points our model is going to have different probabilities for predicting 0 and 1. If the model were to guess that the point with a higher predicted probability for 0 (and therefore a lower predicted probability for 1) corresponds to the point with actual outcome 0, and the other point as having an actual outcome of 1, then the auc represents the probability that the model would be correct. |
| Why can't auc be less than 0.5? |  |
| If we have a data frame df but want to change the variable names so that each starts with a given character, say A, how do we do this? What exactly does this do? | colnames(df) = paste("A", colnames(df))  It leaves a space between the A and the original variable name. |
| What if we don't want this second part? | colnames(df) = paste0("A", colnames(df)) |
| Syntax of grepl() and what is it for? What is the output? | It is for determining if the first string is a substring of the second: grepl("why", "why not", fixed=TRUE). The output is TRUE or FALSE |
| Syntax of ifelse? What is the output? | ifelse(expression,a,b); expression is a vector that outputs TRUE or FALSE; when TRUE return a, when FALSE return b. The result is a's and b's, one for each value in expression. |
| How to sum the columns of a matrix M: | colSums(M) |
| How to sum the rows of a matrix M: | rowSums(M) |
| How to count the number of symbols in a string: | nchar() |
| What types of predictions do random forests make by default? What is another way and how do we achieve it? | Class predictions by default. If we want probabilities for each class then in our prediction we can add type="prob" and then we can extract the relevant probability if needed for example using [,2] to take the probability associated with outcome value 1. |
| What can we say about CART models in comparison to above? | They are the opposite; by default they make probabilities for each possible outcome and if we want class predictions we have to add an argument, type="class". |
| \*A good indication of over-fitting in classification? A good indication of not over-fitting? | When a model has high accuracy and auc on the training set but low on the test set this suggests over-fitting. If the model has high accuracy and auc on the training set and similar values on the test set then this is a good sign of not over-fitting. |
| \*When is the ROC curve appropriate (what types of models)? | It seems for any binary classification model. |
| What was the Netflix challenge? | Participants were provided with a training set of 100m points and a test set of 3m points. They had to build a predictive model that was better than Netflix's recommendation system by at least 10%, as measured by RMSE. |
| What are the two main types of filtering Netflix used? Their weaknesses? | * collaborative filtering; it requires a lot of data and computing power * content filtering; limited in scope, as it does not recommend movies too different than the one's a person already watches |
| What does Netflix use to combine them? | hybrid filtering |
| What is the difference between supervised learning and unsupervised learning? | In supervised learning you have input data (X) and output data (Y) and you try to learn the mapping f given by Y=f(X). In unsupervised learning you only have input data (X) and (in the case of clustering, which is the most common type) you group data points according to commonalities in this input data. |
| Everything up until now has been supervised learning. |  |
| \*What do we have to remember to do when clustering? Why? What do we have to remember when done? | Normalize/standardize the data first (It seems like in the lectures when they say "normalization" they actually mean "standardization"; they say subtract the mean then divide by the standard deviation for each variable, which is standardization). We need this because if not then when the clusters are formed the large-scale variables will overwhelm in the distance calculations, rendering the other variables essentially irrelevant. When done, we have to remember to revert back to the original values; the standardization is only for determining cluster groups. |
| Two main types of clustering: | hierarchical, k-means |
| What is the graph associated with hierarchical clustering called, what is it for and how do we interpret it? | A cluster dendrogram. It is to decide how many clusters/what the clusters we will use from hierarchical clustering. The length of the line between a child cluster and its parent represents how distant (a.k.a. different) they are. |
| How do we use the horizontal line method and why do we use it? | We want to choose clusters by drawing a horizontal line in such a way that the horizontal line can move up and down as much as possible without crossing another horizontal line. The resulting clusters are decided by the clusters immediately beneath each intersection point of this horizontal line with vertical lines in the dendrogram. The interpretation is that the more we can move it without intersection the further (more different) the child clusters are from their parent and therefore each other, as desired. |
| What else should we keep in mind when deciding how many clusters? | * The particular application in mind. For example if clustering by movie genres 2 clusters would probably not be enough. * So we don't want too few clusters as this would be a very coarse clustering, be we also don't want too many as this would spread the data too thinly. |
| When reading a data set from Excel using read.csv() what does it assume about the first row in Excel? What do we do if this assumption is not valid? | The first row corresponds to variable names. If it doesn't (i.e. there are no variable names and the data starts on the first row) then add header=FALSE when reading in the data. |
| \*What does R do when we make this change? | It labels the variables as V1, V2, etc. |
| How can we fix this? | Manually add in names using colnames(df) = c("...","..."...) |
| What is another type of file we loaded in the lectures instead of using read.csv and how was it loaded? | .txt file; it was loaded using read.table("moviesLens.txt", header=FALSE, sep="|", quote="\"". |
| Why did we use the third argument? | If we open the .txt file we see the values are separated by this symbol rather than commas. |
| How do we remove a variable x from a data frame df? | df$x = NULL |
| What might happen to a data frame when a variable is removed? How can this be fixed? | There might be duplicate observations; this can be fixed using df = unique(df) |
| The difference between standardization and normalization and how are they calculated? | It seems in the lecture when "normalization" is used the teacher should have been saying "standardization", which means the data are rescaled so as to have a mean of 0 and sd of 1 using  xnew=(xold-mean(x))/sd(x). In normalization the data is rescaled so that each data point lies in [0,1] using xnew=(xold-xmin)/(xmax-x­min). |
| What is the interpretation of a standardized value? | How many standard deviations the non-standardized value is from its mean. |
| A weakness in each: | Standardized values are not bounded. For normalized values, if there are outliers (usually there are) then the "typical" data points will be more prone to being squished into one small section of the interval [0,1]. |
| How does the dist() function work and what does it output? | dist(df,method="euclidean"), where the variables of df are numerical. It outputs a matrix with the distances between each pair of observations. However it's not a matrix per-se; it only shows 1 distance per observation pair rather than displaying it twice, and omits the distances between any observation and itself. In other words, it omits the main diagonal and everything above it. |
| If x = c(3,5,2) and y = c(5,8,3,4) then what is c(x,y)? | (3,5,2,5,8,3,4) |
| What are the relevant steps for hierarchical clustering? | * Prepare the data frame by removing unwanted variables/changing variable names/using unique() * distances = dist(df[c(relevant variables)], method="euclidean") * dendrogram = hclust(distances, method="ward.D") * plot(dendrogram) * clusterGroups = cutree(dendrogram, k=*#ofcutswedecided*) |
| What does this last command do? | It gives each observation a number from 1 to k. |
| How could we use tapply? | tapply(movies$Action, clusterGroups, mean) for example, which would tell us which % of movies in each cluster group are action. We could do this for each genre and then label the cluster groups according to which genres occur most in their cluster group. |
| When is clustering mostly used? | In the absence of a target variable so as to organized independent variable/data into meaningful groups. |
| What happens if the outcome variable is present? | We may still want to cluster our input data because it might give better predictions. We could remove the outcome variable temporarily while clustering then add it back in when building a predictive model. |
| Vaguely, what is cluster-then-predict? | The train set and test set are divided into clusters. For each train/test pair predictions are made from a model made from the train set on the test set and an accuracy is computed. The overall accuracy is the weighted (by #observations) average accuracy over all the models. |
| Cluster-then-predict, what is it/how does it work? | See page 24 |
| How is accuracy calculated for cluster-then-predict? | The weighted (by # of observations) average accuracy of the models over each cluster. |
| What is the benefit of cluster-then-predict? | It can offer improved accuracy over creating a model from a train/test set without clustering. |
| What if we have an initial data set on which we want to use cluster-then-predict, but we want to separate the data by a specific variable (not the outcome variable)? | We can first bucket the data set according to that specific variable. Then we can cluster-then-predict on each bucket. For example maybe we want to separate by medical costs of a patient (low, medium, high) and derive conclusions regarding heart attack predictions for each; it is fair to assume that predictions will be different in these different buckets because medical costs are an indicator of a person's health. |
| What is image segmentation and what is its goal? | It means dividing a digital image into segments (groups of pixels/clusters). The goal is to simplify/modify the image into something that is more meaningful and easier for us to analyze. |
| How are clusters built? | * According to pixels that share similar visual characteristics. * The pixels in different objects are clustered differently (even if of similar colours). |
| For image segmentation in this course what are the observations and independent variables? | Each pixel in an image corresponds to an observation and the pixel intensity is the independent variable. |
| \*How is grayscale image data represented? How many pixel intensities are there? What do the extreme values correspond to? | As a matrix of pixel intensities; 256; 0~black, 1~white. |
| One important thing to remember when we want to do image segmentation? | Remember to first convert the intensities matrix to a vector using as.vector() because these values represent one variable. |
| What do we do if we don't want any axes for our plot/image? What does this do? | axes=FALSE; it's not for the labels of the axes but for the numbers along the axes. |
| Steps for image segmentation in R (hierarchical clustering): | * Read in the data frame, probably using the argument header=FALSE * Each entry in the data frame will correspond to an observation (unlike usual, where each row corresponds to an observation) so we first convert the data frame to a matrix using as.matrix() and then to a vector using as.vector(). * distances = dist(vector, method="euclidean") * dendrogram = hclust(distances, method="ward.D") * plot(dendrogram) to decide how many cluster groups is appropriate (say 3). * clusterGroups = cutree(dendrogram, k=3) * dim(clusterGroups) = c(nrow(originalmatrix), ncol(originalmatrix)) * image(clusterGroups, axes=FALSE, col=grey(seq(0,1,length=256))) |
| What happens if we try hierarchical clustering for image segmentation on images with high resolution (too many pixels)? What do we do instead? | R can't compute the distances because there are too many, so instead we use k-means clustering. |
| How do we do it in R? | * We might want to consider setting a seed due to the randomness in k-means clustering. * Once we have converted our data frame to a vector as above, we use KMC = kmeans(vector, centers = k, iter.max=1000) * This automatically creates the cluster groups, which we can extract using KMC$cluster. * Then the rest is the same as in hierarchical clustering. |
| Why do we have iter.max? | KMC is an iterative process than can take a long time to converge. |
| What happens if we don't set seed? | The colours in the final image are different. |
| What can we add if we want to colour our image but not with greyscale or the standard red/white/orange/yellow? | col = rainbow(k) |
| How would k be decided for k-means clustering for example in brain image? | It would likely be the number of brain regions/structures we are interested in looking at. |
| From a kmeans object (KMC) we can build predictions. How, and what does it do? | * library(flexclust) * KMC.kcca = as.kcca(KMC, healthyVector) where healthyVector is the training set and data from which KMC was built * pred = predict(KMC.kcca, newdata=tumorVector) where tumorVector is the test set * pred is a vector that predicts the cluster for each of the pixels of tumorVector (rather than our usual predictions in supervised learning that predict the outcome variable). We can then use this vector to convert to a matrix and image as usual. |
| What does KCCA stand for? | K-centroids cluster analysis |
| A pro and con of hierarchical clustering? k-means? | For hierarchical a con is that it doesn't work on large data sets because of the number of distances that need to be computed; a pro is that you don't need to select the number of clusters beforehand because you can visualize with a dendrogram. For k-means a pro is that it works on any size data set; a con is that you have to specify k beforehand. |
| How do we look at the first 6 values of a variable? The last 6? | head(x), tail(x) |
| Note that some texts use "normalization" where I am using "standardization", and this course in particular does that. |  |
| \*How to standardize df: | * library(caret) * preproc = preProcess(df) * dfNorm = predict(preproc, df) |
| What does each of these commands do? | It seems that preproc determines the mean and sd for each variable in df and the second applies the transformation to the stated data frame in predict(), in this case df, but it could be any other with the same variable names, in which case we would need the argument newdata=df2. |
| When preprocessing data (i.e. standardizing) that is separated into a train and test set, what must we remember to do? | For the test set we are using the means/sds of the train set. |
| How do we do cluster-then-predict for k-means clustering (the steps)?  (see p. 87 in The Analytics Edge Word document for example) | * Break data into train/test * Remove the outcome variable from both (but don't overwrite the original train/test) then preprocess by standardizing (using the train's mean/sd for the test). * Create KMC = kmeans(normtrain, centers=k) * Use KMC to make predictions for the cluster groups of normtrain and normtest as follows: * library(flexclust), KMC.kcca=as.kcca(KMC, normtrain) * clustertrain = predict(KMC.kcca) * clustertest = predict(KMC.kcca, newdata=normtest) * train1 = subset(train, clustertrain==1) (do this for each k) * test1 = subset(test, clustertest==1) (do this for each k) * Build a model using train1 and make predictions on test 1 (do this for each k) * Create alltestoutcomes = c(test1$outcome, ... , testk$outcome) and allpreds = c(pred1, ..., predk) * Find total accuracy by table(alltestoutcomes, allpreds>0.5) |
| \*What if we want to cluster-then-predict using a different clustering method? | I'm not sure the above would work because kcca requires k to be a specific number. Retry the homework problem on p.87 with hierarchical clustering, inserting "dendrogram" instead of "KMC". |
| Data visualization, brief definition: | A mapping from digital properties to visual properties. |
| What is the classic example involving 4 data sets that look similar from digital properties but are quite different visually? | Anscombe's Quartet |
| What package do we use for data visualization? | library(ggplot2) |
| What are the three parts that exist to ggplot graphics? | * data frame * aesthetic mapping (describes how data are mapped, axes, size, color, etc.) * geometric objects (determines how data are rendered graphically, e.g. points, lines, rectangles, polygons, etc.) |
| How does aes() work between putting it in the ggplot() argument or a geom() object? | Any argument in aes() defined in ggplot() is inherited by any geom() object built on top of it. However if geom() has a conflicting aes() argument then it overrides the aes() argument defined in ggplot. |
| Some basic geom\_ types: | geom\_point()  geom\_line() geom\_bar()  geom\_polygon()  geom\_tile() |
| Basic syntax of creating a plot from ggplot2 | *plot* = ggplot(df, aes(x=..., y=...)) + geom\_...() +otherlayers... |
| What does the first part do? | It makes the plot but without actually plotting any values; that's what the geom\_() arguments do. |
| What if we want to add a title to our plot? | + ggtitle("....") |
| What if we want to change the x-axis name in our ggplot object? | + xlab("...") |
| \*Suppose we have concluded that a linear regression models the data well and we want to add this line to our plot. How do we do this? | + stat\_smooth(method="lm") |
| How do we create and save a pdf to the current directory in R? Note about this? | pdf("MyPlot.pdf")  print(*plot*)  dev.off()  Note: Make sure we are not working in a directory that is restricted by administrator privilege; if so then move that directory, save the plot there, then move the directory back if need be. |
| What is each item after a + called? | a layer |
| Some examples of arguments we can add inside geom\_() | color = , size = , shape = |
| What do the above arguments do? | Apply these arguments to all of the data points; so they would all have the same color, size, shape. |
| What other option do we have? | Suppose we want the data points to be colored differently according to their value of a variable, z. Then we can add the argument aes(color=z) in ggplot() or geom\_(); if a factor variable then we will see explicitly different colors and if numerical we will see a gradient. Similar for size, shape, etc. |
| What is data visualization great for? | Picking up on relationships between data that might easily be missed otherwise. |
| What should we generally try to do before making a predictive model? | Try to understand the historical data. |
| What happens by default when a data set is loaded in R and some of the variables have text in their values? How can we change this? | They are read as factor variables; they can be read as they are using stringsAsFactors=FALSE |
| What is their type now? | Type chr (for character) |
| What is the main difference between as.Date() and strptime()? | strptime() also allows for times |
| Recall the standard form R can read dates/times is: | "yyyy/mm/dd HH:MM:SS" |
| \*What variable type do as.Date and strptime accept? | chr (but it seems factor does indeed work too; I'll use stringsAsFactors=FALSE just in case) |
| Syntax for strptime() | If we have a date variable of the form m/d/y H:M:S for example then use strptime(datevariable, "%m/%d/%y %H:%M:%S"); if for example of the form H:M:S y/d/m then strptime(datevariable "%H:%M:%S %y/%d/%m") |
| What is the output format? | "yyyy-mm-dd HH:MM:SS EST" |
| For as.Date and strptime how do we know if we should use y or Y after the %? | We use y if the year in the original string does not include its century (for example 14 for 2014) and we use Y if it does include its century (for example 2014). In the first case it will always be prefixed by 19- or 20-, namely 20 for 00 to 68 and 19 for 69 to 99, so it's only good from 1969 to 2068. |
| What happens if we omit the year? The seconds? What happens if we omit any date? | For the year it fills in the current year; for seconds, nothing, they are just forgotten. For date it fills in the current date. |
| When a date is in the form that R can read, how can we extract the day of the week of a date as a word, for e.g. "Monday"? | weekdays() |
| When a date is in the form that R can read, how can we extract the month of a date as a word, for e.g. "January"? | months() |
| When a date variable is in the form that R can read, how can we extract the hour? Why is this different than usual? | datevariable$hour, which is different than usual because usually a data frame comes before $. So here we could have for example mvt$Hour = mvt$Date$hour |
| What would we do if we wanted to make a plot that tells us how many thefts occur on each day of the week from mvt (so Weekday along the x-axis and # of thefts along the y-axis)? Why? | WeekdayThefts = as.data.frame(table(mvt$Weekday)); this is because ggplot requires a data frame as an argument and in mvt each observation corresponds to a theft. |
| What happens if the x-variable in ggplot is a factor variable and we use geom\_line? How can we modify this and what does it do? | For each value along the x-axis it creates a vertical line by joining the points for that factor variable in the order they appear in the data frame (so ultimately a line joining its max y to its min y). We can modify this by adding aes(group=1) which will keep these lines but also adjacent vertical lines will be joined by a line joining the last observation of one factor to the first of the next. |
| What else might we use group for? | geom\_polygon(), for example if we have statesMap loaded from Google, which is a data frame from which we build a map of USA, one of the variables is group, so we would use group=group, which would tell ggmap to group data by each state, so that when one polygon (state) is done being drawn (by joining points in a given group in the order that they appear in the data frame) the program can "lift up its pencil" and move on to drawing the next state without joining it to the previous one. |
| If we create as.data.frame(table(df$x)), what are the variables of this data frame? How many observations are there? | * factor Var1, with levels equal to each of the values of x in df * Freq, which is the frequency of each value of x in df, as shown in the table * 1 observation for each level |
| If we create as.data.frame(table(df$x, df$y)), what are the variables of this data frame? What is one observation? | * factor Var1, with levels equal to each the values of x in df * factor Var2, with levels equal to each of the values of y in df * (So in other words, the rows and column headers from the table are the values of the factors) * Freq, which is the frequencies from the table * One observation corresponds to one value in the table, i.e. one (x,y) pair. |
| Why would we want to do the above for two variables? | * Suppose we want to make a line plot of the frequency of one but separated by the other variable, e.g. hours along the x-axis and Freq of thefts along the y-axis, separated by each day of the week. |
| What particular arguments would we need to do the above if we were plotting Var1 vs. Freq separated by Var2? | * aes(group=Var2) or even better aes(color=Var2) |
| How are factor levels ordered by default? How do we change it? | The levels are ordered alphabetically. We can change this using WeekdayThefts$Var1 = factor(WeekdayThefts$Var1, ordered=TRUE, levels = c("Monday", "Tuesday", ..., "Sunday") |
| How do we convert a variable x of type factor but which consists of numbers to numerical? | x = as.numeric(as.character(x)) |
| Why would we want to? | When creating plots with Freq along the y-axis and we want our x-axis to be understood as numbers or else they are treated as factors (and for example if we use geom\_line() then the lines may be drawn according to each factor as described above, at least if we forget group=1). |
| What types of geometric object is group for that we have seen so far? | geom\_line() or geom\_polygon() |
| If we have Var1 for day of the week, Var2 for hour, Freq for number of thefts in data frame df how might we create a heat map? | ggplot(df, aes(x=Var2, y=Var1)) + geom\_tile(aes(fill=Freq)) |
| How do we get rid of say the y-axis label for ggplot? | ylab(NULL) |
| How do we add a title name to the gradient legend in a heat map? | scale\_fill\_gradient(name="...") |
| How do we change the extreme colors used in a heat map? | scale\_fill\_gradient(low="...", high="...") |
| What are common colors for heat mapping for crime? | white for low, red for high |
| Quickly summarize what 3 general plots we made so far. | * First we plotted Weekday vs. Thefts (line graph) * Then Hours vs. Thefts, grouped by Weekday (so 7 line graphs on the same plot) * Then Hour vs. Weekday heatmap with Thefts (=Freq) as the gradient |
| We can load maps from Google into R and then make plots on top of them. How? | * library(maps) * library(ggmap) * ottawa = get\_map(location="ottawa") * ggmap(ottawa), which displays the map * ggmap(ottawa) acts like a ggplot item and then you can add layers accordingly. |
| What if we want to change the scale of the map? | Add zoom = ... into the definition of ottawa, where ... is a number. |
| What does round(x,3) do? | It rounds numerical variable x to 3 decimal places. |
| What is the syntactical difference between adding data using ggplot and using ggmap | For ggplot the data frame is defined inside ggplot() by simply inputting the data frame name; for ggmap the data frame is defined inside a geom\_() layer and requires data= before the data frame name. |
| After loading a map from Google what will we have as our x and y variables? What do we do after that? | From the data frame that we are trying to layer on top we extract its Longitude variable for x and its Latitude variable for y, and then we plot according to some third variable for example using aes(fill = 3rdvariable) or aes(size=3rdvariable). |
| What if we have too many points in the data frame? | We can group them together by rounding to for example 2 decimal places, counting how many observations there are in each of the squares after Longitude and Latitude have been rounding to 2 decimal places, then create the table and use Var1 and Var2 as the Longitude/Latitude and Freq as the number of observations in each square. |
| E.g. of what that third variable might be? | Freq |
| What is the difference between the color = and fill = argument in aes()? | color for points, fill for tile/bar |
| What is an alternative for tile? What does it do? | alpha=variable in aes() and fill="somecolor" with somecolor inside the geometric object but outside aes(); it is like a gradient but with one color (different shades). |
| \*What must we remember to do when layering data from a data frame on top of a ggmap? | The Longitude and Latitude variables from the data frame need to be numeric. |
| How do we change the specific colors used for color or fill for points/tiles/polygons? What should we remember though? | * For points use scale\_color\_gradient(low="...", high="...") * For tiles and polygons use scale\_fill\_gradient(low="...", high="...") * Remember that color=variable or fill=variable is defined however in aes() of ggplot or geom\_ |
| Where is the variable we are filling/coloring according to defined? | in aes() |
| What is the syntax of merge? What does it do? | * For data frames A and B with common variable name y that we want to merge over we write merge(A,B, by="y") * It creates a new data frame that brings the original two data frames together; the first column is y, then the variables of A (without y) then the variables of B (without y). * If we look at the columns of merge(A,B, by="y") without the A columns then we see we did not invent any new observations; we just have potential duplicates and some left out. * The ones that are left out are the ones corresponding to y-values that occur only in one of A and B. * We have duplicates because for each y-value that exists in both A and B we are creating all possible ways of choosing the correspond A variables' and B variables' values corresponding to that y-value (Cartesian product). |
| How can we draw a map of USA by state? (We can do this for other regions too, such as France for example) | * statesMap = map\_data("state") * ggmap(statesMap, aes(x=long, y=lat, group=group)) + geom\_polygon(color="black", fill="white") |
| What type of object is statesMap? | data frame |
| What does color = do for polygon or bar? fill= inside aes()? fill= outside aes() in geom\_? | Color the borders of the polygons/bars; creates a gradient of colors according to a given variable; creates one fill color that is written. |
| If we have a string/variable that we want to convert to all lowercase, how do we do this? | tolower() |
| What do we do if we want to use a map such as statesMap and then color the states according to data from another data frame? | Remember to merge the two data frames; the state variable has to be written in the same format in both which is why you might need tolower(). |
| \*How do we restrict the values of the variable we are filling/coloring according to, and why would we want to? What happens to values that don't meet the restriction? | in scale\_fill\_gradient use limits=c(0,10) for example; we might want to do this because for example without the limits perhaps some outliers are making the limits too high and therefore coloring everything approximately the same color, so we want to remove them from considering in the scale. If they fall outside the scale then they are grey. |
| If we want to the change the order of variable x according to increasing values of y, how do we do this? Decreasing? | reorder(x,y); reorder(x,-y) |
| What does is.na(df) output if df=data.frame(x,y) | A table with columns is.na(x) and is.na(y) |
| How can we convert all of the NA's in df to 0's? | df[is.na(df)] = 0 |
| Recall if we want to color all points the same color, how do we do this? | Use color= in geom\_point(), no need for aes() |
| What if we want to color the interiors of all polygons or bars (rectangles) the same color, say green? | geom\_polygon(fill="green") or geom\_bar(fill="green") |
| What argument do we need to remember when creating bars? What does it do? | in geom\_bar(stat="identity"); it makes the height of the bars equal to the y-value in ggplot() |
| What if we want to add the y-variable (i.e. height) of each bar? | geom\_text(aes(label = y-variable)) |
| What else can we do? (see p.109 from The Analytics Edge Word document) | * Adjust the angle at which the values on an axis are displayed (for example the country names along the x-axis may be overlapping) * The placement of the text using hjust and vjust. |
| How to create a world map: | worldMap = map\_data("world")  ggplot(worldMap, aes(x=long, y=lat, group=group)) + geom\_polygon() |
| Perhaps redo the section on world map (Unit 7, video 5) |  |
| What does the melt() function do? | In a data frame df if we melt over variable x then this means we are in a sense making our observations as ordered pairs between x and the other variables, rather than having ordered ncol(df)-tuples for observations; we compare x to the other variables pairwise rather than comparing all variables at once. |
| How do we use the melt() function? | * library(reshape2) * m = melt(df, id="x") |
| What type is the output and what does it look like? | * The output is a data frame with 3 variable names in the following order:   x, variable, value   * variable is the name of the current variable from df being paired with x and for this variable we write out all the values of x under x and all of this variables' values under value (in the same order as they appear in df). * We repeat this for each of the variables that aren't x. |
| When might we use melt()? | When we have a data set that consists of many variables but there is one particularly important one that we want to compare to the others just one at a time. For e.g. the melted variable might be our independent variable in ggplot() and all the others are dependent variables which we can extract from value and separate by group=variable. |
| An example of what x in melt might be, and then the rest of the variables? | x might be Year and the other variables might be other statistics we want to compare individually to the year, such as MarriedWChildren, Divorced, MarriedWOChildren, etc. |
| What is order(x)? | It outputs the size ranking of each of the values of x, e.g. order(c(8,6,19)) gives 2 1 3 |
| Easy way to remember which arguments belong in aes() and which don't: | The ones which are based on a variable from the data frame need to be inside aes. |
| How do we know if there is a gradient or not? | It seems gradient corresponds to a continuous variable, so the geometric object could be point, polygon, bar for example; if a factor then there is no gradient but rather distinct colors for each factor. Also, the gradient would need the color/fill to be defined from a variable, not a constant like "green". |
| What if we want to use a legend rather than a gradient? How do we control what is displayed on the legend? | * In scale\_fill\_gradient write legend="guide" * If we want to display 5 items on our legend which originally ranges from 0 to 20 we could write breaks=c(0,5,10,15,20), labels=c("a","b","c","d","e") in scale\_fill\_gradient; so at 0 on the guide there would be a, at 5 there would be b, etc. |
| Which package do we need when building graphs from graph theory? | library(igraph) |
| How do we build a graph from graph theory? | g=graph.data.frame(edges, FALSE, vertices),  where edges and vertices are data frame, FALSE means the graph is undirected. |
| What can we say about the two data frames? | * The first variable in vertices is understood to be the labels for the vertices and the remaining variables are additional attributes of each vertex. * The edge set consists of two variables, each a list of vertex labels, and each observation corresponds to an edge. |
| How do we look at g? What are some arguments we might add? | plot(g, vertex.size=5, vertex.label=NA) |
| What is some information we can access after we created g? | * V(g); a vector of all the different vertices * E(g); a vector of all the different edges * degree(g); a vector of all the vertex degrees * V(g)$size; a vector of all the sizes of the vertices if they are plotted * V(g)$color; a vector of all the colors of the vertices if they are plotted * V(g)$x where x is any of the other variables defined in the vertex data frame |
| What is a word cloud? | * It is a visualization of words which displays the words in a text with the size of each displayed word dependent on the frequency of the word in the text; more frequent are larger. * The words are either horizontal or perpendicular, can be colored, etc. |
| How do we build a word cloud? | * library(wordcloud) * wordcloud(words, freq, ...) * We may need to preprocess our data using corpus, etc. and when ultimately converted to a data frame df whose column names are the words and whose values in a given column are the frequency of that word over all observations, then we create wordcloud(colnames(df), colSums(df), ...) * We don't necessarily need to go through the steps of building a corpus; the whole purpose of this in the above example is just to reduce our information to a data frame from which we can extract various words and their frequencies. We could for example create wordcloud(c("apple", "banana", "orange"), c(10,5,3)) |
| When preprocessing words to be able to use in a word cloud what should we remember? | We probably don't want to stem the words because we want the words to appear fully in the word cloud. |
| What package does wordcloud also come with? | RColorBrewer |
| How do we look at it? | display.brewer.all() |
| \*How do we use it? | * For our wordcloud argument colors we can write colors=brewer.pal(#diffColorsInChosenPalette, "paletteName") * If we want just a subset of colors from the given palette we can add for example [5:9] after the above argument, which gives colors 5 through 9. |
| How do you create a cell in Excel that looks at the current date? | =TODAY() |
| In the American Airlines example we assumed the demand for both the regular and the discounted tickets were given. In practice how would the demand be determined? | They would be forecasted using analytics. |
| What are the names of the different components of a linear optimization problem? | objective, decision variables, constraints |
| What is the name of the region where an optimal solution point must lie in a linear optimization problem? | The feasible space |
| What is sensitivity analysis? | The analysis of how the solution to a linear optimization model changes when the data / constraints change. |
| What is marginal revenue? | The revenue of selling one item |
| What is shadow price for revenue? In general? | * For revenue it is the added revenue generated by increasing demand by 1 unit. * I think it depends on whether or not the optimization problem is a maximization or minimization as well as whether the constraint on the decision variable is >= (bounded below) or <= (bounded above). * For maximization with the decision variable bounded above it is the increase in the objective when the bound is increased by 1. * For maximization with the decision variable bounded below it is the decrease in the objective when the bound is increase by 1. * For minimization with the decision variable bounded above it is the decrease in the objective when the bound is increased by 1. * For minimization with the decision variable bounded below it is the increase in the objective with the bound is increased by 1. * (not certain about all of this) |
| What is the point of marketing? When is it appropriate? | The point is to increase demand and it is appropriate when supply is greater than demand and the cost of marketing is less than the added revenue that the marketing will generate. |
| What is it called when trying to decide how many tickets of various prices to sell to maximize revenue/profit? | Revenue management |
| What is the technology called where beamlets of different intensities are sent to the bottom to try to eradicate tumors? How are the regions in the area receiving IMRT divided up? | Intensity-modulated radiation therapy; they are divided into voxels, usually 4mmx4mmx4mm |
| What is the objective in the IMRT problem from the course? The decision variables? The constraints? | The objective is to minimize total dosage to the healthy tissues; the decision variables are the intensities of each of the beamlets; the constraints are how much dosage should be sent to the healthy tissue voxels, how much should be sent to the spinal cord voxel and how much should be sent to the tumor voxels, as well as non-negativity. |
| What is the unit of dosage and how is dosage in a voxel calculated? | The unit is gray (Gy) and the dosage in a given voxel is calculated as the sum of the dosages of all beamlets contributing to that voxel. |
| Are intensity and dosage the same? | No, but they are directly proportional; for a given intensity, the dosage changes with the distance the beamlet has to travel and which structures it has to pass through. |
| How can we modify a maximization problem so as to shrink the value of a given decision variable, say x, in the optimal solution? (Similarly with growing the value of x, or with shrinking/growing in minimization) | * We can manipulate our objective function in such a way that the weight of x has decreased. For e.g. if the objective involves +x then we can change this to +x/10. * We can play around with the constraints. |
| Why would we want to do the above modifications? | To explore tradeoffs; maybe we're willing to increase on variable at the expense of decreasing another for example. |
| What are companies doing to try get their ads displayed by Google? | They bid to have their ads displayed on certain queries. |
| What do we in a linear optimization problem if one of our constraints is a fraction of variables? | Multiply both sides of the inequality by the denominator so that it is not a fraction; otherwise it won't work for linear optimization. |
| For a linear optimization problem where for each decision variable there are two pieces of information to take into consideration, with each piece of information having various values, how do we model our decision variables in Excel? | As a table; e.g. for Google AdWords, for the various advertisers along the rows and the various queries along the columns, with the decision variables representing how many ads of each type on each query are displayed by Google; e.g. how many kgs of different types of fabric various manufacturers need to produce, with manufacturers as rows and fabrics as columns. |
| For a given shadow price, what is the name of the most number of units you can change the constraint by while keeping the shadow price unchanged? | The allowable increase |
| What is the difference between integer optimization and the linear optimization we have seen so far? | In integer optimization the decision variables are integers. |
| What is an example of where integer optimization is useful? Specifically, what kind do we use? | Creating schedules; binary optimization |
| How does it work? | Our decision variables each take on a value of either 1 or 0; for example if a given baseball team pair is going to play in a given week then their decision variable is 1, else 0, so we make a table with team pairs along the top and weeks along the bottom. |
| How are binary decision variables incorporated into a binary linear optimization problem? | Usually each decision variable is associated with a specific value and the decision variable decides whether or not that value is included in the particular calculation of the objective; so the binary variable is 1 if true/0 if false; it is associated with if/else. For example for baseball teams if teams A and B play on week 4 then we add +8 to the objective, +0 if not, so we can write this term in the objective as 8x­AB4, where xAB4 is 1 if A and B play on week 4, 0 else. |
| Another example of above | eHarmony; there is a pool of men and women and any man/woman pair may be matched (1 if matched, 0 else). Also each pair has a compatibility score, so the objective is computed by summing the products of the decision variables by their compatibility scores. |
| One nice thing about binary decision variables: | They can be used to write logical constraints such as *and,* *or*, *implies*, etc. |
| What are heuristics? | Solutions that are not necessarily optimal but still good. They are useful in cases where computing the optimal solution is too computationally complex. |
| One main difference between integer optimization in general and binary optimization? | In integer optimization the decision variable doesn't need to be multiplied by another value because its value is contained within the decision variable, e.g. number of operating rooms assigned to department j on day k. Here the decision variable isn't deciding TRUE/FALSE (as to whether or not a value should be added in the objective) but is already the appropriate numerical value. |
| What if we have a binary optimization problem where there is only one piece of information for each decision variable? | We can still make it a table for example by making the columns the two binary options; so each binary option has its own decision variable for each of the value of the piece of information, but now we have the added constraint that row sums must be 1. |
|  |  |
|  |  |