Project 2: Modeling, Testing, and Predicting

SDS348 Fall 2019

Modeling

Instructions

A knitted R Markdown document (as a PDF) and the raw R Markdown file (as .Rmd) should both be submitted to Canvas by 11:59pm on 11/24/2019. These two documents will be graded jointly, so they must be consistent (i.e., don't change the R Markdown file without also updating the knitted document). Knit an html copy too, for later! In the .Rmd file for Project 2, you can copy the first code-chunk into your project .Rmd file to get better formatting. Notice that you can adjust the opts_chunk$set(…) above to set certain parameters if necessary to make the knitting cleaner (you can globally set the size of all plots, etc). I have gone ahead and set a few for you (such as disabling warnings and package-loading messges when knitting)!

Like before, I envision your written text forming something of a narrative structure around your code/output. All results presented must have corresponding code. Any answers/results/plots etc. given without the corresponding R code that generated the result will not be graded. Furthermore, all code contained in your final project document should work properly. Please do not include any extraneous code or code which produces error messages. (Code which produces warnings is acceptable, as long as you understand what the warnings mean).

Find data:

Find one dataset with at least 5 variables that wish to use to build model. At least one should be categorical (with 2-5 groups) and at least two should be numeric. Ideally, one of your variables will be binary (if not, you will need to create one by discretizing a numeric, which is workable but less than ideal). You will need a minimum of 40 observations (*at least* 10 observations for every explanatory variable you have, ideally 20+ observations/variable).

It is perfectly fine to use either dataset (or the merged dataset, or a subset of your variables) from Project 1. However, you could also diversify your portfolio a bit by choosing a different dataset to work with (particularly if the variables did not reveal interesting associations in Project 1). The only requirement/restriction is that you may not use data from any examples we have done in class or lab. It would be a good idea to pick more cohesive data this time around (i.e., variables that you actually thing might have a relationship you would want to test). Think more along the lines of your Biostats project.

Again, you can use data from anywhere you want (see bottom for resources)! If you want a quick way to see whether a built-in (R) dataset has binary and/or character (i.e., categorical) variables, check out this list: https://vincentarelbundock.github.io/Rdatasets/datasets.html (https://vincentarelbundock.github.io/Rdatasets/datasets.html).

Guidelines and Rubric

- **0. (5 pts)** Introduce your dataset and each of your variables (or just your main variables if you have lots) in a paragraph.
- **1. (15 pts)** Perform a MANOVA testing whether any of your numeric variables (or a subset of them, if including them all doesn't make sense) show a mean difference across levels of one of your categorical variables (3). If they do, perform univariate ANOVAs to find response(s) showing a mean difference across groups (3), and perform post-hoc t tests to find which groups differ (3). Discuss the number of tests you have performed, calculate the probability of at least one type I error (if unadjusted), and adjust the significance level accordingly (bonferroni correction) before discussing significant differences (3). Briefly discuss assumptions and whether or not they are likely to have been met (2).
- **2. (10 pts)** Perform some kind of randomization test on your data (that makes sense). This can be anything you want! State null and alternative hypotheses, perform the test, and interpret the results (7). Create a plot visualizing the null distribution and the test statistic (3).
- **3. (35 pts)** Build a linear regression model predicting one of your response variables from at least 2 other variables, including their interaction. Mean-center any numeric variables involved in the interaction.
    - Interpret the coefficient estimates (do not discuss significance) (10)
    - Plot the regression using `ggplot()` . If your interaction is numeric by numeric, refer to code near the end of WS15 to make the plot. If you have 3 or more predictors, just chose two to plot for convenience. (7)
    - Check assumptions of linearity, normality, and homoskedasticity either graphically or using a hypothesis test (3)
    - Regardless, recompute regression results with robust standard errors via `coeftest(..., vcov=vcovHC(...))` . Discuss significance of results, including any

    changes from before/after robust SEs if applicable. (7)
    - What proportion of the variation in the outcome does your model explain? (3)
    - Finally, rerun the regression but without interactions (just main effects); compare this with the interaction model and the null model using a likelihood ratio test (4)
- **4. (5 pts)** Rerun same regression model (with interaction), but this time compute bootstrapped standard errors. Discuss any changes you observe in SEs and p-values using these SEs compared to the original SEs and the robust SEs)

- **5. (40 pts)** Perform a logistic regression predicting a binary categorical variable (if you don't have one, make/get one) from at least two explanatory variables (interaction not necessary).

    - Interpret coefficient estimates in context (10)
    - Report a confusion matrix for your logistic regression (2)
    - Compute and discuss the Accuracy, Sensitivity (TPR), Specificity (TNR), and Recall (PPV) of your model (5)
    - Using ggplot, plot density of log-odds (logit) by your binary outcome variable (3)
    - Generate an ROC curve (plot) and calculate AUC (either manually or with a package); interpret (10)
    - Perform 10-fold (or repeated random sub-sampling) CV and report average out-of-sample Accuracy, Sensitivity, and Recall (10)
- **6. (10 pts)** Choose one variable you want to predict (can be one you used from before; either binary or continuous) and run a LASSO regression inputting all the rest of your variables as predictors. Choose lambda to give the simplest model whose accuracy is near that of the best (i.e., `lambda.1se`). Discuss which variables are retained. Perform 10-fold CV using this model: if response in binary, compare model's out-of-sample accuracy to that of your logistic regression in part 5; if response is numeric, compare the residual standard error (at the bottom of the summary output, aka RMSE): lower is better fit!

Where do I find data again?

You can choose ANY datasets you want that meet the above criteria for variables and observations. You can make it as serious as you want, or not, but keep in mind that you will be incorporating this project into a portfolio webpage for your final in this course, so choose something that really reflects who you are, or something that you feel will advance you in the direction you hope to move career-wise, or something that you think is really neat, or whatever. On the flip side, regardless of what you pick, you will be performing all the same tasks, so it doesn't end up being that big of a deal.

If you are totally clueless and have no direction at all, log into the server and type

```
data(package = .packages(all.available = TRUE))
```

This will print out a list of **ALL datasets in ALL packages** installed on the server (a ton)! Scroll until your eyes bleed! Actually, do not scroll that much… To start with something more manageable, just run the command on your own computer, or just run `data()` to bring up the datasets in your current environment. To read more about a dataset, do `?packagename::datasetname` .

If it is easier for you, and in case you don't have many packages installed, a list of R datasets from a few common packages (also downloadable in CSV format) is given at the following website: https://vincentarelbundock.github.io/Rdatasets/datasets.html (https://vincentarelbundock.github.io/Rdatasets/datasets.html).

- A good package to download for fun/relevant data is `fivethiryeight` . Run `install.packages("fivethirtyeight"),` load the packages with `library(fivethirtyeight)` , run `data()` , and then scroll down to view the datasets. Here is an online list of all 127 datasets (with links to the 538 articles). Lots of sports, politics, current events, etc.

- If you have already started to specialize (e.g., ecology, epidemiology) you might look at discipline-specific R packages (vegan, epi, respectively). We will be using some tools from these packages later in the course, but they come with lots of data too, which you can explore according to the directions above

- However, you *emphatically DO NOT* have to use datasets available via R packages! In fact, I would much prefer it if you found the data from completely separate sources and brought them together (a much more realistic experience in the real world)! You can even reuse data from your SDS328M project, provided it shares a variable in common with other data which allows you to merge the two together (e.g., if you still had the timestamp, you could look up the weather that day: https://www.wunderground.com/history/ (https://www.wunderground.com/history/)). If you work in a research lab or have access to old data, you could potentially merge it with new data from your lab!

- Here is a curated list of interesting datasets (read-only spreadsheet format): https://docs.google.com/spreadsheets/d/1wZhPLMCHKJvwOkP4juclhjFgqIY8fQFMemwKL2c64vk/edit (https://docs.google.com/spreadsheets/d/1wZhPLMCHKJvwOkP4juclhjFgqIY8fQFMemwKL2c64vk/edit)

- Here is another great compilation of datasets: https://github.com/rfordatascience/tidytuesday (https://github.com/rfordatascience/tidytuesday)

- Here is the UCI Machine Learning Repository: https://archive.ics.uci.edu/ml/index.php (https://archive.ics.uci.edu/ml/index.php)
    - See also https://en.wikipedia.org/wiki/List_of_datasets_for_machine-learning_research#Biological_data (https://en.wikipedia.org/wiki/List_of_datasets_for_machine-learning_research#Biological_data)
- Here is another good general place to look: https://www.kaggle.com/datasets (https://www.kaggle.com/datasets)

- To help narrow your search down or to see interesting variable ideas, check out https://www.tylervigen.com/spurious-correlations (https://www.tylervigen.com/spurious-correlations). This is the spurious correlations website, and it is fun, but if you look at the bottom of each plot you will see sources for the data. This is a good place to find very general data (or at least get a sense of where you can scrape data together from)!

- If you are interested in medical data, check out www.countyhealthrankings.org

- If you are interested in scraping UT data, they make *loads* of data public (e.g., beyond just professor CVs and syllabi). Check out all the data that is available in the statistical handbooks: https://reports.utexas.edu/statistical-handbook (https://reports.utexas.edu/statistical-handbook)

Broader data sources:

Data.gov (www.data.gov) 186,000+ datasets!

Social Explorer (Social%20Explorer) is a nice interface to Census and American Community Survey data (more user-friendly than the government sites). May need to sign up for a free trial.

U.S. Bureau of Labor Statistics (www.bls.gov)

U.S. Census Bureau (www.census.gov)

Gapminder (www.gapminder.org/data), data about the world.

0. Introduction

My dataset is part of the DAAG package. It is called leafshape and includes information about shapes of leaves in several regions. There are a lot of variables, so I will be looking at bladelen (leaf length), petiole and bladewid (leaf width) for my numeric variables, location for my categorical variable and arch (leaf architecture, 0 being plagiotropic and 1 being orthotropic) for my binary variable. Latitude is numeric, but is more categorical as well (i.e. not continuous). The other variables are log quantities of the other variables.

1. MANOVA

```
library(DAAG)
library(tidyverse)
library(ggplot2)
leaf <- leafshape
leaf %>% head
```

```
## bladelen petiole bladewid latitude logwid logpet loglen
arch location
## 1 33.88 1.402632 13.65 5 2.613740 0.33835047 3.522825 0
Sabah
## 2 33.32 1.016260 10.26 5 2.328253 0.01612922 3.506158 0
Sabah
## 3 29.35 2.392025 12.21 5 2.502255 0.87214029 3.379293 0
Sabah
## 4 26.87 0.808787 8.70 5 2.163323 -0.21221968 3.291010 0
Sabah
## 5 26.67 0.802767 8.41 5 2.129421 -0.21969077 3.283539 0
Sabah
## 6 24.23 1.490145 7.70 5 2.041220 0.39887343 3.187592 0
Sabah
```

```
manov <- manova(cbind(bladelen,petiole) ~ location, data= leaf)
summary(manov)
```

```
## Df Pillai approx F num Df den Df Pr(>F)
## location 5 0.22203 6.9931 10 560 2.222e-10 ***
## Residuals 280
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1
```

```
aov1 <- aov(bladelen~location, data=leaf)
aov2 <- aov(petiole~location, data=leaf)
summary(aov1)
```

```
## Df Sum Sq Mean Sq F value Pr(>F)
## location 5 4428 885.6 12.79 3.3e-11 ***
## Residuals 280 19388 69.2
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1
```

```
summary(aov2)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## location      5    210   41.98    1.53  0.181
## Residuals   280   7684   27.44
```

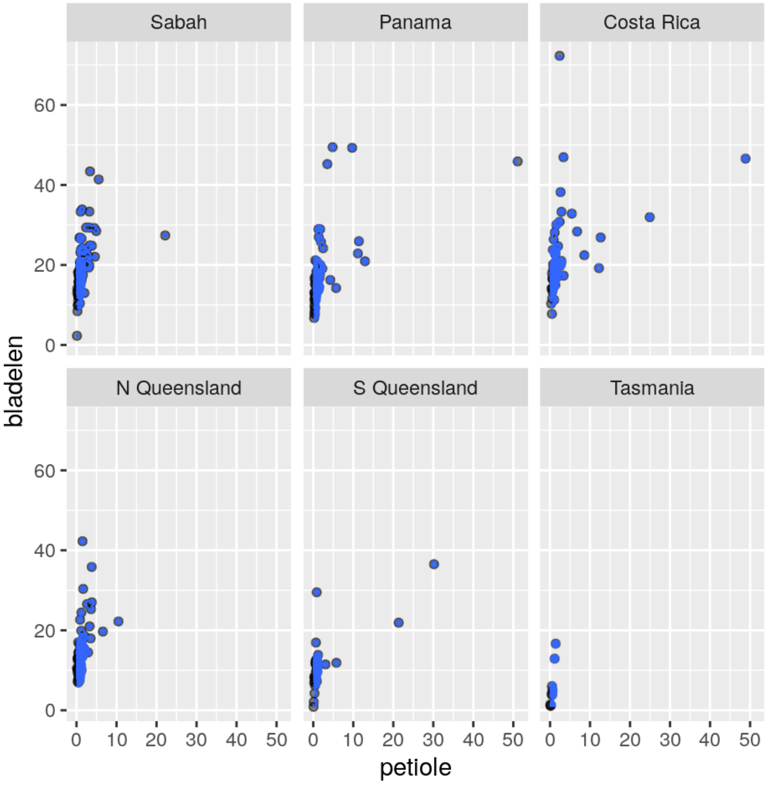
```
pairwise.t.test(leaf$bladelen,leaf$location, p.adj="none")
```

```
##
## Pairwise comparisons using t tests with pooled SD
##
## data: leaf$bladelen and leaf$location
##
## Sabah Panama Costa Rica N Queensland S Queensland
## Panama 0.4642 - - - -
## Costa Rica 0.0383 0.0105 - - -
## N Queensland 0.0021 0.0330 3.6e-06 - -
## S Queensland 6.1e-06 0.0002 1.0e-08 0.0430 -
## Tasmania 4.0e-06 3.0e-05 5.1e-08 0.0018 0.0745
##
## P value adjustment method: none
```

```
pairwise.t.test(leaf$petiole,leaf$location, p.adj="none")
```

```
##
## Pairwise comparisons using t tests with pooled SD
##
## data: leaf$petiole and leaf$location
##
## Sabah Panama Costa Rica N Queensland S Queensland
## Panama 0.202 - - - -
## Costa Rica 0.040 0.451 - - -
## N Queensland 0.778 0.145 0.029 - -
## S Queensland 0.454 0.772 0.353 0.349 -
## Tasmania 0.543 0.224 0.107 0.642 0.325
##
## P value adjustment method: none
```

```
# Assumptions
ggplot(leaf, aes(x = petiole, y = bladelen)) +
 geom_point(alpha = .5) + geom_density_2d(h=2) + coord_fixed() + facet_wrap(~lo
cation)
```



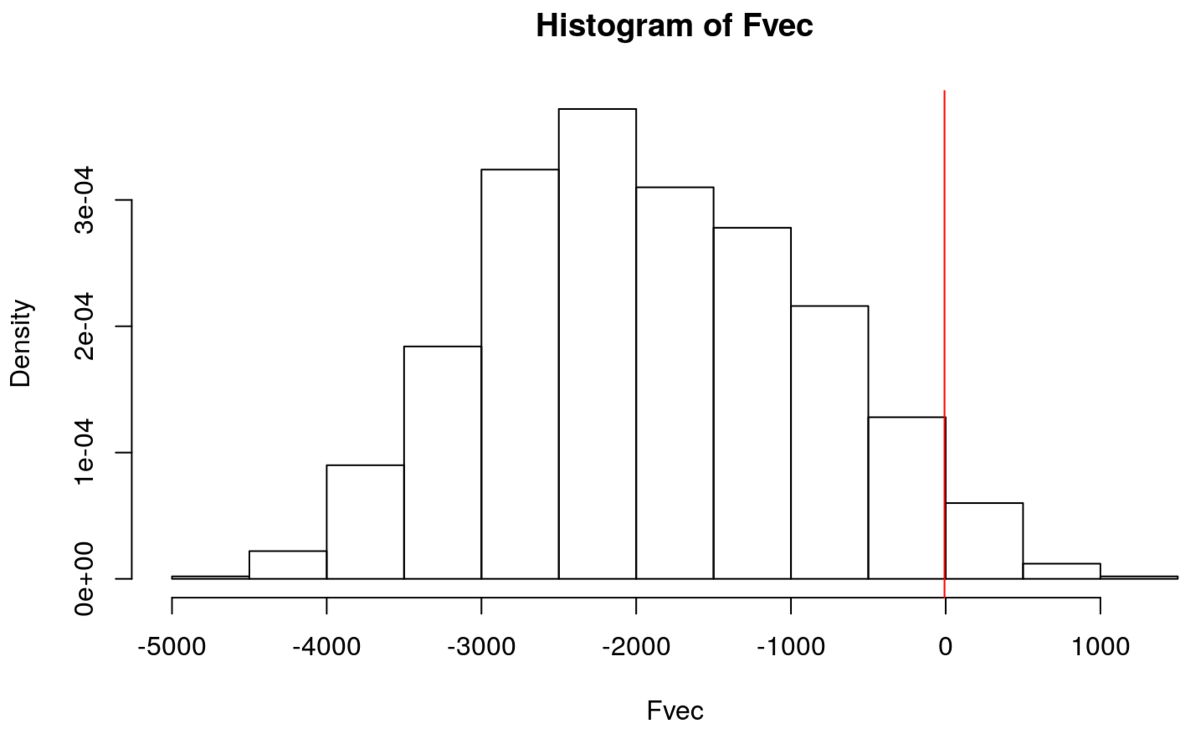
A Multi-variate ANOVA test confirms that the test is significant. Therefore, at least one of the groups has a sinificantly different mean to the other when it comes to leaf length and petiole length. Significant differences were found on these two independent measures with a Pilai trace of 0.22. The approximate F statistic was 7.0 with (10,560) degrees of freedom and a p-statistic on the order of 10^-10. Univariate ANOVAs were run for both dependent variables. It was found that leaf length was significantly different across means between the locations, but that petiole length was not significantly different in means across the locations. The Bonferroni correction is 0.05/3 = 0.016. However, the results are still significant for the MANOVA and the ANOVA for leaf width only. In the post-hoc test, it is clear that Sabah and Costa rica as well as Costa Rica and S Queensland differ in petiole size. However, none of the groups are significant even with the previous Bonferroni correction. Leaf length differs across the following groups: Sabah and S Queensland, Sabah and Tasmania, Panama and S Queensland, Panama and Tasmania, Costa Rica and all locations but Panama and Sabah, Tasmania and N Queensland. This is true even with a Bonferroni correction of 0.05/9 = 0.00556. In terms of assumptions, we are assuming multivariate normality of dependent variables, independent samples and observations, homogeneity between groups, linear relationships across response variables, no outliers and no multi-correlation between the variables (i.e. if we included the log results). Looking at the graph, only Costa Rica seems to violate normality. It is hard to meet these assumptions with the data, though, and could very well explain why Costa Rica differed across so many groups.

2. Randomization Test

```
# Adonis method
library(vegan)
distances <- leaf %>% select(bladelen, petiole) %>% dist()
adonis(distances~location, data=leaf)
```

```
##
## Call:
## adonis(formula = distances ~ location, data = leaf)
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
## Df SumsOfSqs MeanSqs F.Model R2 Pr(>F)
## location 5 4638 927.55 9.5935 0.14626 0.001 ***
## Residuals 280 27072 96.69 0.85374
## Total 285 31710 1.00000
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1
```

```
# By hand
size <- leaf %>% summarize(n = n())
SST <- sum(distances^2)/size

SSW <- leaf %>% group_by(location) %>% select(bladelen, petiole) %>% do(d=dist
(.[1:2],"euclidean"))%>%ungroup()%>%
 summarize(sum(d[[1]]^2)/48 + sum(d[[2]]^2)/48 + sum(d[[3]]^2)/48)%>%pull
#There are 286/6 = 47.6667 or 48 observations per location

F_obs<-((SST-SSW)/5)/(SSW/280)

# Null distribution
Fs<-replicate(1000,{
new<-leaf%>%mutate(location=sample(location))
SSW<-new%>%group_by(location)%>%select(bladelen, petiole)%>%
 do(d=dist(.[1:2],"euclidean"))%>%ungroup()%>%
 summarize(sum(d[[1]]^2)/48 + sum(d[[2]]^2)/48 + sum(d[[3]]^2)/48)%>%pull
((SST-SSW)/5)
})

Fvec <- vector()
for (i in 1:1000){
  Fvec[i] <- Fs[i]$n
}

{hist(Fvec,prob = T); abline(v=F_obs, col="red", add=T)}
```



```
mean(Fvec > F_obs$n)
```

```
## [1] 0.037
```

```
mean(Fvec > F_obs$n) == mean(Fs > F_obs$n)
```

```
## [1] TRUE
```

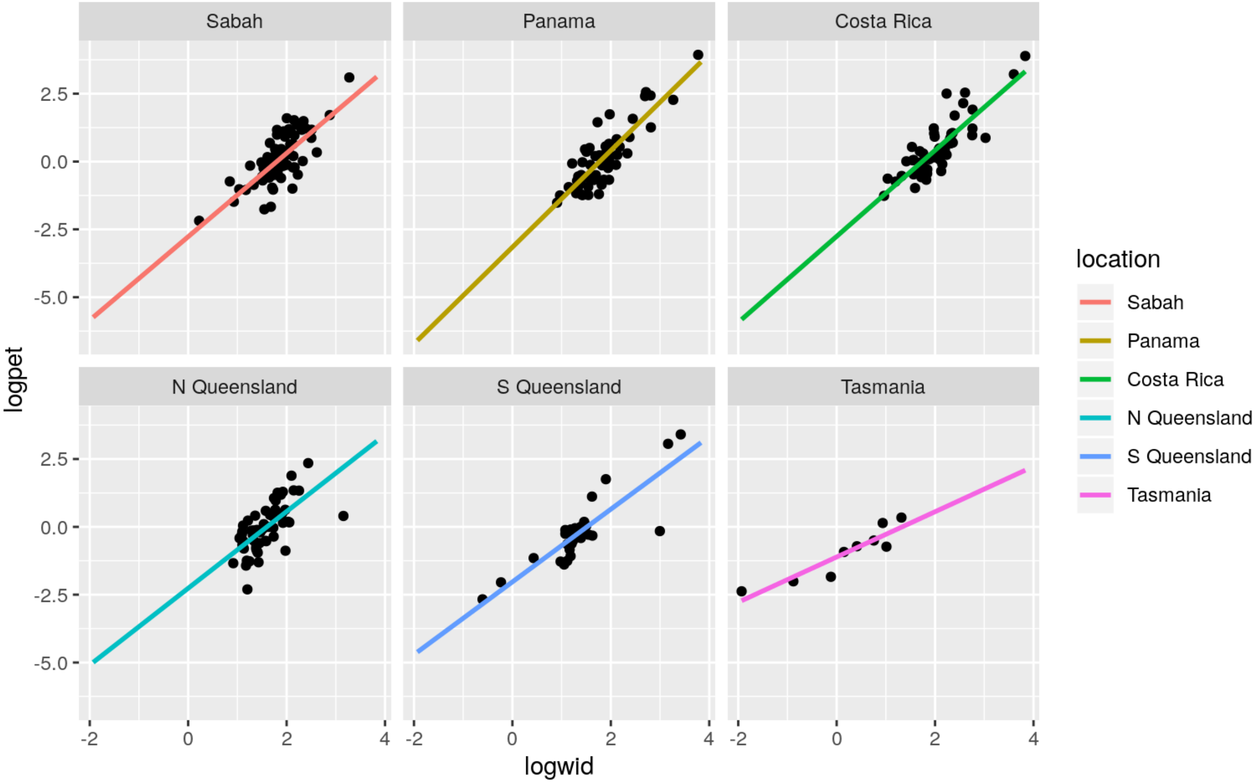
The Null Hypothesis states that there is no significant difference between the means of any of the locations in relation to leaf width and petiole size. The alternate hypothesis states that there is at least one statistically significant difference in means in relation to leaf width and petiole size. The test showed a significant p-value, implying that, for this randomized test, there was at least one significant difference in means between the locations. Therefore, the null hypothesis is rejected in favor of the alternate hypothesis. The histogram displays the null distribution with the simulated F statistics. The mean at the end displays the simulated p-value from the PERMANOVA. The p-value is about 0.05 or lower, which is close to significant. The p-value will depend on chance, however. In the histogram, the red line displays the observed F-satistic. For values above the F-statistic, the results are not significantly different. Since less than 5% fall above that line, the results are overall significant.

3. Linear Regression

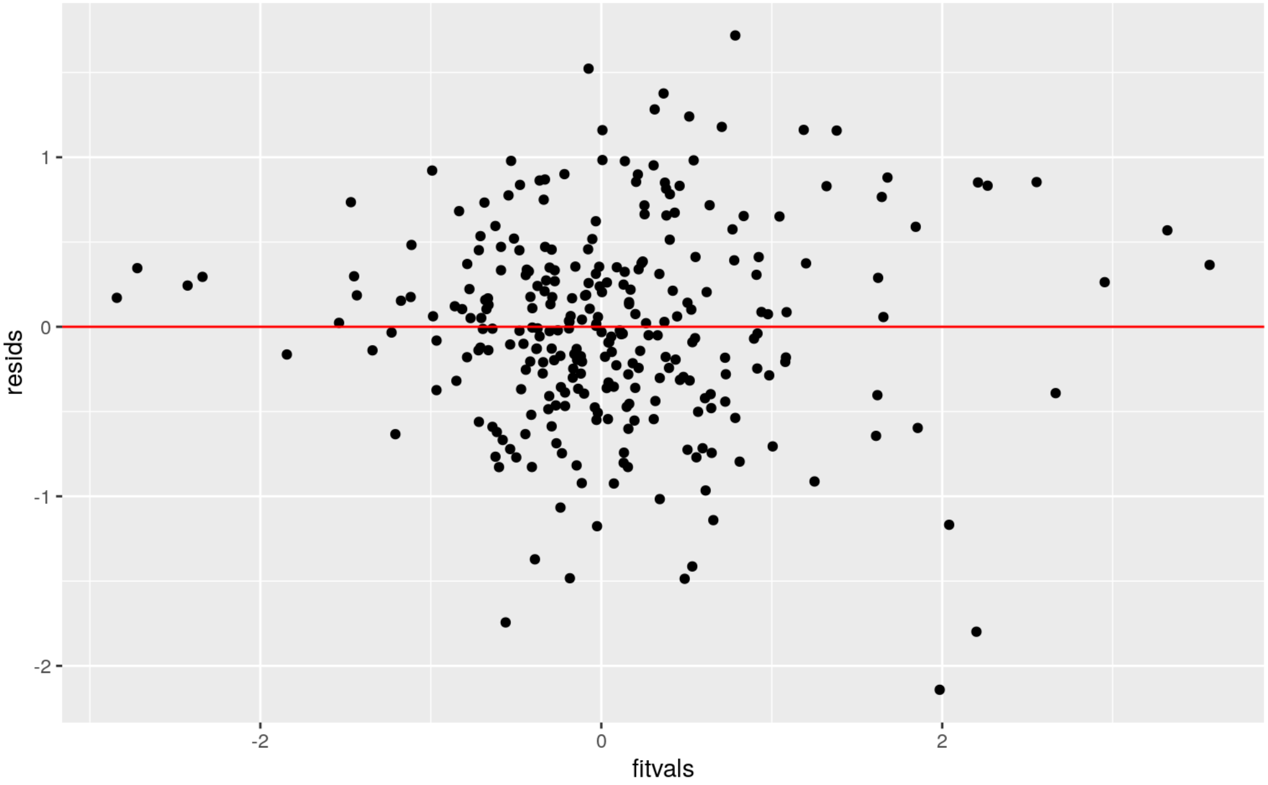
```
pet <- scale(leaf$petiole)
wid <- scale(leaf$bladewid)
model <- lm(pet~wid*leaf$location, data=leaf)
summary(model)
```

```
##
## Call:
## lm(formula = pet ~ wid * leaf$location, data = leaf)
##
## Residuals:
## Min 1Q Median 3Q Max
## -2.65830 -0.12183 -0.01945 0.15909 2.27594
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.12156 0.05243 -2.318 0.021159 *
## wid 0.65908 0.08662 7.609 4.46e-13 ***
## leaf$locationPanama 0.09515 0.08243 1.154 0.249390
## leaf$locationCosta Rica -0.06373 0.08752 -0.728 0.467107
## leaf$locationN Queensland 0.04108 0.08545 0.481 0.631031
## leaf$locationS Queensland 0.38306 0.10058 3.809 0.000173
***
## leaf$locationTasmania 0.13553 0.73476 0.184 0.853794
## wid:leaf$locationPanama 0.39633 0.10215 3.880 0.000131
***
## wid:leaf$locationCosta Rica 0.29569 0.09930 2.978
0.003162 **
## wid:leaf$locationN Queensland -0.38665 0.14147 -2.733
0.006684 **
## wid:leaf$locationS Queensland 0.23394 0.11199 2.089
0.037640 *
## wid:leaf$locationTasmania -0.29711 0.77837 -0.382
0.702972
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1
##
## Residual standard error: 0.4687 on 274 degrees of
freedom
## Multiple R-squared: 0.7888, Adjusted R-squared: 0.7804
## F-statistic: 93.05 on 11 and 274 DF, p-value: < 2.2e-16
```

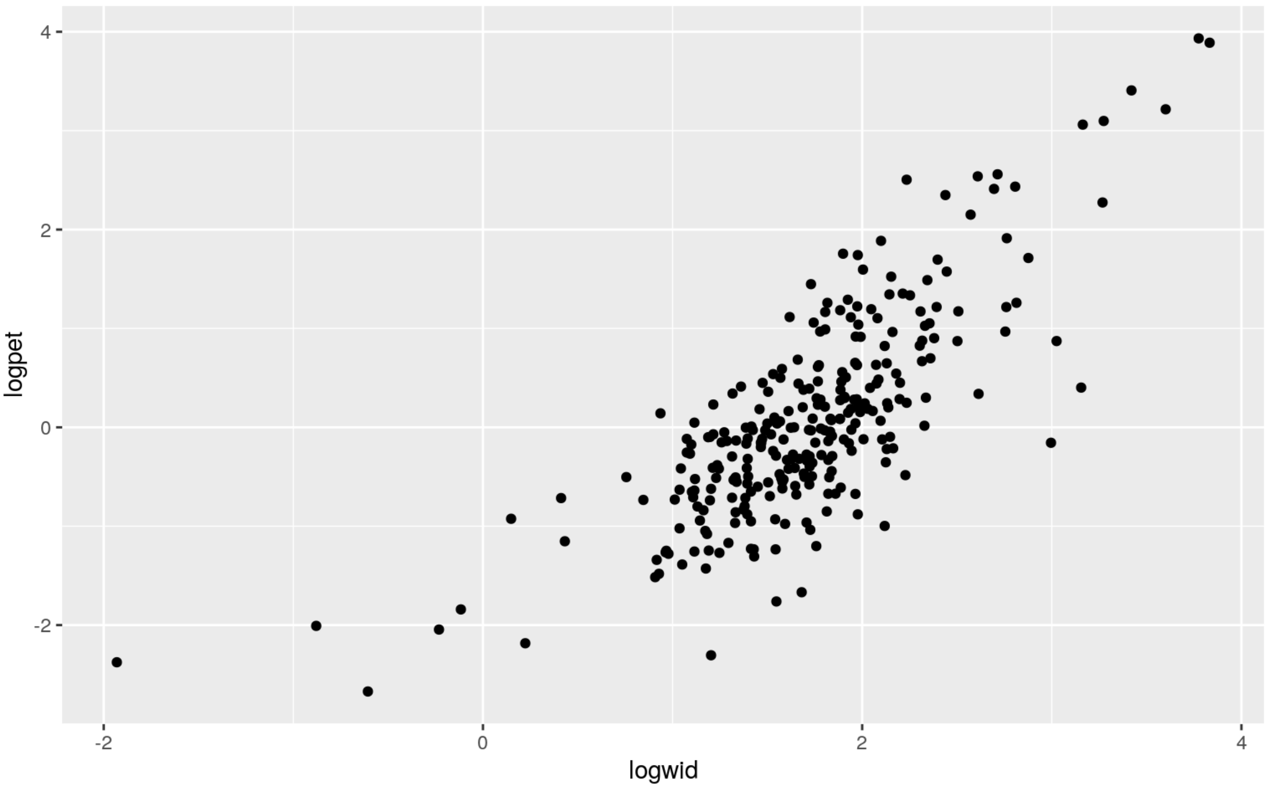
```
ggplot(leaf, aes(x=logwid, y=logpet,group=location))+geom_point()+geom_smooth(m
ethod="lm",se=F,fullrange=T,aes(color=location))+facet_wrap(~location)
```



```
model1 <- lm(logpet~logwid*location, data=leaf)

resids<-model1$residuals
fitvals<-model1$fitted.values
ggplot()+geom_point(aes(fitvals,resids))+geom_hline(yintercept=0, col="red")
```



```
ggplot(leaf,aes(logwid,logpet))+geom_point()
```



```
library(broom)
library(lmtest)
library(sandwich)
coeftest(model1, vcov=vcovHC(model1))
```

```
##
## t test of coefficients:
##
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.770160 0.287966 -9.6198 < 2.2e-16 ***
## logwid 1.538397 0.160418 9.5899 < 2.2e-16 ***
## locationPanama -0.386679 0.380990 -1.0149 0.31103
## locationCosta Rica 0.012214 0.417380 0.0293 0.97668
## locationN Queensland 0.506922 0.755336 0.6711 0.50271
## locationS Queensland 0.741431 0.419756 1.7663 0.07845 .
## locationTasmania 1.659417 0.346439 4.7899 2.736e-06 ***
## logwid:locationPanama 0.243343 0.209034 1.1641 0.24538
## logwid:locationCosta Rica 0.047717 0.223271 0.2137
0.83092
## logwid:locationN Queensland -0.123527 0.494705 -0.2497
0.80301
## logwid:locationS Queensland -0.198547 0.333309 -0.5957
0.55188
## logwid:locationTasmania -0.704052 0.299207 -2.3531
0.01933 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1
```

```
model2 <- lm(logpet ~ logwid + location, data=leaf)
summary(model2)
```

```
##
## Call:
## lm(formula = logpet ~ logwid + location, data = leaf)
##
## Residuals:
## Min 1Q Median 3Q Max
## -2.34403 -0.35400 -0.02842 0.36602 1.74030
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.62979 0.14191 -18.532 < 2e-16 ***
## logwid 1.46234 0.06737 21.705 < 2e-16 ***
## locationPanama 0.05892 0.10713 0.550 0.58273
## locationCosta Rica 0.12646 0.11117 1.138 0.25629
## locationN Queensland 0.29191 0.10557 2.765 0.00607 **
## locationS Queensland 0.43697 0.13380 3.266 0.00123 **
## locationTasmania 1.40365 0.24243 5.790 1.89e-08 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1
##
## Residual standard error: 0.6116 on 279 degrees of
freedom
## Multiple R-squared: 0.6558, Adjusted R-squared: 0.6484
## F-statistic: 88.6 on 6 and 279 DF, p-value: < 2.2e-16
```

```
summary(model1)
```

```
##
## Call:
## lm(formula = logpet ~ logwid * location, data = leaf)
##
## Residuals:
## Min 1Q Median 3Q Max
## -2.14117 -0.35930 -0.00958 0.34421 1.71837
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.77016 0.30192 -9.175 < 2e-16 ***
## logwid 1.53840 0.15949 9.645 < 2e-16 ***
## locationPanama -0.38668 0.41553 -0.931 0.35290
## locationCosta Rica 0.01221 0.44465 0.027 0.97811
## locationN Queensland 0.50692 0.43737 1.159 0.24745
## locationS Queensland 0.74143 0.37067 2.000 0.04646 *
## locationTasmania 1.65942 0.36413 4.557 7.81e-06 ***
## logwid:locationPanama 0.24334 0.21843 1.114 0.26623
## logwid:locationCosta Rica 0.04772 0.22117 0.216 0.82934
## logwid:locationN Queensland -0.12353 0.25209 -0.490
0.62452
## logwid:locationS Queensland -0.19855 0.21150 -0.939
0.34869
## logwid:locationTasmania -0.70405 0.25939 -2.714 0.00706
**
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1
##
## Residual standard error: 0.6002 on 274 degrees of
freedom
## Multiple R-squared: 0.6745, Adjusted R-squared: 0.6614
## F-statistic: 51.61 on 11 and 274 DF, p-value: < 2.2e-16
```

```
lrtest(model1, model2)
```

```
## Likelihood ratio test
##
## Model 1: logpet ~ logwid * location
## Model 2: logpet ~ logwid + location
## #Df LogLik Df Chisq Pr(>Chisq)
## 1 13 -253.66
## 2 8 -261.64 -5 15.957 0.006967 **
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1
```

Coefficient estimates suggest that petiole size is smaller for regions with negative coefficient estimates, such as Costa Rica, and considering the interaction between leaf width and N Queensland or Tasmania. Leaf petiole is proportional to leaf width, which is to be expected, but is even so when the values are mean centered, suggesting that leaves with higher widths have proportionally bigger petioles. The control group for location is Sabah, although the intercept does not make sense to interpret since we would have to set the leaf width to zero. if leaf width is not the mean, or the scaled width is not 0, then the control leaf petiole is given by 0.66(width) - 0.12, showing that the petiole size increases with width but is lower overall for Sabah.

The plot made is difficult to interpret because the magnitude of the change in petiole size is small with the change in leaf width size, so the lines appear to be horizontal. In fact, many of the points are centered around the same size of leaf petiole and width, suggesting that outliers may also pose a problem in displaying the correlation. However, with logpet and logwid, the results appear mroe linear. As such, we can confirm that the logs of petiole and width are linear. Rerunning the model with the log values, we can see from the residual plot that the linearity assumption is met. The other graph shows no fanning pattern, so homeoskadicity is met for the log-scaled values.

Using robust standard errors from the coeftest, as well as the log-scaled values from before, since they meet the assumptions made, we can see that width is still a significant predictor of petiole, with locations Panama and Tasmania having significance as well. Furthermore, one interaction, that between Tasmania and width, is significant as well.

The proportion of variance in petiole explained by the log-scaled model is 0.67.

It seems that, from the likelyhood ratio test, the model with just main effects has a higher likelyhood to be different to the null hypothesis. That is, without interactions, the model fits the data better and better explains the correlation between variblaes.

4. Bootstrapped SE

```
samp_distn<-replicate(1000, {
 boot_dat<-leaf[sample(nrow(leaf),replace=TRUE),]
 bootmodel<-lm(logpet~logwid*location,data=boot_dat)
 coef(bootmodel)
})

samp_distn%>%t%>%as.data.frame%>%summarize_all(sd)
```

```
## (Intercept) logwid locationPanama locationCosta Rica
locationN Queensland locationS Queensland
## 1 0.2895589 0.157706 0.3788699 0.4091063 0.5882355
0.4388352
## locationTasmania logwid:locationPanama
logwid:locationCosta Rica logwid:locationN Queensland
## 1 0.3497429 0.2083349 0.2159856 0.3774302
## logwid:locationS Queensland logwid:locationTasmania
## 1 0.329302 NA
```

The bootstrapped SE's are overall much lower than the normal-test standard errors. However, it may not be worth to list them all due to the randomness of the sampling. The boostrap SE's are lower still than the robust SE's from the coeftest.

5. Logistic Regression

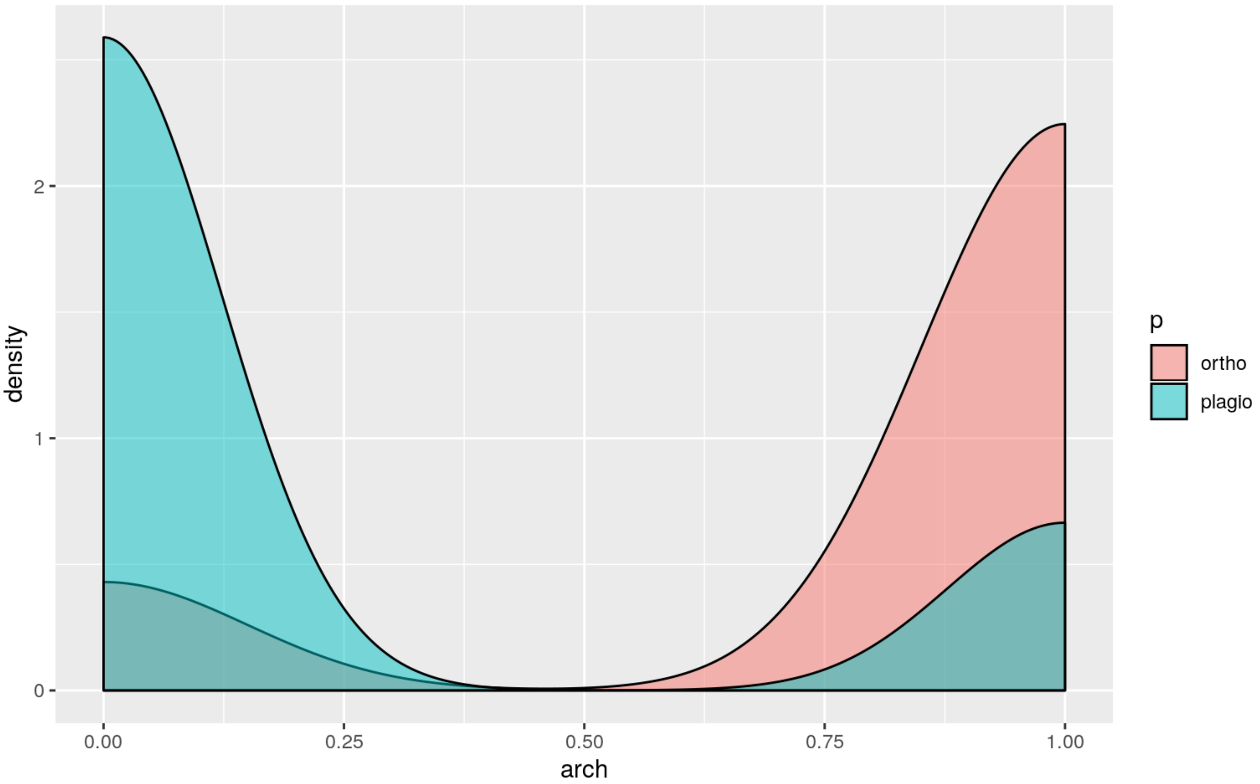
```
logmodel <- glm(arch~loglen+logpet+logwid, data=leaf, family=binomial(link="log
it"))
summary(logmodel)
```

```
##
## Call:
## glm(formula = arch ~ loglen + logpet + logwid, family =
binomial(link = "logit"),
## data = leaf)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -1.7664 -0.6689 -0.3567 0.5015 3.2695
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) 2.0187 1.2262 1.646 0.0997 .
## loglen 0.8529 0.7003 1.218 0.2232
## logpet 2.8693 0.3710 7.735 1.04e-14 ***
## logwid -3.2705 0.7502 -4.359 1.30e-05 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 362.21 on 285 degrees of freedom
## Residual deviance: 239.08 on 282 degrees of freedom
## AIC: 247.08
##
## Number of Fisher Scoring iterations: 5
```

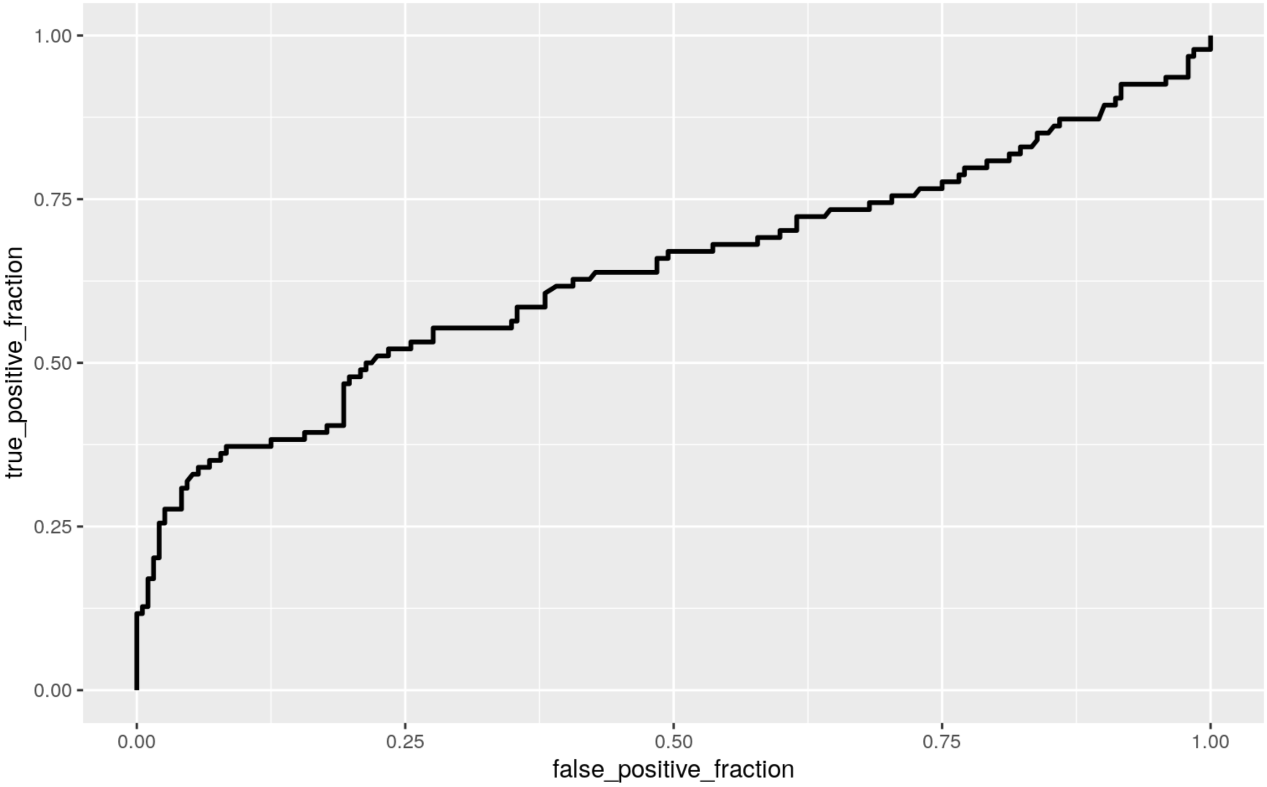
```
predicted <- predict(logmodel, type = "response", leaf)
predicted1 <- predict(logmodel, leaf)
predicted_error <- ifelse(predicted > 0.5, "ortho", "plagio")
p <- ifelse(predicted1 > 0.5, "ortho", "plagio")


table(truth = leaf$arch, prediction = predicted_error) %>% addmargins
```

```
##      prediction
## truth ortho plagio Sum
##   0      12    180 192
##   1      60     34  94
##   Sum    72    214 286
```

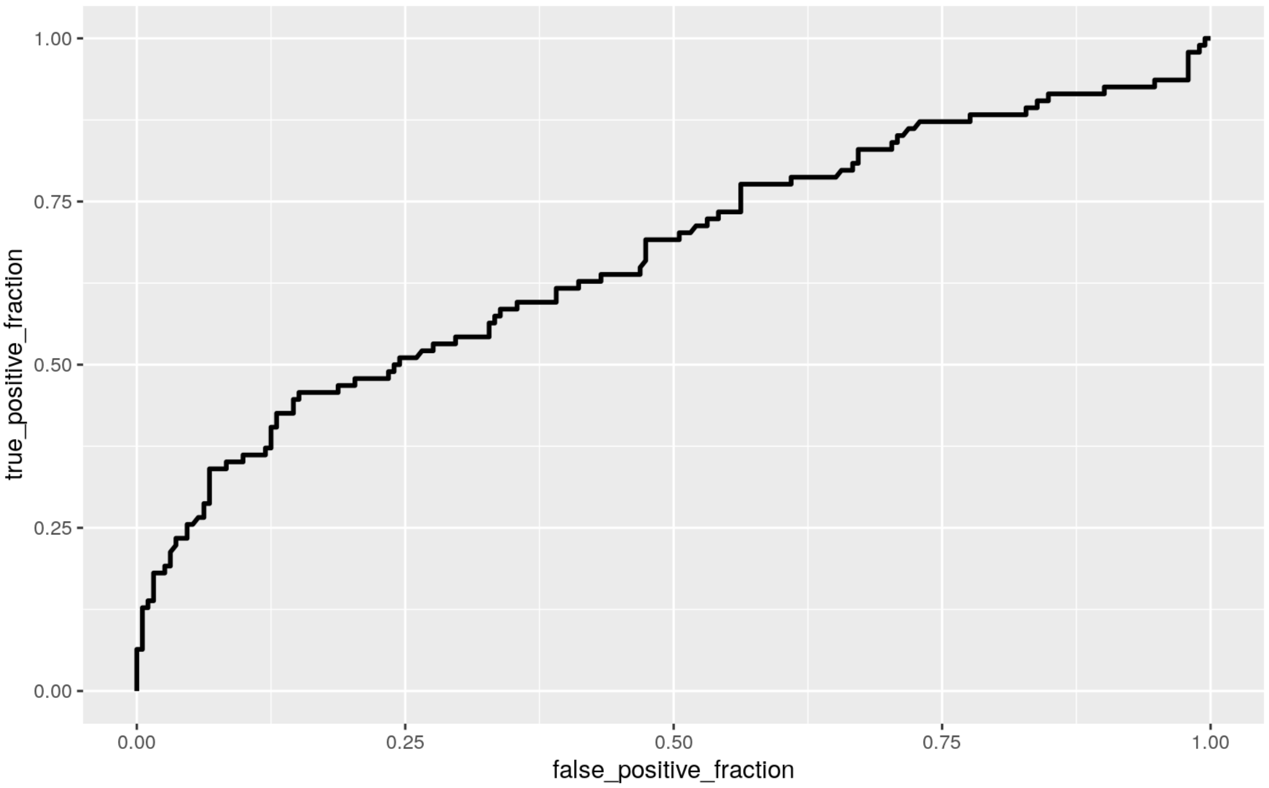
```
ggplot(leaf)+geom_density(aes(arch, fill= p), alpha = 0.5)
```



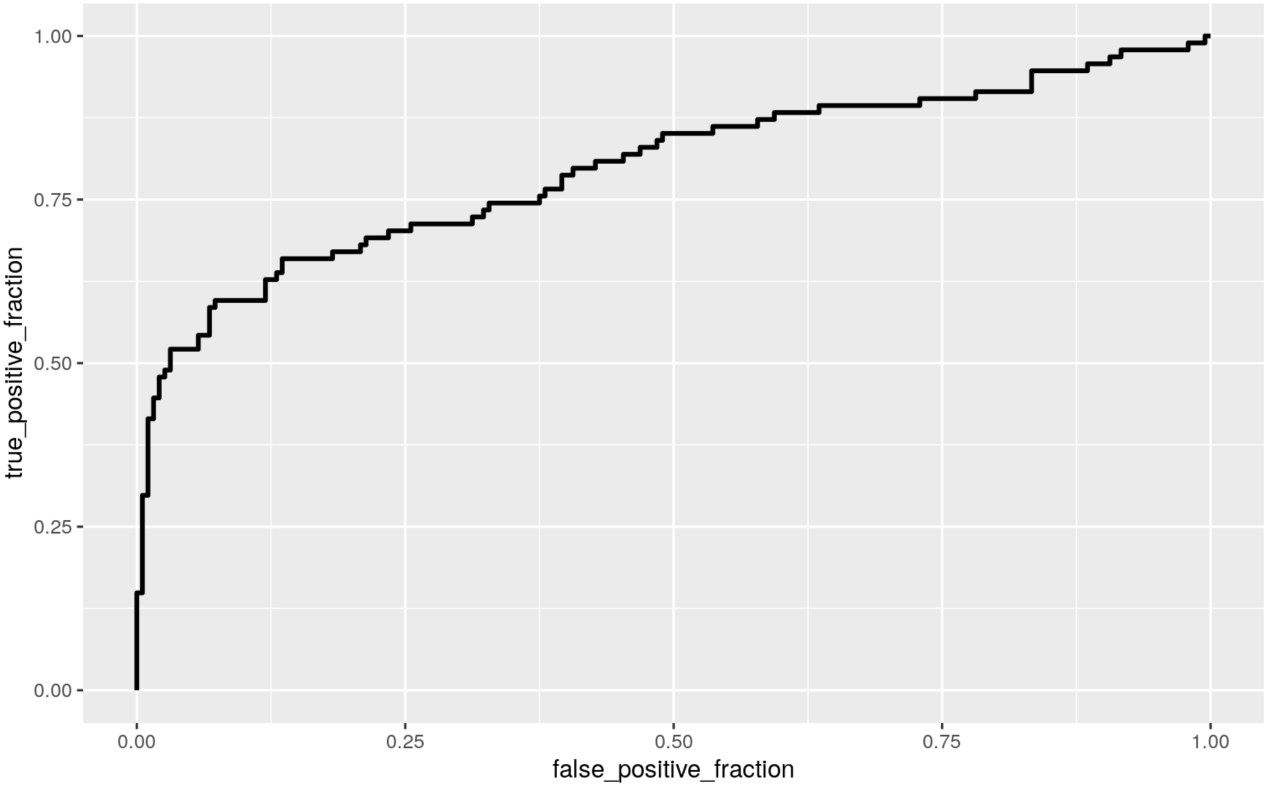
```r
library(plotROC)
ROCplot <- ggplot(leaf) + geom_roc(aes(d = arch, m = logwid), n.cuts = 0)
ROCplot1 <- ggplot(leaf) + geom_roc(aes(d = arch, m = loglen), n.cuts = 0)
ROCplot2 <- ggplot(leaf) + geom_roc(aes(d = arch, m = logpet), n.cuts = 0)
ROCplot
```



```
ROCplot1
```



```
ROCplot2
```



```
calc_auc(ROCplot)
```

```
##   PANEL group       AUC
## 1     1    -1 0.6386026
```

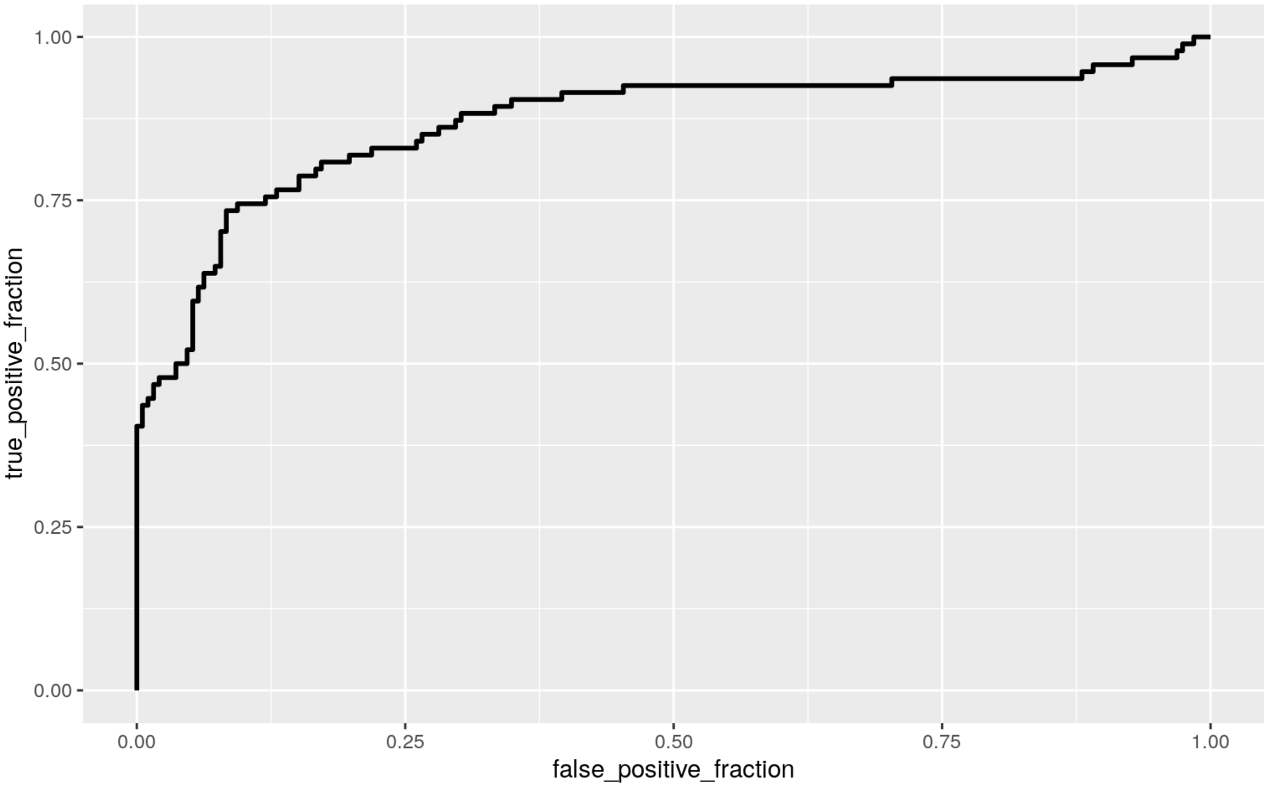
```
calc_auc(ROCplot1)
```

```
##   PANEL group       AUC
## 1     1    -1 0.6661126
```

```
calc_auc(ROCplot2)
```

```
##   PANEL group       AUC
## 1     1    -1 0.7978169
```

```
ROCplot3 <- ggplot(leaf) + geom_roc(aes(d = arch, m = predicted), n.cuts = 0)
ROCplot3
```



```
calc_auc(ROCplot3)
```

```
##   PANEL group       AUC
## 1     1    -1 0.8699025
```

```
### Class Diags Function ###

class_diag <- function(probs,truth){

  tab<-table(factor(probs>.5,levels=c("FALSE","TRUE")),truth)
  acc=sum(diag(tab))/sum(tab)
  sens=tab[2,2]/colSums(tab)[2]
  spec=tab[1,1]/colSums(tab)[1]
  ppv=tab[2,2]/rowSums(tab)[2]

  if(is.numeric(truth)==FALSE & is.logical(truth)==FALSE) truth<-as.numeric(tru
th)-1

  #CALCULATE EXACT AUC
  ord<-order(probs, decreasing=TRUE)
  probs <- probs[ord]; truth <- truth[ord]

  TPR=cumsum(truth)/max(1,sum(truth))
  FPR=cumsum(!truth)/max(1,sum(!truth))

  dup<-c(probs[-1]>=probs[-length(probs)], FALSE)
  TPR<-c(0,TPR[!dup],1); FPR<-c(0,FPR[!dup],1)

  n <- length(TPR)
  auc<- sum( ((TPR[-1]+TPR[-n])/2) * (FPR[-1]-FPR[-n]) )

  data.frame(acc,sens,spec,ppv,auc)
}

###


k=10

data1<-leaf[sample(nrow(leaf)),] #put dataset in random order
folds<-cut(seq(1:nrow(leaf)),breaks=k,labels=F) #create folds

diags<-NULL
for(i in 1:k){ # FOR EACH OF 10 FOLDS
  train<-data1[folds!=i,] #CREATE TRAINING SET
  test<-data1[folds==i,] #CREATE TESTING SET
  truth<-test$arch

  fit<- glm(arch~logwid+logpet+loglen, data = leaf, family="binomial"(link="log
it"))
  probs<- predict(fit, newdata=test, type="response")

  diags<-rbind(diags,class_diag(probs,truth))
}

apply(diags,2,mean) #AVERAGE THE DIAGNOSTICS ACROSS THE 10 FOLDS
```

```
##       acc      sens      spec       ppv       auc 
## 0.8389163 0.6441667 0.9389129 0.8376190 0.8717528
```

The logistic regression on the effect of the log length, log width and log petiole length with no interaction showed a significant difference in the odds ratio of leaf architecture due to petiole length annd leaf width, but not leaf length. an increase in log width results in a decrease in the log of the odds, since it is negative, whereas all other vlaues will increase the log of the odds of leach architecture. The control is given when arch = 0 or when leaf architecture is plagiotropic. The model corresponds to the following equation: log(odds) = 0.85(loglen) + 2.87(logpet) - 3.27(logwid) + 2.02. Eqiavalently: odds = $e^{(0.85(loglen))e}$(2.87(logpet))$e^{(-3.27(logwid))e}$(2.02). Here, odds represents the odds that the binary response variable, leaf architecture, is 1 or orthotropic.

The confusion matrix reveals a small number of false positives and false negatives, meaning the model is good at predicting leaf architecture. Keep in mind that I set >0.5 to be equal to 1 for this model.

From the confusion matrix, the accuracy is given by (60+180)/286 = 0.839, the True positive rate (TPR) is given by 60/94 = 0.638, the true negative rate (TNR) is given by 180/192 = 0.938, and the precision is given by 60/72 = 0.833 (PPV).

Looking at the ROC plots, while they do not look perfect (i.e. there are some false positives in the data), they all have an AUC of over 50%, meaning that the results are not likely due to chance. The AUCs are nevertheless relatively poor, with only baout 0.6-0.7 as its value (area under each curve). However, the logpet AUC is a lot better at close to 0.8, and can be considered good (or almost so). If we run an ROC on the predicted model, we see a much better curve with a corresponding AUC of 0.86, which is good. This means that it is liekly that we can predict the leaf architecture from all three log measurements. In other words, we have a pretty good accuracy when it comes to predicting leaf architecture from these three variables, with a high TPR for relatively low FPR.

Running the 10-fold CV showed that the AUC was close to that of before. It was around 0.855 rather than 0.86, so it was essentially the same. The accuracy was 0.839, the sensitivity was 0.630, the specificity was 0.932 and the precision was 0.856. These are all out-of sample. Every value is in fact very close, to about three decimal places, except for precision (PPV) which differed by about 0.023.

6. LASSO

```
library(glmnet)

y<-as.matrix(leaf$arch)  ###save response variable
x<-leaf[5:7]%>%scale%>%as.matrix  ###save matrix of all predictors (dropping th
e response variable)

cv<-cv.glmnet(x,y,family="binomial")
lasso<-glmnet(x,y,family="binomial",lambda=cv$lambda.1se)
coef(lasso)
```

```
## 4 x 1 sparse Matrix of class "dgCMatrix"
##                      s0
## (Intercept) -0.8308769
## logwid      -0.2805750
## logpet       1.3220459
## loglen       .
```

```
y<-as.matrix(leaf$arch)  ###save response variable
x<-leaf[5:7]%>%as.matrix  ###save matrix of all predictors (dropping the response variable)

cv<-cv.glmnet(x,y,family="binomial")
lasso<-glmnet(x,y,family="binomial",lambda=cv$lambda.1se)
coef(lasso)
```

```
## 4 x 1 sparse Matrix of class "dgCMatrix"
##                      s0
## (Intercept)  0.09770395
## logwid      -0.59835907
## logpet       1.38678247
## loglen       .
```

```
###10 fold CV



k=10

data1<-leaf[sample(nrow(leaf)),] #put dataset in random order
folds<-cut(seq(1:nrow(leaf)),breaks=k,labels=F) #create folds

diags<-NULL
for(i in 1:k){ # FOR EACH OF 10 FOLDS
  train<-data1[folds!=i,] #CREATE TRAINING SET
  test<-data1[folds==i,] #CREATE TESTING SET
  truth<-test$arch

  fit<- glm(arch~logwid+logpet, data = leaf, family="binomial"(link="logit"))
  probs<- predict(fit, newdata=test, type="response")

  diags<-rbind(diags,class_diag(probs,truth))
}

apply(diags,2,mean) #AVERAGE THE DIAGNOSTICS ACROSS THE 10 FOLDS
```

```
##       acc      sens      spec       ppv       auc
## 0.8289409 0.6251465 0.9352846 0.8125794 0.8637259
```

From the lasso, it is clear that logwid and logpet are “important” variables, i.e. they are retained in the model. Note that I did not keep the other variables (the non-log scaled) because they will provide redundant information. However, these values are still scaled twice. It can be shown that both scaled and non-sclaed values provide the same result. That is why there are two lasso coefficient tests, but the information in each is the same, essentially, since we only care that they are non-zero. The accuracy is lower by a small margin (0.01). This means that only using the variables from the LASSO gives a slightly better fit to the data than considering all variables.

…