HW2_Wang_Chaoran

Chaoran Wang

2017-09-13

Fineshed Problem 1 to 3

Problem 4

One of the benefits of Version Control is that it is not necessary to notify my teammates that I would modify some files in our shared folders and they should not work on them at the same time. The original process was unrealistic and wrong. Now, with the Version Control system, all my teammates should be able to work on the files at the same time and we are not worried about losing anything.

Problem 5

(a)

First, I read in and create a tidy dataset. To tidy the data, I first read in the data and remove the first row. Then I extract all rows with interal number 1-10 out and put them in (a) set. In this dataset, I corrected the column name to be what they should to be, Item and Operator 1 to 5. Next, I extract all other rows into (b) set. Since the first column is character at first, I change it into numeric. Then, I combine two sets, gather them based on operator number, and arrange them into the correct order.

After tidying, a summary is in Table 1. Since I am not sure what the experiment is actually, I cannot analyze the data based on the information I have. It looks not like a simple linear model. The codes are in Appendix.

```
## Warning: package 'bindrcpp' was built under R version 3.3.3
```

Table 1: Sensory Data Summary

Item	Operator	value
Length:150	Length:150	Min. :0.700
Class :character	Class :character	1st Qu.:3.025
Mode :character	Mode :character	Median :4.700
NA	NA	Mean :4.657
NA	NA	3rd Qu.:6.000
NA	NA	Max. :9.400

(b)

First, I read in and create a tidy dataset. To tidy the dataset, I first rename the columns. I then separate and combine the dataset to make it has two columns only. In order to make the data be more readable, I add another column which is the respective year (1900 + *).

After tidying, a summary is in Table 2 with a plot in Figure 1 are created. Then I fit the linear model (Table 3) and plot the fitted line on Figure 1. The fitted model looks not bad. The codes are in Appendix.

Table 2: LJD Data Summary

Year_00	Year	Long_Jump
Min. :-4.00	Min. :1896	Min. :249.8
1st Qu.:21.00	1st Qu.:1921	1st Qu.:295.4
Median :50.00	Median :1950	Median :308.1
Mean :45.45	Mean :1945	Mean :310.3
3rd Qu.:71.00	3rd Qu.:1971	3rd Qu.:327.5
Max. :92.00	Max. :1992	Max. :350.5

Table 3: Fitting Linear Models of LJD Data

	Dependent variable:
	Long_Jump
Year	0.709*** (0.078)
Constant	−1,069.333*** (151.777)
Observations	22
R^2	0.805
Adjusted R^2	0.795
Residual Std. Error	11.019 (df = 20)
F Statistic	82.645*** (df = 1; 20)
Note:	*p<0.1; **p<0.05; ***p<0.01

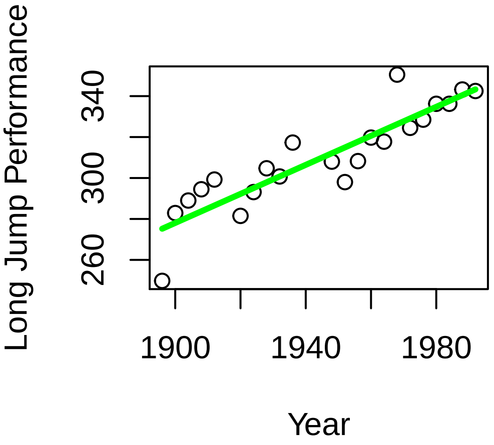

Figure 1: Gold Medal performance for Olympic Men's Long Jump since 1886

(c)

First, I read in and create a tidy dataset. I first rename the columns. I then separate and combine the dataset to make it has two columns only. In order to make the variables to be continuous and increasing, I order the weight column.

After tidying, a summary is in Table 4 with a plot in Figure 2 are created. The codes are in Appendix.

Table 4: BBW Data Summary

Body_Wt	Brain_Wt
Min. : 0.005	Min. : 0.10
1st Qu.: 0.600	1st Qu.: 4.25
Median : 3.342	Median : 17.25
Mean : 198.790	Mean : 283.13
3rd Qu.: 48.203	3rd Qu.: 166.00
Max. :6654.000	Max. :5712.00

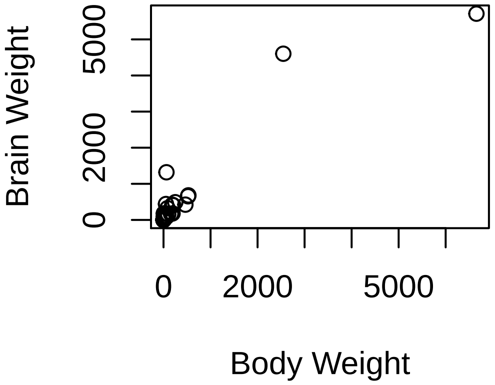

Figure 2: Brain weight (g) vs. body weight (kg) for 62 species

However, from the Figure 2 above, we can see there are two obvious outliers which might affect the accurate of our fitted model. I choose to delete them and fit the linear model. The summary of the modified dataset are in Table 5. Then I fit the linear model (Table 6) and plot the modified data and the fitted line on Figure 3. We can see the point at top left corner seems to be another outlier but the fitted line looks not bad. Since the procedure are similar and we are not focus on analysis at this time, I do not remove that point and process the data again. The codes are in Appendix.

Table 5: BBW Data Summary Re-do

Body_Wt	Brain_Wt
Min. : 0.0050	Min. : 0.100
1st Qu.: 0.5325	1st Qu.: 3.975
Median : 3.1500	Median : 16.250
Mean : 52.0663	Mean : 120.656
3rd Qu.: 35.3325	3rd Qu.: 128.875
Max. :529.0000	Max. :1320.000

Table 6: Fitting Linear Models of BBW Data

	Dependent variable:
	Brain_Wt
Body_Wt	1.243*** (0.190)
Constant	55.952** (24.650)
Observations	60
R^2	0.423
Adjusted R^2	0.413
Residual Std. Error	174.805 (df = 58)
F Statistic	42.560*** (df = 1; 58)
Note:	*p<0.1; **p<0.05; ***p<0.01

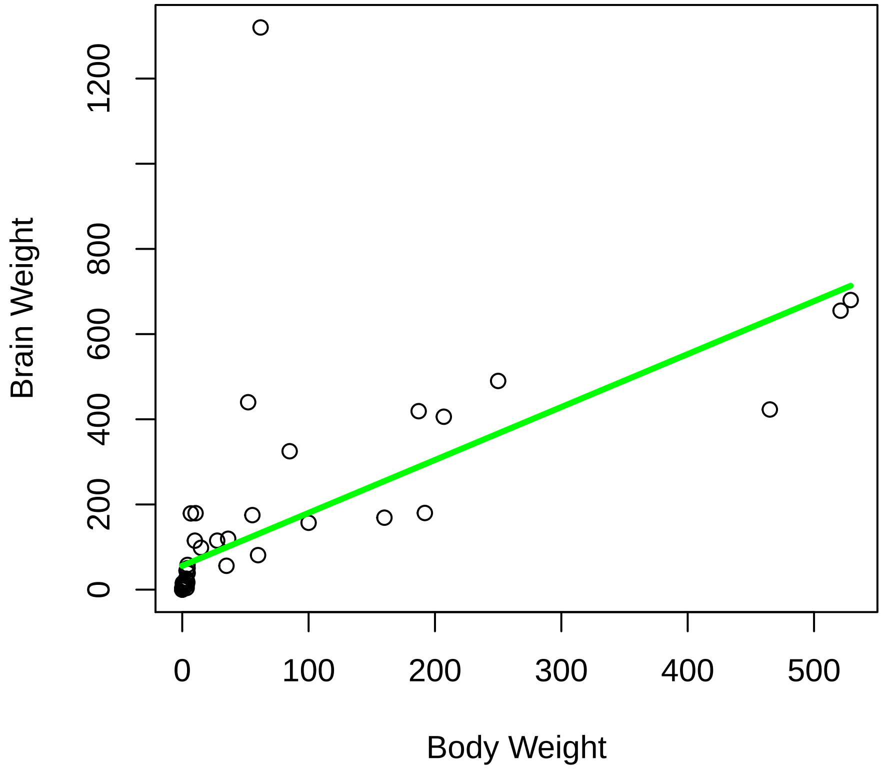

Figure 3: Brain weight (g) vs. body weight (kg) for 60 species

(d)

First, I read in and create a tidy dataset. To tidy the data, I first read it in R. The dataset looks messy because it has 3 values in one cell and the first row name looks bad. I separate the data based on 3 different densities. Since the data has "10.1," after separating, I use mutate function to remove it. Then, I just do some basic cleaning work to rename and re-order the dataset.

After tidying, a summary is in Table 7. Since I am not sure what the experiment is actually, I cannot analyze the data based on the information I have. It looks not like a simple linear model so I do not to linear model analysis. The codes are in Appendix.

Table 7: Tomato Data Summary

Varieties	Density	Triplicates	Yields
Length:18	Length:18	Length:18	Min. : 8.10
Class :character	Class :character	Class :character	1st Qu.:12.95
Mode :character	Mode :character	Mode :character	Median :15.35
NA	NA	NA	Mean :15.07
NA	NA	NA	3rd Qu.:17.88
NA	NA	NA	Max. :21.00

Problem 6

After reading the data in R, I done a brief clean of it such as rename the variables, remove NAs, and select the three columns we are interested in. Since the Foliage Color is a categorical response, in order to to do linear regression analysis, I changed them to numeric. Even though I think this way is rough and not very correct, I do not know other way to solve the problem.

After converting the response into numeric, I begin to analysis the relationship. Since I am not sure what combination you want for pH_Min and pH_Max, I am doing simple linear regression on them separately, multiple linear regression on them, simple regression on their difference, and take log of the response to get the fit. The summary and ANOVA table are shown below. The one with pH_Max only seems to be the best one. However, I have to say, the basic of this analysis is **not reasonable** in my idea because it seems not logical to change the categorical response to numeric. The codes are in Appendix.

Note: I do not know why but I keep getting error when using "stargazer" to show lm table. So I just print the summary directly.

Table 8: Plants Data Summary

pH_Min	pH_Max	Color.factor	pH_Dif
Min. :3.000	Min. : 5.100	Min. :1.000	Min. :0.400
1st Qu.:4.500	1st Qu.: 7.000	1st Qu.:3.000	1st Qu.:1.900
Median :5.000	Median : 7.300	Median :3.000	Median :2.200
Mean :4.994	Mean : 7.345	Mean :2.839	Mean :2.351
3rd Qu.:5.500	3rd Qu.: 7.800	3rd Qu.:3.000	3rd Qu.:2.900
Max. :7.000	Max. :10.000	Max. :6.000	Max. :5.600

Call: lm(formula = Color.factor ~ pH_Min)

Residuals: Min 1Q Median 3Q Max -1.9064 0.1250 0.1608 0.1832 3.2056

Coefficients: Estimate Std. Error t value Pr(>|t|)

(Intercept) 2.61515 0.19384 13.492 <2e-16 *** pH_Min 0.04481 0.03848 1.165 0.244
— Signif. codes: 0 ‘’ ***0.001*** ’’ *0.01* ” 0.05 ‘.’ 0.1 ‘’ 1

Residual standard error: 0.7344 on 830 degrees of freedom Multiple R-squared: 0.001632, Adjusted R-squared: 0.0004287 F-statistic: 1.356 on 1 and 830 DF, p-value: 0.2445

Table 9: ANOVA of fitting Plants Data with pH_Min

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
pH_Min	1	0.7316227	0.7316227	1.35641	0.2444952
Residuals	830	447.6866465	0.5393815	NA	NA

Call: lm(formula = Color.factor ~ pH_Max)

Residuals: Min 1Q Median 3Q Max -1.9622 0.1056 0.1479 0.1903 3.2072

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.21674 0.27466 8.071 2.44e-15 ** *pH_Max 0.08471 0.03723 2.275 0.0232*
— Signif. codes: 0 ‘’ ***0.001*** ’’ *0.01* ” 0.05 ‘.’ 0.1 ‘’ 1

Residual standard error: 0.7327 on 830 degrees of freedom Multiple R-squared: 0.006198, Adjusted R-squared: 0.005 F-statistic: 5.176 on 1 and 830 DF, p-value: 0.02315

Table 10: ANOVA of fitting Plants Data with pH_Max

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
pH_Max	1	2.779217	2.7792168	5.176274	0.0231516
Residuals	830	445.639052	0.5369145	NA	NA

Call: lm(formula = Color.factor ~ pH_Min + pH_Max)

Residuals: Min 1Q Median 3Q Max -1.9636 0.0994 0.1565 0.1921 3.2080

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.16108 0.29718 7.272 8.21e-13 ** *pH_Min 0.01984 0.04036 0.492 0.6231*
pH_Max 0.07880 0.03915 2.013 0.0444
— Signif. codes: 0 ‘’ ***0.001*** ’’ *0.01* ” 0.05 ‘.’ 0.1 ‘’ 1

Residual standard error: 0.7331 on 829 degrees of freedom Multiple R-squared: 0.006488, Adjusted R-squared: 0.004091 F-statistic: 2.707 on 2 and 829 DF, p-value: 0.06735

Table 11: ANOVA of fitting Plants Data with pH_Min and pH_Max

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
pH_Min	1	0.7316227	0.7316227	1.361398	0.2436309
pH_Max	1	2.1775153	2.1775153	4.051904	0.0444444
Residuals	829	445.5091312	0.5374055	NA	NA

Call: lm(formula = Color.factor ~ pH_Dif)

Residuals: Min 1Q Median 3Q Max -1.9101 0.1342 0.1595 0.1753 3.2070

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.76455 0.07984 34.626 <2e-16 *** pH_Dif 0.03165 0.03219 0.983 0.326

— Signif. codes: 0 ‘’ ***0.001*** ’’ *0.01* ” 0.05 ‘.’ 0.1 ‘’ 1

Residual standard error: 0.7346 on 830 degrees of freedom Multiple R-squared: 0.001163, Adjusted R-squared: -4.026e-05 F-statistic: 0.9665 on 1 and 830 DF, p-value: 0.3258

Table 12: ANOVA of fitting Plants Data with pH_Dif

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
pH_Dif	1	0.5215808	0.5215808	0.9665445	0.32583
Residuals	830	447.8966884	0.5396346	NA	NA

Call: lm(formula = log(Color.factor) ~ pH_Min + pH_Max)

Residuals: Min 1Q Median 3Q Max -1.06553 0.06992 0.09987 0.12081 0.82317

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.622243 0.140983 4.414 1.15e-05 ** *pH_Min 0.006924 0.019146 0.362 0.7177*
pH_Max 0.046046 0.018571 2.480 0.0134
— Signif. codes: 0 ‘’ ***0.001*** ’’ *0.01* ” 0.05 ‘.’ 0.1 ‘’ 1

Residual standard error: 0.3478 on 829 degrees of freedom Multiple R-squared: 0.009016, Adjusted R-squared: 0.006625 F-statistic: 3.771 on 2 and 829 DF, p-value: 0.02342

Table 13: ANOVA of fitting Plants Data with pH_Min and pH_Max

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
pH_Min	1	0.1686304	0.1686304	1.394286	0.2380213
pH_Max	1	0.7435624	0.7435624	6.147994	0.0133539
Residuals	829	100.2624976	0.1209439	NA	NA

Problem 7

For this problem, we are trying to munge some large datasets and give a briefly summary. Since the datasets are very large, it might be a good idea to read some of them (first 100 rows here) in R and finish the necessary work of columns first. The files are local on my computer. The directory might be different.

[1] “Gebreken” Gebrek.identificatie Ingangsdatum.gebrek Einddatum.gebrek “character” “integer” “integer” Gebrek.paragraaf.nummer Gebrek.artikel.nummer Gebrek.omschrijving “integer” “character” “character” [1] “Geconstat” Kenteken Soort.erkenning.keuringsinstantie “character” “character” Meld.datum.door.keuringsinstantie Meld.tijd.door.keuringsinstantie “integer” “integer” Gebrek.identificatie Soort.erkenning.omschrijving “character” “character” Aantal.gebreken.geconstateerd “integer” [1] “Personen” Kenteken Voertuigsoort “character” “character” Merk Handelsbenaming “character” “character” Datum.tenaamstelling Bruto.BPM “character” “integer” Cilinderinhoud Massa.ledig.voertuig “integer” “integer” Toegestane.maximum.massa.voertuig Datum.eerste.toelating “integer” “character” Datum.eerste.afgifte.Nederland Catalogusprijs “character” “integer” WAM.verzekerd “character”

What columns they have are not shown in English which makes me confused. But never mind, I use Google Translate to understand what the columns mean. Continuing with the datasets, I figured that it was unrealistic to read all data in and analyze them because there are so much information in these three datasets. That might be better to focus just one aspect, maybe the data in one year, to be our interest. So I choose rows in 2017 to be my interest just as Dr. Settlage required. And for the columns, I also just choose few of them. For “Gebreken”, it has defect code information and description. For “Geconstat”, it has inspection date and defect code in respective to license plates. For “Personen”, it has make and model of vehicle in respective to license plates. Using Google Translate, I figured which columns I need, first and sixth columns of “Gebreken”, first, third, and fifth columns of “Geconstat” (Defects), and first, third, and forth columns of “Person”. Thanks to the work from Dr. Settlage, I learned that “fread” is a function which is similar to read.table but much faster and convienient. I use it then to read the specfic data I am interested in.

Next, I use merge function to merge three datasets by plates and defect code. During cleaning data, I translate the columns into English.

[1] 503 [1] 33470

AC1 K04 RA2 205 497 385621 271259 223142 171244 139980

```
## Warning in anova.lmlist(object, ...): models with response '"NULL"' removed
## because response differs from model 1
```

```
## Warning in anova.lm(object): ANOVA F-tests on an essentially perfect fit
## are unreliable
```

Table 14: Defects Data Summary

Make	Num_Def
A.C.L. : 1	Min. : 1.0
A.M.C. : 1	1st Qu.: 3.0
ACCUBUILT: 1	Median : 9.0
ACM : 1	Mean : 8727.3
ACURA : 1	3rd Qu.: 60.5
ADRIA : 1	Max. :466979.0
(Other) :497	NA

```
## Warning in anova.lm(lm_defect_make): ANOVA F-tests on an essentially
## perfect fit are unreliable
```

Table 15: ANOVA of fitting Number of defects with Make

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
defect_make$Make	502	1.102845e+12	2196902116	NaN	NaN
Residuals	0	0.000000e+00	NaN	NA	NA

Hence, there are 503 different makes and 33470 different models reported defects in 2017. And the regression summary and ANOVA table are shown. For the lm and anova analysis, I finished it for Make but keep getting error “cannot allocate vector of size 8.3 Gb” when doing same thing for Model. I guess that might because of the size of dataframe is to large for lm function. Hope you would not mark me wrong. Besides the output of lm summary is too long, so I do not show it here. The codes are in Appendix.

Appendix 1: R code

```
########################### Problem5_Sensory_analysis get data
url1 <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/Sensory.dat"
sen_raw <- read.table(url1, header = F, skip = 1, fill = T, stringsAsFactors = F)
sen_tidy <- sen_raw[-1, ]
sen_tidy_a <- filter(.data = sen_tidy, V1 %in% 1:10)
sen_tidy_a <- rename(sen_tidy_a, Item = V1, V1 = V2, V2 = V3, V3 = V4, V4 = V5,
    V5 = V6)
sen_tidy_b <- filter(.data = sen_tidy, !(V1 %in% 1:10))
sen_tidy_b <- mutate(sen_tidy_b, Item = rep(as.character(1:10), each = 2))
sen_tidy_b <- mutate(sen_tidy_b, V1 = as.numeric(V1))
sen_tidy_b <- select(sen_tidy_b, c(Item, V1:V5))
sen_tidy <- bind_rows(sen_tidy_a, sen_tidy_b)
sen_tidy <- gather(sen_tidy, Operator, value, V1:V5)
sen_tidy <- mutate(sen_tidy, Operator = gsub("V", "", Operator))
sen_tidy <- arrange(sen_tidy, Item)
########################## Problem5_LJD_analysis get data
url2 <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/LongJumpData.dat"
LJD_raw <- read.table(url2, header = F, skip = 1, fill = T, stringsAsFactors = F)
colnames(LJD_raw) <- rep(c("Year_00", "Long_Jump"), 4)
LJD_raw1 <- bind_rows(LJD_raw[, 1:2], LJD_raw[, 3:4])
LJD_raw2 <- bind_rows(LJD_raw[, 5:6], LJD_raw[1:4, 7:8])
LJD_tidy <- bind_rows(LJD_raw1, LJD_raw2)
Year <- LJD_tidy[, 1] + 1900
LJD_tidy$Year <- Year
LJD_tidy <- LJD_tidy[c(1, 3, 2)]
########################### Problem5_LJD_analysis plot
knitr::kable(summary(LJD_tidy), caption = "LJD Data Summary")
LJD_tidy_lm <- lm(Long_Jump ~ Year, data = LJD_tidy)
stargazer(LJD_tidy_lm, title = "Fitting Linear Models of LJD Data", header = F,
    no.space = T, single.row = T)
plot(LJD_tidy$Year, LJD_tidy$Long_Jump, xlab = "Year", ylab = "Long Jump Performance")
lines(LJD_tidy$Year, LJD_tidy_lm$fitted, col = "green", lwd = 3, lty = 1)
########################### Problem5_BBW_analysis get data
url3 <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/BrainandBodyWeight.dat"
BBW_raw <- read.table(url3, header = F, skip = 1, fill = T, stringsAsFactors = F)
colnames(BBW_raw) <- rep(c("Body_Wt", "Brain_Wt"), 3)
BBW_raw1 <- bind_rows(BBW_raw[, 1:2], BBW_raw[, 3:4])
BBW_tidy <- bind_rows(BBW_raw1, BBW_raw[1:20, 5:6])
attach(BBW_tidy)
BBW_tidy <- BBW_tidy[order(Body_Wt), ]
########################### Problem5_BBW_analysis plot
knitr::kable(summary(BBW_tidy), caption = "BBW Data Summary")
plot(BBW_tidy, xlab = "Body Weight", ylab = "Brain Weight")
########################### Problem5_BBW_analysis remove outliers
BBW_tidy_new <- BBW_tidy[-c(61, 62), ]
knitr::kable(summary(BBW_tidy_new), caption = "BBW Data Summary Re-do")
BBW_tidy_lm <- lm(Brain_Wt ~ Body_Wt, data = BBW_tidy_new)
stargazer(BBW_tidy_lm, title = "Fitting Linear Models of BBW Data", header = F,
    no.space = T, single.row = T)
plot(BBW_tidy_new, xlab = "Body Weight", ylab = "Brain Weight")
lines(BBW_tidy_new$Body_Wt, BBW_tidy_lm$fitted, col = "green", lwd = 3, lty = 1)
```

```r
########################### Problem5_tomato_analysis get data
url4 <- "http://www2.isye.gatech.edu/~jeffwu/wuhamadabook/data/tomato.dat"
tom_raw <- read.table(url4, header = F, skip = 2, fill = T, stringsAsFactors = F,
    comment.char = "")
tom_tidy <- separate(tom_raw, V2, into = paste("Den_10000", 1:3, sep = "_"),
    sep = ",", remove = T, extra = "merge")
tom_tidy <- separate(tom_tidy, V3, into = paste("Den_20000", 1:3, sep = "_"),
    sep = ",", remove = T, extra = "merge")
tom_tidy <- separate(tom_tidy, V4, into = paste("Den_30000", 1:3, sep = "_"),
    sep = ",", remove = T, extra = "merge")
tom_tidy <- mutate(tom_tidy, Den_10000_3 = gsub(",", "", Den_10000_3))
tom_tidy <- gather(tom_tidy, Density, value, Den_10000_1:Den_30000_3)
tom_tidy <- mutate(tom_tidy, Density = gsub("Den_", "", Density))
tom_tidy <- separate(tom_tidy, Density, into = c("Density", "Triplicates"))
tom_tidy <- mutate(tom_tidy, Varieties = gsub("\\\\#", "", V1))
tom_tidy <- transform(tom_tidy, Yields = as.numeric(value))
tom_tidy <- select(tom_tidy, Varieties, Density, Triplicates, Yields)
tom_tidy <- arrange(tom_tidy, Varieties)
########################## Problem6_analysis plot
knitr::kable(summary(plants_tidy_new), caption = "Plants Data Summary")
lm1 <- lm(Color.factor ~ pH_Min)
summary(lm1)
# stargazer(lm1,title = 'Fitting Linear Models of fitting Plants Data with
# pH_Min',header = F,no.space=T,single.row=T)
knitr::kable(anova(lm1), caption = "ANOVA of fitting Plants Data with pH_Min")
lm2 <- lm(Color.factor ~ pH_Max)
summary(lm2)
# stargazer(lm2,title = 'Fitting Linear Models of fitting Plants Data with
# pH_Max',header = F,no.space=T,single.row=T)
knitr::kable(anova(lm2), caption = "ANOVA of fitting Plants Data with pH_Max")
lm3 <- lm(Color.factor ~ pH_Min + pH_Max)
summary(lm3)
# stargazer(lm3,title = 'Fitting Linear Models of fitting Plants Data with
# pH_Min and pH_Max',header = F,no.space=T,single.row=T)
knitr::kable(anova(lm3), caption = "ANOVA of fitting Plants Data with pH_Min and pH_Max")
lm4 <- lm(Color.factor ~ pH_Dif)
summary(lm4)
# stargazer(lm4,title = 'Fitting Linear Models of fitting Plants Data with
# pH_Dif',header = F,no.space=T,single.row=T)
knitr::kable(anova(lm4), caption = "ANOVA of fitting Plants Data with pH_Dif")
lm5 <- lm(log(Color.factor) ~ pH_Min + pH_Max)
summary(lm5)
# stargazer(lm5,title = 'Fitting Linear Models of fitting log of Plants Data
# with pH_Min and pH_Max',header = F,no.space=T,single.row=T)
knitr::kable(anova(lm5), caption = "ANOVA of fitting Plants Data with pH_Min and pH_Max")

########################### Problem7 munge

# defect code and description
Car_Gebreken_select <- fread(input = "D:/Open_Data_RDW__Gebreken.csv", header = T,
    select = c(1, 6), showProgress = F)
# license plate, inspection date and defect code
Car_Geconstat_select <- fread(input = "D:/Open_Data_RDW__Geconstateerde_Gebreken.csv",
    header = T, select = c(1, 3, 5), showProgress = F)
# license plate, make and model of vehicle
Car_Person_select <- fread(input = "D:/Personenauto_basisdata.csv", header = T,
    showProgress = F, select = c(1, 3, 4))

Car_Geconstat_select_2017 <- Car_Geconstat_select[grep("2017", Car_Geconstat_select$"Meld datum door keu
    ]
# merge datasets
Car_plates <- merge(Car_Geconstat_select_2017, Car_Person_select, by = "Kenteken")
Car_defect <- merge(Car_Gebreken_select, Car_plates, by = "Gebrek identificatie")
# Clean data
Car_defect <- Car_defect[complete.cases(Car_defect), ]
# Translate
colnames(Car_defect) <- c("Defect_Code", "Description", "License_Plate", "Report_Date",
    "Make", "Model")
num_makes <- length(unique(Car_defect$Make))
print(num_makes)
num_models <- length(unique(Car_defect$Model))
print(num_models)
# most frequent defects
print(sort(table(Car_defect$Defect_Code), decreasing = TRUE)[1:5])
# number of defects vs. make
defect_make <- as.data.frame(table(Car_defect$Make))
colnames(defect_make) <- c("Make", "Num_Def")
lm_defect_make <- lm(defect_make$Num_Def ~ defect_make$Make)
ano_defect_make <- anova(lm_defect_make, data = defect_make)
# summary(lm_defect_make)
knitr::kable(summary(defect_make), caption = "Defects Data Summary")
knitr::kable(anova(lm_defect_make), caption = "ANOVA of fitting Number of defects with Make")
# number of defects vs. model
defect_model <- as.data.frame(table(Car_defect$Model))
colnames(defect_model) <- c("Model", "Num_Def")
# lm_defect_model <- lm(defect_model$Num_Def~defect_model$Model)
# ano_defect_make <- anova(lm_defect_make, data=defect_make)
# summary(lm_defect_make) summary(ano_defect_make)
```