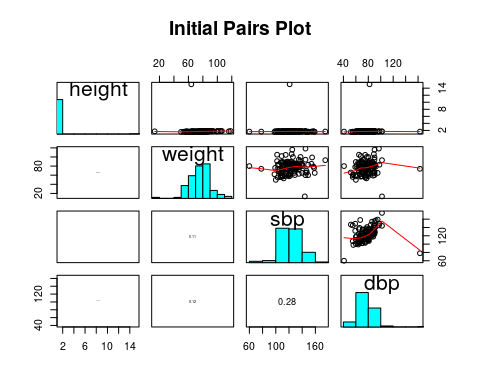
Stats 330 Assignment 1

Hasnain Cheena

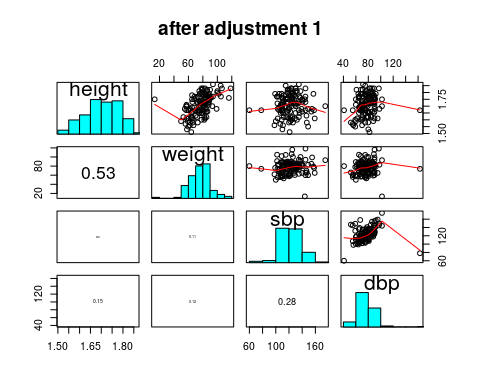
16/03/2020

## Data Cleaning

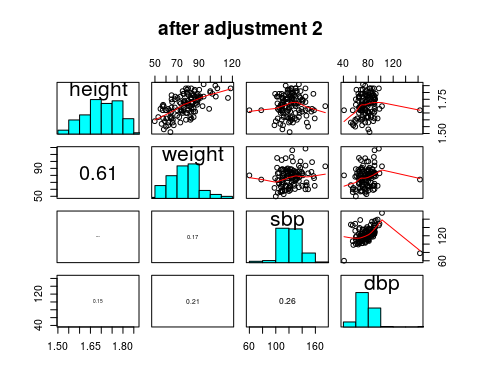
#read in data  
  
heartHealth.df = read.csv("hearthealth.csv")  
#make a copy for this question  
heartHealth.cleaned.df = heartHealth.df[c("height", "weight", "sbp", "dbp")]  
  
#initial pairs plot  
pairs20x(heartHealth.cleaned.df, main="Initial Pairs Plot")



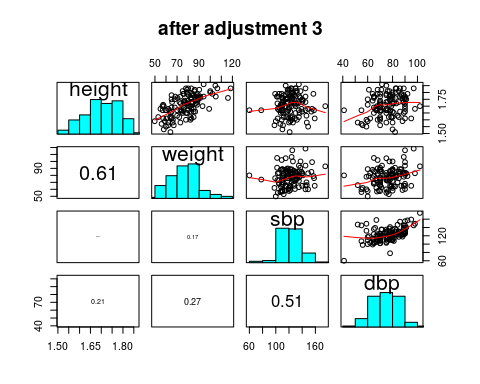
#adjustment 1  
heartHealth.cleaned.df[heartHealth.cleaned.df["height"] == 15.1, "height"] = 1.51  
pairs20x(heartHealth.cleaned.df, main="after adjustment 1")



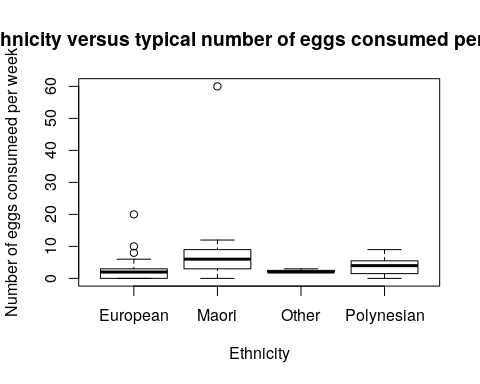
#adjustment 2  
heartHealth.cleaned.df = heartHealth.cleaned.df[-c(106), ]  
pairs20x(heartHealth.cleaned.df, main="after adjustment 2")



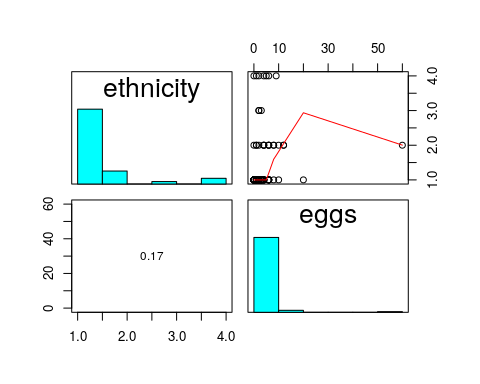
#adjustment 3  
sdp = heartHealth.cleaned.df[44, "sbp"]  
dbp = heartHealth.cleaned.df[44, "dbp"]  
heartHealth.cleaned.df[44, "dbp"] = sdp  
heartHealth.df[44, "sbp"] = dbp  
pairs20x(heartHealth.cleaned.df, main="after adjustment 3")



#remove any NA's  
eggs.df = heartHealth.df[,c("ethnicity", "eggs")]  
eggs.df = eggs.df[complete.cases(eggs.df[, "eggs"]),]  
  
#box plot  
plot(eggs.df$ethnicity, eggs.df$eggs, main="Ethnicity versus typical number of eggs consumed per week",  
 xlab="Ethnicity", ylab="Number of eggs consumeed per week")



#pairs plot  
pairs20x(eggs.df[c("ethnicity", "eggs")])



#fit model  
eggs.fit = glm(eggs ~ ethnicity, data=eggs.df, family="poisson")  
#anova   
anova(eggs.fit, test = "Chisq")

## Analysis of Deviance Table  
##   
## Model: poisson, link: log  
##   
## Response: eggs  
##   
## Terms added sequentially (first to last)  
##   
##   
## Df Deviance Resid. Df Resid. Dev Pr(>Chi)   
## NULL 117 591.34   
## ethnicity 3 142.71 114 448.63 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

#summary   
summary(eggs.fit)

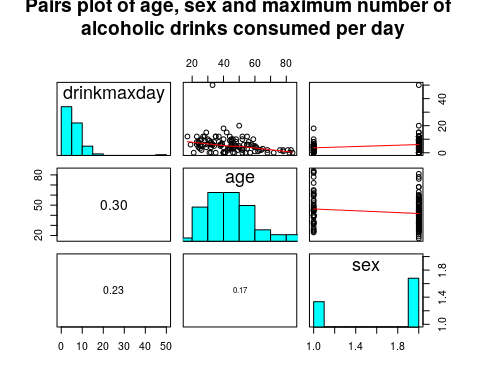
##   
## Call:  
## glm(formula = eggs ~ ethnicity, family = "poisson", data = eggs.df)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.2426 -1.8742 -0.1699 0.4755 11.2096   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.81093 0.06950 11.667 < 2e-16 \*\*\*  
## ethnicityMaori 1.38629 0.10851 12.776 < 2e-16 \*\*\*  
## ethnicityOther 0.03637 0.38430 0.095 0.92461   
## ethnicityPolynesian 0.53900 0.20462 2.634 0.00843 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for poisson family taken to be 1)  
##   
## Null deviance: 591.34 on 117 degrees of freedom  
## Residual deviance: 448.63 on 114 degrees of freedom  
## AIC: 727.5  
##   
## Number of Fisher Scoring iterations: 5

#confidence intervals - european baseline  
100\*(exp(confint(eggs.fit)) -1)

## Waiting for profiling to be done...

## 2.5 % 97.5 %  
## (Intercept) 95.72387 157.0573  
## ethnicityMaori 222.86037 394.2273  
## ethnicityOther -55.86501 103.7081  
## ethnicityPolynesian 12.25882 151.1673

#subset and remove any NA's  
drinking.df = heartHealth.df[, c("drinkmaxday", "age", "sex")]  
  
pairs20x(drinking.df, main="Pairs plot of age, sex and maximum number of \n alcoholic drinks consumed per day")



#fit model  
drinks.fit = glm(drinkmaxday ~ age\*sex, data=drinking.df, family="poisson")  
summary(drinks.fit)

##   
## Call:  
## glm(formula = drinkmaxday ~ age \* sex, family = "poisson", data = drinking.df)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.4430 -1.1539 -0.3180 0.5838 10.0326   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 2.474042 0.247327 10.003 < 2e-16 \*\*\*  
## age -0.024327 0.005548 -4.385 1.16e-05 \*\*\*  
## sexM 0.212089 0.282507 0.751 0.453   
## age:sexM 0.005440 0.006463 0.842 0.400   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for poisson family taken to be 1)  
##   
## Null deviance: 569.26 on 119 degrees of freedom  
## Residual deviance: 474.67 on 116 degrees of freedom  
## AIC: 844.69  
##   
## Number of Fisher Scoring iterations: 5

#fit model - remove interaction  
drinks.fit = glm(drinkmaxday ~ age+sex, data=drinking.df, family="poisson")  
#model summary  
summary(drinks.fit)

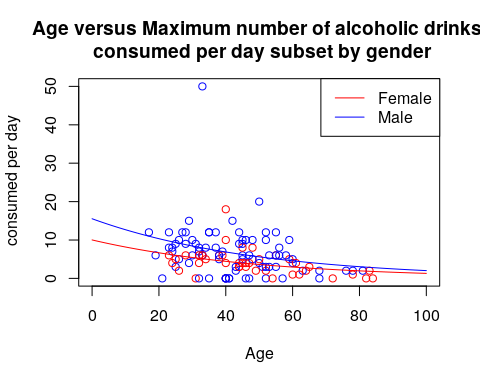
##   
## Call:  
## glm(formula = drinkmaxday ~ age + sex, family = "poisson", data = drinking.df)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -4.5017 -1.1361 -0.2448 0.6126 9.9960   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 2.304301 0.145300 15.859 < 2e-16 \*\*\*  
## age -0.020351 0.002839 -7.167 7.66e-13 \*\*\*  
## sexM 0.438832 0.090249 4.862 1.16e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for poisson family taken to be 1)  
##   
## Null deviance: 569.26 on 119 degrees of freedom  
## Residual deviance: 475.39 on 117 degrees of freedom  
## AIC: 843.4  
##   
## Number of Fisher Scoring iterations: 5

#confidence intervals   
100\*(exp(confint(drinks.fit))-1)

## Waiting for profiling to be done...

## 2.5 % 97.5 %  
## (Intercept) 651.913430 1229.154739  
## age -2.562935 -1.472227  
## sexM 30.281393 85.621145

#plot part d  
ages <- data.frame(age=seq(0,100), sex="M")  
malePredictedValues <- predict(drinks.fit, newdata = ages)  
malePredictedValues <- exp(malePredictedValues)  
ages <- data.frame(age=seq(0,100), sex="F")  
femalePredictedValues <- predict(drinks.fit, newdata = ages)  
femalePredictedValues <- exp(femalePredictedValues)  
  
plot(drinkmaxday ~ age, data = heartHealth.df,col = ifelse(heartHealth.df$sex == "M" , "blue", "red"), xlim=c(0, 100),   
 main="Age versus Maximum number of alcoholic drinks \n consumed per day subset by gender", xlab="Age", ylab="Maximum number of alcoholic drinks \n consumed per day")  
lines(ages$age, femalePredictedValues, col="red")  
lines(ages$age, malePredictedValues, col="blue")  
legend("topright", legend=c("Female", "Male"),  
 col=c("red", "blue"), lty=1)



#e  
test1.df = data.frame(age=c(30,50), sex=c("M", "M"))  
test1.predict = predict(drinks.fit, newdata = test1.df, type = "response")  
test2.df = data.frame(age=c(40,40), sex=c("M", "F"))  
test2.predict = predict(drinks.fit, newdata = test2.df, type = "response")  
print(test1.predict)

## 1 2   
## 8.436875 5.615880

print(test2.predict)

## 1 2   
## 6.883348 4.438309

## Exercise

1. The response variable is continous and not a count/proportion type and thus a linear regression model is appropriate.
2. We have evidence to show that age and smoking history are related to the number of hours typical number of hours of vigorous exercise undertaken in a week. However, we have no evidence that the relationship between age and the number of hours typical number of hours of vigorous exercise undertaken in a week, depended on smoking history.

Interpretation: We estimate that

exercise.df = heartHealth.df[c("exermin", "exerhour", "exerday", "smoke", "age")]  
  
#create exerHours field   
exerMinsInHours = exercise.df$exermin/60  
exerHoursPerDay = exercise.df$exerhour + exerMinsInHours  
exerHoursPerWeek = exercise.df$exerday \* exerHoursPerDay  
exercise.df$exerHoursPerWeek = exerHoursPerWeek  
  
#fit model  
exer.fit = glm(exerHoursPerWeek ~ age\*smoke, data=exercise.df, family="gaussian")  
summary(exer.fit)

##   
## Call:  
## glm(formula = exerHoursPerWeek ~ age \* smoke, family = "gaussian",   
## data = exercise.df)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -10.068 -4.035 -2.286 0.686 44.109   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.36621 3.76211 1.692 0.0933 .  
## age -0.09067 0.08471 -1.070 0.2867   
## smokeYes 7.77353 5.06345 1.535 0.1275   
## age:smokeYes -0.08635 0.11038 -0.782 0.4357   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 75.38409)  
##   
## Null deviance: 9500.7 on 117 degrees of freedom  
## Residual deviance: 8593.8 on 114 degrees of freedom  
## (2 observations deleted due to missingness)  
## AIC: 850.87  
##   
## Number of Fisher Scoring iterations: 2

#fit model - remove interaction  
exer.fit = glm(exerHoursPerWeek ~ age+smoke, data=exercise.df, family="gaussian")  
summary(exer.fit)

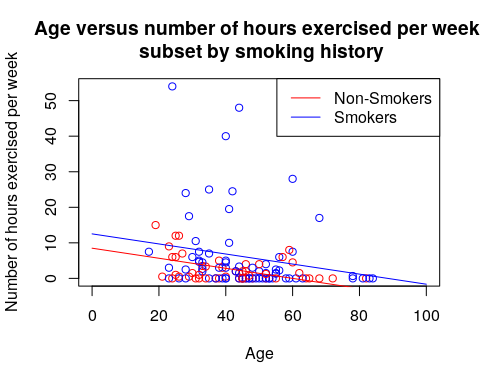
##   
## Call:  
## glm(formula = exerHoursPerWeek ~ age + smoke, family = "gaussian",   
## data = exercise.df)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -9.267 -4.315 -2.272 1.007 44.875   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 8.49572 2.59233 3.277 0.00139 \*\*  
## age -0.14152 0.05422 -2.610 0.01025 \*   
## smokeYes 4.02595 1.63675 2.460 0.01539 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 75.12972)  
##   
## Null deviance: 9500.7 on 117 degrees of freedom  
## Residual deviance: 8639.9 on 115 degrees of freedom  
## (2 observations deleted due to missingness)  
## AIC: 849.5  
##   
## Number of Fisher Scoring iterations: 2

confint(exer.fit)

## Waiting for profiling to be done...

## 2.5 % 97.5 %  
## (Intercept) 3.4148570 13.57658732  
## age -0.2477918 -0.03525431  
## smokeYes 0.8179667 7.23392379

#plot d   
pred.df <- data.frame(age=seq(0,100), smoke="Yes")  
smokersPredictions <- predict(exer.fit, newdata = pred.df)  
pred.df <- data.frame(age=seq(0,100), smoke="No")  
nonsmokersPredictions <- predict(exer.fit, newdata = pred.df)  
  
plot(exerHoursPerWeek ~ age, data = exercise.df, col = ifelse(exercise.df$smoke == "Yes" , "blue", "red"),xlim=c(0, 100),   
 main="Age versus number of hours exercised per week \n subset by smoking history", xlab="Age",   
 ylab = "Number of hours exercised per week")  
lines(pred.df$age, nonsmokersPredictions, col="red")  
lines(pred.df$age, smokersPredictions, col="blue")  
legend("topright", legend=c("Non-Smokers", "Smokers"),  
 col=c("red", "blue"), lty=1)



## Heart Attacks

heartattack.df = heartHealth.df[c("heartattack", "chol", "age")]  
ages <- as.numeric(levels(factor(heartattack.df$age)))  
chol <- levels(factor(heartattack.df$chol))  
  
# #group data  
# #success in this case is people have heartattacks and failures isnt   
# heartattack.grouped.df = data.frame()  
# names(heartattack.grouped.df) = c("success", "fail", "chol", "age")  
# for(cholLevel in chol){  
# for(age in ages){  
# #group data  
# mask <- (heartattack.df$chol == cholLevel) & (heartattack.df$age==age)  
# if (ncol(heartattack.df[mask]) > 0){  
#   
# }  
# break  
# }  
# break  
# }  
  
  
#fit model  
heartattack.fit = glm(heartattack ~ age\*chol, data=heartattack.df, family="binomial")  
summary(heartattack.fit)

##   
## Call:  
## glm(formula = heartattack ~ age \* chol, family = "binomial",   
## data = heartattack.df)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.10547 -0.25130 -0.14950 -0.09596 2.53261   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -12.9650 9.1288 -1.420 0.156  
## age 0.2063 0.1560 1.323 0.186  
## cholLow 4.7798 9.4066 0.508 0.611  
## age:cholLow -0.1151 0.1597 -0.720 0.471  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 53.366 on 119 degrees of freedom  
## Residual deviance: 34.657 on 116 degrees of freedom  
## AIC: 42.657  
##   
## Number of Fisher Scoring iterations: 7

#fit model - remove interaction  
heartattack.fit = glm(heartattack ~ age+chol, data=heartattack.df, family="binomial")  
summary(heartattack.fit)

##   
## Call:  
## glm(formula = heartattack ~ age + chol, family = "binomial",   
## data = heartattack.df)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.00569 -0.23428 -0.13777 -0.07972 2.57403   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -6.89477 2.07903 -3.316 0.000912 \*\*\*  
## age 0.10205 0.03351 3.046 0.002322 \*\*   
## cholLow -1.99378 0.96266 -2.071 0.038349 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 53.366 on 119 degrees of freedom  
## Residual deviance: 35.467 on 117 degrees of freedom  
## AIC: 41.467  
##   
## Number of Fisher Scoring iterations: 7

100\*(exp(confint(heartattack.fit))-1)

## Waiting for profiling to be done...

## 2.5 % 97.5 %  
## (Intercept) -99.999128 -96.063922  
## age 4.383572 19.667131  
## cholLow -98.216632 -7.210025

#plot d  
pred.df <- data.frame(age=seq(0,100), chol="High")  
highCholPredictions <- predict(heartattack.fit, newdata = pred.df)  
highCholPredictions = exp(highCholPredictions)  
pred.df <- data.frame(age=seq(0,100), chol="Low")  
lowCholPredictions <- predict(heartattack.fit, newdata = pred.df)  
lowCholPredictions = exp(lowCholPredictions)  
  
plot(heartattack ~ age, data = heartattack.df, col = ifelse(heartattack.df$chol == "High" , "blue", "red"),xlim=c(0, 100), ylim=c(0,10))  
lines(pred.df$age, lowCholPredictions, col="red")  
lines(pred.df$age, highCholPredictions, col="blue")

