MATH 2780 Chapter 7 Worksheet

Summary

In general, we need at least $p+1$ data points for a p^{th} order polynomial. $E(y) = \beta_0+\beta_1x+\beta_2x^2+\ldots+\beta_px^p$

If we want to estimate the residual variance, we need $n>p+1$. $s^2 = (y - X\hat{\beta})'(y - X\hat{\beta})/(n-p-1)$

Coefficient and Residual Variance Estimation:

$Y = X\beta + E$ $\hat{\beta} = (X'X)^{-1}X'y$ $s^2 = \dfrac{(y - X\hat{\beta})'(y - X\hat{\beta})}{n-k-1}$ $MSE = s^2,\ s = \sqrt{s^2}$	$Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}, X = \begin{bmatrix} 1 & x_{11} & x_{21} & \cdots & x_{kn} \\ 1 & x_{12} & x_{22} & \cdots & x_{k2} \\ 1 & x_{13} & x_{23} & \cdots & x_{k3} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{1n} & x_{2n} & \cdots & x_{kn} \end{bmatrix}, \beta = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix}, E = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix}$

Individual Coefficient Test: $t = \hat{\beta}_i / s_{\hat{\beta}_i}$, $s_{\hat{\beta}_i} = s\sqrt{W_{ii}}$, W_{ii} is the i^{th} diagonal of $W=(X'X)^{-1}$.

Two Tailed: H_0: $\beta_i=0$ *vs*. H_a: $\beta_i\neq 0$ w/ RR $|t|>t_{\alpha/2,n-k-1}$ or $\alpha>p\text{-}value=2P(|t|>t_{\alpha,n-k-1})$.

Coefficient of determination R^2 and R^2_a. Want simple model with large R^2 and R^2_a close full model.

$$R^2 = 1 - SSE/SS_{yy},\quad 0 \le R^2 \le 1,\ SSE = \sum (y_i - \hat{y}_i)^2,\ SS_{yy} = \sum (y_i - \bar{y})^2$$

$$R_a^2 = 1 - [SSE/(n-k-1)]/[SS_{yy}/(n-1)] = 1 - [(n-1)/(n-k-1)](1-R^2),\quad R_a^2 \le R^2$$

***F*-Test for Comparing Nested Models**

Reduced Model: $E(y|x's) = \beta_0+\beta_1x_1+\ldots+\beta_gx_g$

Complete Model: $E(y|x's) = \beta_0+\beta_1x_1+\ldots+\beta_gx_g+\beta_{g+1}x_{g+1}+\ldots+\beta_kx_k$

H_0: $\beta_{g+1}=\ldots=\beta_k=0$ *vs*. H_a: At least one tested $\beta_i \neq 0$.

$F = \dfrac{(SSE_R - SSE_C)/(k-g)}{SSE_C/(n-k-1)}$, Reject if $F>F_{\alpha,k-g,n-k-1}$ or $\alpha>p\text{-}value=P(F>F_{\alpha,k-g,n-k-1})$.

Multicollinearity exists when two or more of the independent variables used in regression are moderately or highly correlated. Correlation between x_i and x_j pairs and nonsignificant t-tests. With severe multicollinearity, the computer has difficulty inverting the information matrix ($X'X$).

Detecting Multicollinearity in the Regression Model

1. Significant correlations between pairs of independent variables in the model.
2. Nonsignificant t-tests for all (or nearly all) the individual β parameters when the F-test for overall model adequacy H_0: $\beta_1=\beta_2=\ldots=\beta_k=0$ is significant.
3. Opposite signs (from what is expected) in the estimated parameters.
4. A variance inflation factor (VIF) for a β parameter greater than 10, where $(VIF)_i=1/(1-R_i^2)$, $i=1,\ldots,k$ and R_i^2 is the multiple coefficient of determination for $E(x_i)=\alpha_0+\alpha_1x_1+\ldots+\alpha_{i-1}x_{i-1}+\alpha_3x_{i+1}+\ldots+\alpha_kx_k$.

Solutions to Some Problems Created by Multicollinearity

1. Drop one or more of the correlated x's. Stepwise regression is helpful in deciding which to drop.
2. If you decide to keep all the independent variables in the model:
   a. Avoid making inferences about the individual coefficient parameters.
   b. Restrict inferences about $E(y)$ and future y-values to the experimental region.
3. To establish cause-and-effect between y and the x's, use a designed experiment.
4. To reduce rounding errors in polynomial regression, code the x variables, $x_i^* = (x_i - \bar{x})/s_x$, so that the 1^{st}, 2^{nd}, and higher-order terms for a particular x are not highly correlated.
5. To reduce rounding errors and stabilize the regression coefficients, use ridge regression to estimate the β parameters. $\hat{\beta}_R = (X'X + cI)^{-1}X'y$, c often by cross validation but has Bayesian meaning.

Reliably interpolate between observations, but exercise caution forecasting outside observations.

Transformations: Transforming y and/or the x's in a model can provide a better model fit.

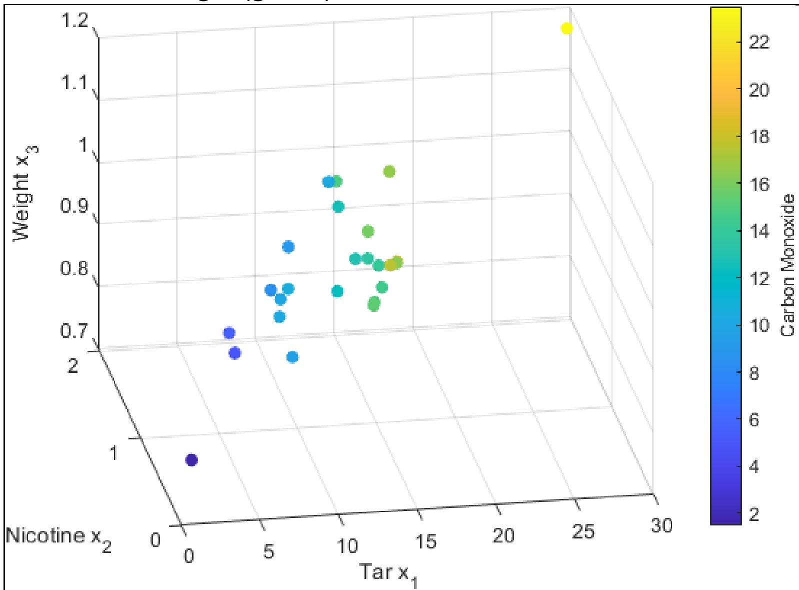
Example 7.5: Federal Trade Commission ranks cigarettes.

y = Carbon Monoxide Content (milligrams)

x_1 = Tar Content (milligrams)

x_2 = Nicotine Content (milligrams)

x_3 = Weight (grams)

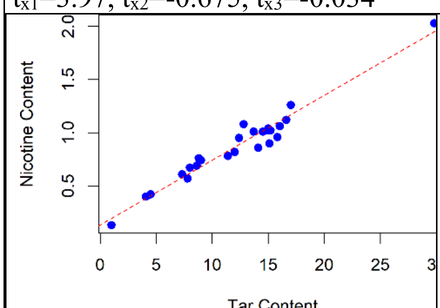


Tar	Nicotine	Weight	CO
14.1	0.86	0.9853	13.6
16	1.06	1.0938	16.6
29.8	2.03	1.165	23.5
8	0.67	0.928	10.2
4.1	0.4	0.9462	5.4
15	1.04	0.8885	15
8.8	0.76	1.0267	9
12.4	0.95	0.9225	12.3
16.6	1.12	0.9372	16.3
14.9	1.02	0.8858	15.4
13.7	1.01	0.9643	13
15.1	0.9	0.9316	14.4
7.8	0.57	0.9705	10
11.4	0.78	1.124	10.2
9	0.74	0.8517	9.5
1	0.13	0.7851	1.5
17	1.26	0.9186	18.5
12.8	1.08	1.0395	12.6
15.8	0.96	0.9573	17.5
4.5	0.42	0.9106	4.9
14.5	1.01	1.007	15.9
7.3	0.61	0.9806	8.5
8.6	0.69	0.9693	10.6
15.2	1.02	0.9496	13.9
12	0.82	1.1184	14.9

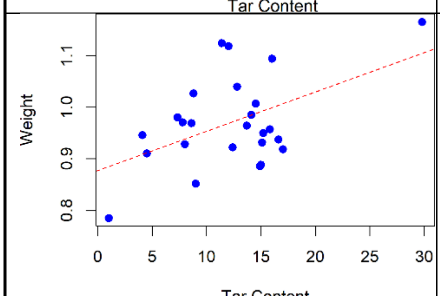
Model: $E(y) = \beta_0+\beta_1x_1+\beta_2x_3$

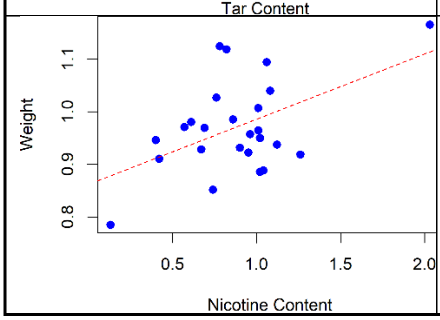
Fstat=79

t_{x1}=3.97, t_{x2}=-0.675, t_{x3}=-0.034



$$x_2 = 0.13087532+0.06102854x_1$$



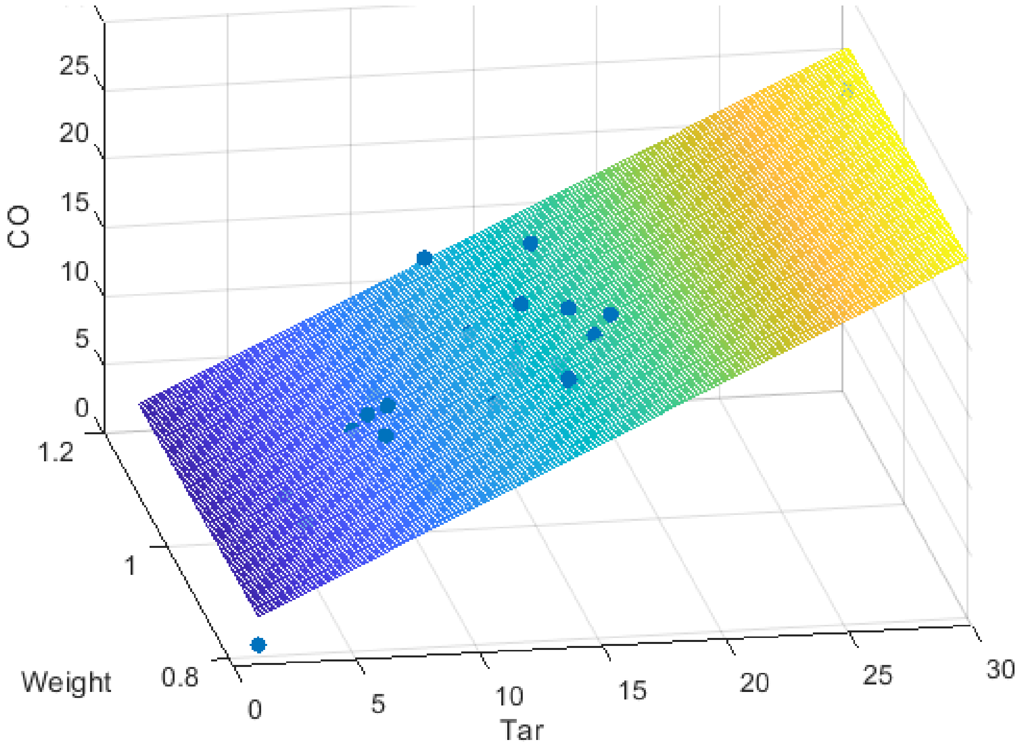


R=

```
          x1        x2        x3
x1 1.0000000 0.9766076 0.4907654
x2 0.9766076 1.0000000 0.5001827
x3 0.4907654 0.5001827 1.0000000
```

VIFs=

```
       x1        x2        x3
21.630706 21.899917  1.333859
```


CO
25
20
15
10
5
0
1.2
1
Weight
0.8
0
5
10
15
20
25
30
Tar

MATH 2780 Chapter 7 Worksheet

```
# R Code
# install.packages("car")
library(car)

# read data
mydata <- read.delim("ftccigar.txt",header=TRUE,sep="",dec=".")
head(mydata)

# parse out variables
n <- nrow(mydata)
k <- ncol(mydata)-1
x1 <- c(mydata[, 1]) #x1 tar content
x2 <- c(mydata[, 2]) #x2 nicotine content
x3 <- c(mydata[, 3]) #x3 weight
y  <- c(mydata[, 4]) #y  carbon monoxide

df        <- data.frame(cbind(x1,x2,x3))
names(df) <- c("x1","x2","x3")
head(df)

# scatter plot with line
plot(x1,y,xlab='Tar Content',     ylab='Carbon Monoxide',
     pch=19,col="blue")
abline(lm(y~x1),col='red',lty=2)
plot(x2,y,xlab='Nicotine Content',ylab='Carbon Monoxide',
     pch=19,col="blue")
abline(lm(y~x2),col='red',lty=2)
plot(x3,y,xlab='Weight',          ylab='Carbon Monoxide',
     pch=19,col="blue")
abline(lm(y~x3),col='red',lty=2)

# scatter plot
library("plot3D")
scatter3D(x1,x2,y,pch=19,cex=1,colvar=NULL,
          col="red",theta=20, phi=10,bty="b",
          xlab="Tar",ylab="Nicotine",zlab="Carbon Monoxide",
          main = "Cigarettes")
scatter3D(x1,x3,y,pch=19,cex=1,colvar=NULL,
          col="red",theta=20,phi=10,bty="b",
          xlab="Tar",ylab="Weight",  zlab="Carbon Monoxide",
          main = "Cigarettes")
scatter3D(x2,x3,y,pch=19,cex=1,colvar=NULL,
          col="red",theta=20,phi=10,bty="b",xlab="Nicotine",
          ylab = "Weight", zlab = "Carbon Monoxide",
          main = "Cigarettes")

# x1-x3 fit
lmx1to3<- lm(y~x1+x2+x3,data=df)
temp<-anova(lmx1to3)
out <- temp
n <- nrow(temp)
out$Df <- with(temp,c(sum(Df[1:(n-1)]),Df[n],rep(NA_real_,n-2)))
out$`Sum Sq` <- with(temp,c(sum(`Sum Sq`[1:(n-1)]),
                             `Sum Sq`[n],rep(NA_real_,n-2)))
out$`Mean Sq` <- with(out,out$`Sum Sq`/out$Df)
out$`F value` <- c(out$`Mean Sq`[1]/out$`Mean Sq`[2],rep(NA_real_,n-1))
out$`Pr(>F)` <- c(pf(out$`F value`[1],out$Df[1],out$Df[2],
                      lower.tail = FALSE),rep(NA_real_,n-1))
out <- out[1:2,]
rownames(out) <- c("Model","Residuals")
out

summary(lmx1to3)

# Calculating VIF
vif_values <- vif(lmx1to3)
vif_values
```

```
# correlation between variables
cor(df)

# scatter plot of x's with line
plot(x1,x2,xlab='Tar Content',   ylab='Nicotine Content',
      pch=19,col="blue")
abline(lm(x2~x1),col='red',lty=2)
plot(x1,x3,xlab='Tar Content',   ylab='Weight',
      pch=19,col="blue")
abline(lm(x3~x1),col='red',lty=2)
plot(x2,x3,xlab='Nicotine Content',   ylab='Weight',
      pch=19,col="blue")
abline(lm(x3~x2),col='red',lty=2)

install.packages("olsrr")
library(olsrr)
model = lm(y~.,data=df)
k=ols_step_all_possible(model,max_order = 3)
k

# worksheet
# since x1 and x2 highly correlated remove x2
linex1x2 <- lm(x2~x1)
coefficients(linex1x2)

# Compute the linear regression
fit <- lm(y~x1+x3)#+x1:x3
summary(fit)

# create a grid from the x and y values and predict values
# for every point, this will become the regression plane
grid.lines = 40
x1.pred <- seq(min(x1),max(x1),length.out=grid.lines)
x3.pred <- seq(min(x3),max(x3),length.out=grid.lines)
x1x3   <- expand.grid(x1=x1.pred,x3=x3.pred)
y.pred <- matrix(predict(fit,newdata=x1x3),
                 nrow=grid.lines,ncol=grid.lines)

# create the fitted points for droplines to the surface
fitpoints <- predict(fit)
# scatter plot with regression plane
library("plot3D")
scatter3D(x1,x3,y,pch=19,cex=1,colvar=NULL,col="red",
          theta=20,phi=10,bty="b",grid=TRUE,col.grid="grey",
          xlab="Tar",ylab="Weight",zlab="CO",
          surf = list(x=x1.pred,y=x3.pred,z=y.pred,
                      facets=TRUE,fit=fitpoints,col=ramp.col
                      (col=c("dodgerblue3","seagreen2"),n=300,alpha=0.5),
                      border="black"))#,main="Cigarettes"

summary(mydata)
```

```
% Matlab Code
load cigarette.txt

n=size(cigarette,1);
k=size(cigarette,2)-1;
y =ftccigar(:,4);
x1=ftccigar(:,1);
x2=ftccigar(:,2);
x3=ftccigar(:,3);

figure;
scatter3(x1,x2,x3,40,y,'filled')
xlabel('Tar x_1'),ylabel('Nicotine x_2'),zlabel('Weight x_3')
cb = colorbar; % create and label the colorbar
cb.Label.String = 'Carbon Monoxide';
view([-10,30])
%print(gcf,'-dtiffn','-r100','cigarettes')

% remove x2
X=[ones(n,1),x1,x3];
k=size(X,2)-1;
% estimate coefficients
mdl = fitlm([x1,x3],y)
mdltable=anova(mdl,'summary')
MSE=mdltable.MeanSq(3,1);

% 3D plot
figure;
scatter3(x1,x3,y,'filled')
hold on
x1fit = min(x1):(max(x1)-min(x1))/100:max(x1);
x3fit = min(x3):(max(x3)-min(x3))/100:max(x3);
[X1FIT,X3FIT] = meshgrid(x1fit,x3fit);
b0=mdl.Coefficients(1,1).(1);
b1=mdl.Coefficients(2,1).(1);
b2=mdl.Coefficients(3,1).(1);
YFIT = b0 + b1*X1FIT + b2*X3FIT;
mesh(X1FIT,X3FIT,YFIT,'FaceAlpha','0.5')
xlabel('Tar'), ylabel('Weight'), zlabel('CO')
view(-10,30)
hold off
%print(gcf,'-dtiffn','-r100','cigarettesFit')
```