Name: LP Dangal

Mini project: #2

---

1. Consider the prostate cancer dataset available on eLearning as `prostate_cancer.csv`. It consists of data on 97 men with advanced prostate cancer. A description of the variables is given in Figure 1. We would like to understand how PSA level is related to the other predictors in the dataset. Note that `vesinv` is a qualitative variable. You can treat `gleason` as a quantitative variable.

header	name	description
subject	ID	1 to 97
psa	PSA level	Serum prostate-specific antigen level (mg/ml)
cancervol	Cancer Volume	Estimate of prostate cancer volume (cc)
weight	Weight	prostate weight (gm)
age	Age	Age of patient (years)
benpros	Benign prostatic hyperplasia	Amount of benign prostatic hyperplasia (cm^2)
vesinv	Seminal vesicle invasion	Presence (1) or absence (0) of seminal vesicle invasion
capspen	Capsular penetration	Degree of capsular penetration (cm)
gleason	Gleason score	Pathologically determined grade of disease (6, 7 or 8)

Figure 1: List of variables in the prostate cancer data

.

(a) Perform an exploratory analysis of data.

(b) Is `psa` appropriate as a response variable or a transformation is necessary? In case a transformation of response is necessary, try the natural log transformation or some other transformation and use it for the rest of this problem.

(c) Do part (a) of Exercise 15 in Chapter 3 for these data.

(d) Do part (b) of Exercise 15 in Chapter 3 for these data.

(e) Build a "reasonably good" multiple regression model for these data. Carefully justify all the choices you make in building the model. Be sure to verify the model assumptions.

(f) Write the final model in equation form, being careful to handle qualitative predictors (if any) properly.

(g) Use the final model to predict the PSA level for a patient whose quantitative predictors are at the sample means of the variables and qualitative predictors (if any) are at the most frequent category.

(a) Perform an exploratory analysis of data.

Exploratory Analysis: This means analysing the datasets to summarize their main characteristics, often visually. In short, Exploratory Analysis means "Understanding data visually". Following four steps for exploratory analysis of data.

#1)Understanding the data

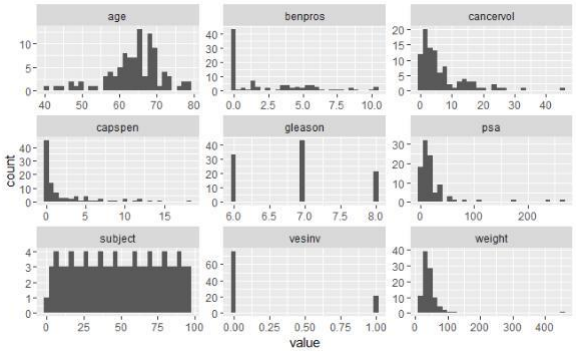
#load data

```
> prostate <- read.csv("C:/Users/LPD/Desktop/Stat ML Pankaj/Project/project2/prostate_cancer.csv",head=TRU
E, sep=",")
> attach(prostate)

> #know the dimensions of the data
> dim(prostate)
[1] 97 9

> #know the column names
> colnames(prostate)
[1] "subject"  "psa"   "cancervol" "weight"  "age"   "benpros"  "vesinv"  "capspen"
[9] "gleason"

> #know the data types of each variable
> str(prostate)
'data.frame':                97 obs. of 9 variables:
 $ subject : int 1 2 3 4 5 6 7  8910...
 $ psa   : num 0.651 0.852     0.852 0.852 1.448 ...
 $ cancervol: num 0.56 0.372 0.601 0.301 2.117 ...
 $ weight  : num 16 27.7 14.7 26.6 30.9 ...
 $ age     : int 50 58 74 58 62 50 64 58 47 63 ...
 $ benpros : num 0 0 0 0 0 ...
 $ vesinv  : int 0 0 0 0 0 0 0 0 0 0 ...
 $ capspen : num 0 0 0 0 0 0 0 0 0 0 ...
 $ gleason : int 6 7 7 6 6 6 6 6 7 6 ...
```

Here, all data are shown in numerical and integer data types. But according to questions, vesinv is categorical so w e need to factor it.

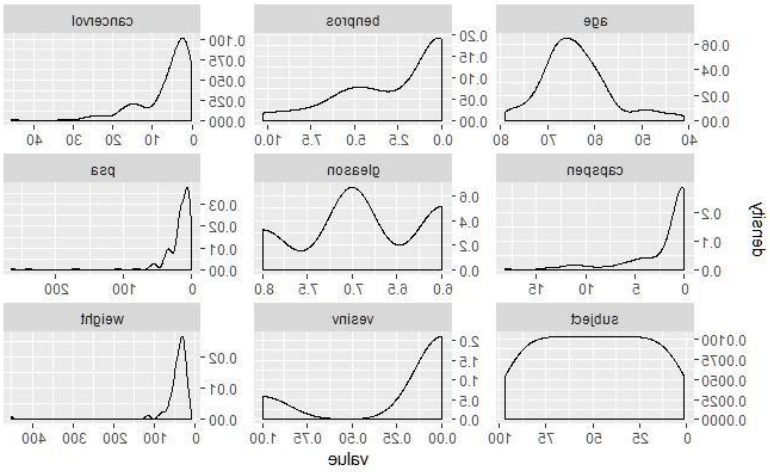
```
> head(prostate)
  subject  psa cancervol weight age benpros vesinv capspen gleason
1       1 0.651   0.5599 15.959  50       0      0       0       6
2       2 0.852   0.3716 27.660  58       0      0       0       7
3       3 0.852   0.6005 14.732  74       0      0       0       7
4       4 0.852   0.3012 26.576  58       0      0       0       6
5       5 1.448   2.1170 30.877  62       0      0       0       6
6       6 2.160   0.3499 25.280  50       0      0       0       6

 [1] "cancervol" "weight"  "age"   "benpros"  "vesinv"
[6] "capspen"  "gleason"
```

```
> #checking NA(missing values) in datasets
> anyNA(prostate)
[1] FALSE
  > colSums(sapply(prostate, is.na))
   subject      psa cancervol   weight      age   benpros    vesinv  capspen  gleason
         0        0         0        0        0         0         0        0        0

> #2) data type conversion like int into factor
> vesinv<-factor(vesinv)
> str(vesinv)
 Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...

> #3)Summary of each and every variable
> summary(prostate)
   subject        psa             cancervol        weight          age             benpros
 Min.   : 1   Min.   : 0.651   Min.   : 0.2592   Min.   : 10.70   Min.           :41.00   Min.      : 0.000
 1st Qu.:25   1st Qu.: 5.641   1st Qu.: 1.6653   1st Qu.: 29.37   1st Qu.:60.00         1st Qu.: 0.000
 Median :49   Median : 13.330   Median : 4.2631      Median : 37.34   Median :65.00   Median : 1.350
 Mean   :49   Mean   : 23.730   Mean   : 6.9987   Mean   : 45.49      Mean   :63.87   Mean   : 2.535
 3rd Qu.:73   3rd Qu.: 21.328   3rd Qu.: 8.4149   3rd Qu.: 48.42   3rd Qu.:68.00   3rd Qu.: 4.759
 Max.   :97   Max.   :265.072   Max.   :45.6042   Max.   :450.34   Max.   :79.00   Max.   :10.278
    vesinv         capspen           gleason
 Min.   :0.0000   Min.   : 0.0000   Min.   :6.000
 1st Qu.:0.0000   1st Qu.: 0.0000   1st Qu.:6.000
 Median :0.0000   Median : 0.4493   Median :7.000
 Mean   :0.2165   Mean   : 2.2454   Mean   :6.876
 3rd Qu.:0.0000   3rd Qu.: 3.2544   3rd Qu.:7.000
 Max.   :1.0000   Max.   :18.1741   Max.   :8.000



> #4)visualization
> #histogram of all variables
> library(purrr)
> library(tidyr)
> library(ggplot2)

> prostate %>%
+   keep(is.numeric) %>%
+   gather() %>%
+   ggplot(aes(value)) +
+   facet_wrap(~ key, scales = "free") +
+   geom_histogram()
`
```



Histogram showing the nature of predictors whether they are normally distributed or not. Here no any predictor s hows normality.

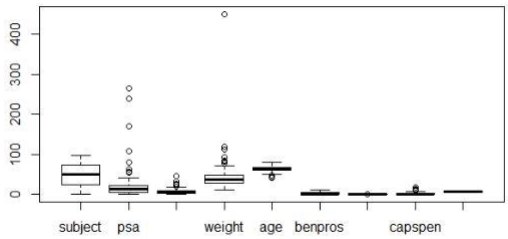
```
> #density plot
> prostate %>%
  + keep(is.numeric) %>%                  # Keep only numeric columns
  + gather() %>%                          # Convert to key-value pairs
  + ggplot(aes(value)) +                  # Plot the values
  + facet_wrap(~ key, scales = "free") +  # In separate panels
  + geom_density()
```



Subject is identiy, it has nothing to do with all predictors. Psa, cancervol, vesinv, weight, benpors, vesinv and caps pen are highly skewed.Gleason and Age have moderate skewness. This means most of our data will need to be tra nsformed.

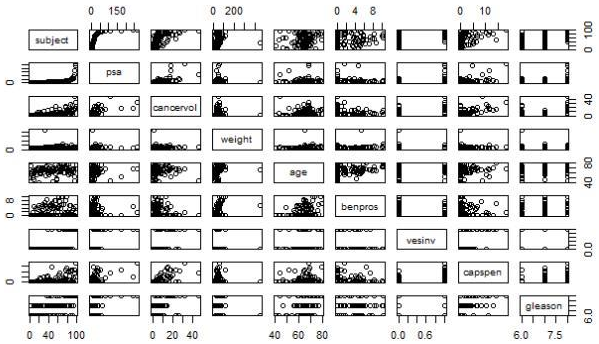
#Identifying outliers

```
> boxplot(prostate)
```

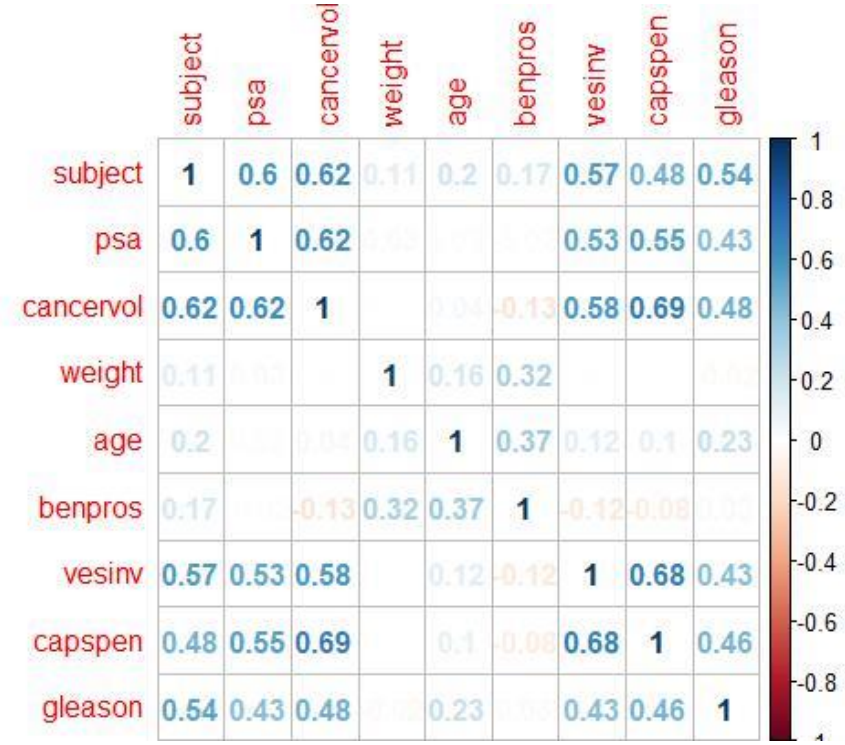


Weight appears to have an outlier.

> #scatterplot matrix
> pairs(prostate)



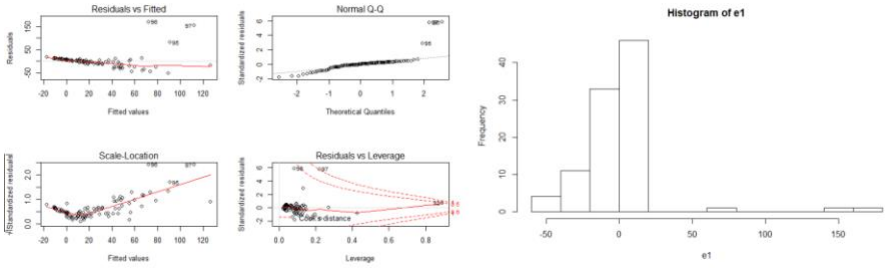
#correlation
> require(corrplot)
> corvalue<-cor(prostate)
> corrplot(corvalue, method="number")



PSA appears to be correlated moderately to subject, cancervol, vesinv, capspen and geason while it is rarely correlated with weight, age and benpros.

b) Is psa appropriate as a response variable or a transformation is necessary? In case a transformation of response is necessary, try the natural log transformation or some other transformation and use it for the rest of this problem.

```
> #b) checking response varaible, transformation needs or not by using residual plot and histogram of response
> modfirst<-lm(psa~., data=prostate)
> par(mfrow=c(2,2))
> plot(modfirst)
```



Above diagnostic plots shows that errors are not normal and variance decreases when predictor increases. So, we need to transform any variables.

```
> e1<-resid(modfirst)
> hist(e1)
> shapiro.test(e1)
```

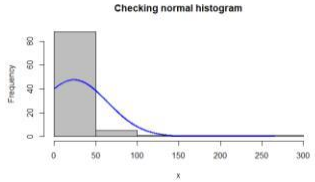
Shapiro-Wilk normality test

data: e1
W = 0.62576, p-value = 2.32e-14

Normality assumptions also failed as p-values<0.05 rejecting null hypothesis and hence distribution of errors are n ot normal.

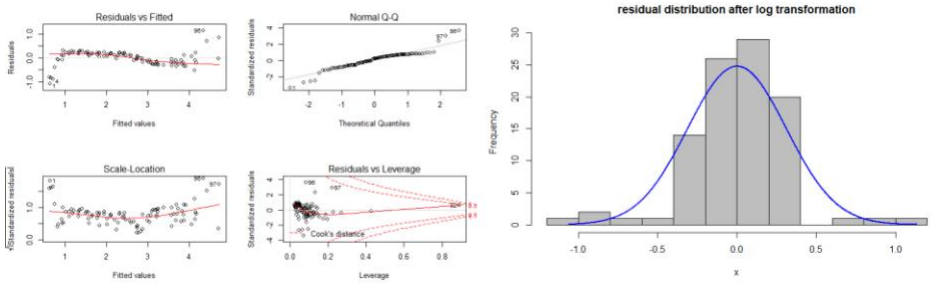
#Normality checking for response variables

```
> library(rcompanion)
> plotNormalHistogram(psa, main="Checking normal histogram")
```



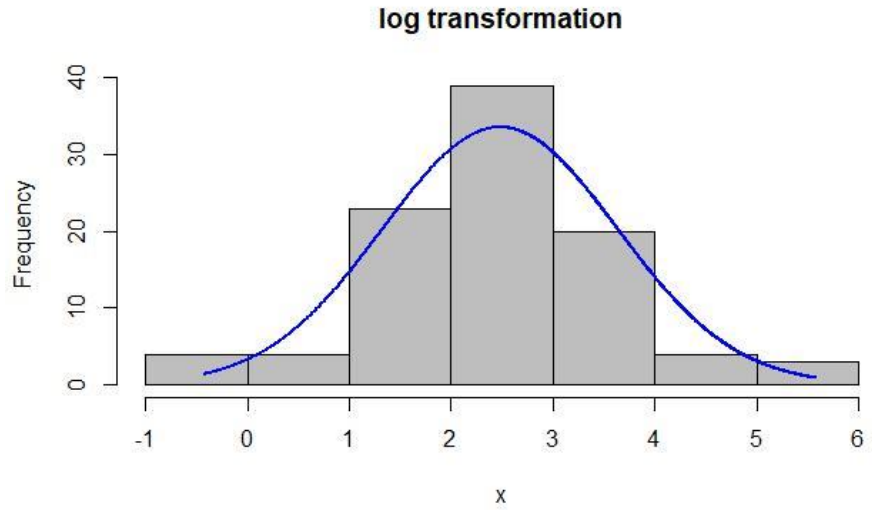
Generally, for right-skewed data, common transformation include square root, cute root and log and for left-skewed data, common transformation include square root(constant-x), cube root(constant-x) and log(constant-x) . Others are Tukey's Ladder of Powers or Box-Cox transformation.

```
> #checking after transformation
modsecond<-lm(log(psa)~., data=prostate)
> par(mfrow=c(2,2))
> plot(modsecond)

> plotNormalHistogram(resid(modsecond), main="residual distribution
after log transformation")
```



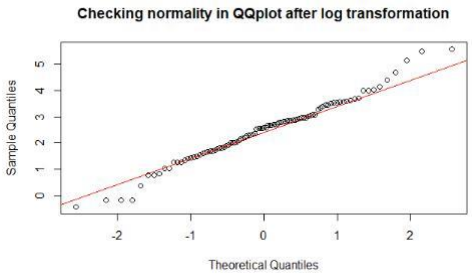
After log transformation of response variable, the distribution of residual looks normal and most of data points are seen near Fitted line which means model might be good however we need to test it statistically for homoscedasticity, normality, linearity and independence.

```
# Distribution of psa after log transformation
> plotNormalHistogram(log(psa), main="log transformation")
```



It appears after log transformation, response psa shows normal distribution.
So, drawing QQplot and use shaper-test to verify.

```
> qqnorm(log(psa), main="Checking normality in QQplot after log transformation")
> qqline(log(psa), col="red")
```




```
> #normality test after log transformation
> shapiro.test(log(psa))

Shapiro-Wilk normality test

data: log(psa)
W = 0.98442, p-value = 0.3082
```

The p-values>0.05 implying that it failed to reject null hypothesis and hence we can assume the normality.

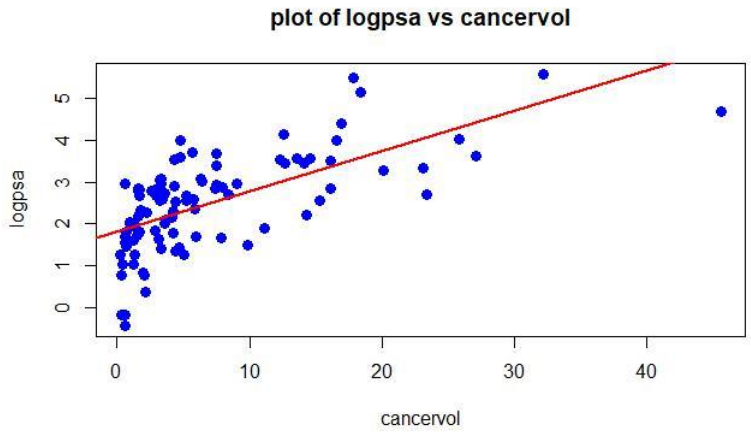
c)For each predictor, fit a simple linear regression model to predict the response. Describe your results. In which of the models is there a statistically significant association between the predictor and the response? Create some plots to back up your assertions.

```
> logpsa<-log(psa)

> mod_cancervol<-lm(logpsa~cancervol)
> summary(mod_cancervol)

Call:
lm(formula = logpsa ~ cancervol)

Residuals:
    Min      1Q  Median      3Q     Max
-2.2886 -0.6590  0.1493  0.5769  1.9610

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.80549    0.11899  15.174 < 2e-16 ***
cancervol   0.09619    0.01132   8.496 2.69e-13 ***
---
```

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

Residual standard error: 0.8742 on 95 degrees of freedom
Multiple R-squared: 0.4317, Adjusted R-squared: 0.4258
F-statistic: 72.18 on 1 and 95 DF, p-value: 2.688e-13

```
> plot(logpsa~cancervol, pch=16, cex=1.3,col="blue", main="plot of logpsa vs cancervol")
> abline(mod1, col="red", lwd=2)
```




```
>
> mod_weight<-lm(logpsa~weight)
> summary(mod_weight)
```

Call:
lm(formula = logpsa ~ weight)

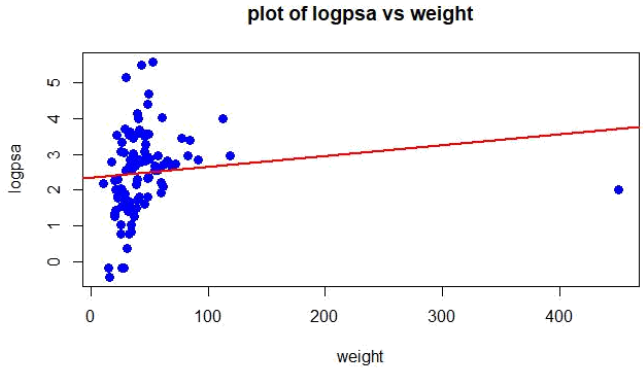
Residuals:
Min 1Q Median 3Q Max
-2.8172 -0.7291 0.1300 0.6144 3.0783

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.338901 0.165328 14.147 <2e-16 ***
weight 0.003072 0.002570 1.195 0.235

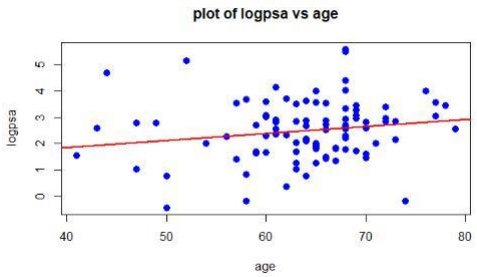
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.151 on 95 degrees of freedom
Multiple R-squared: 0.01482, Adjusted R-squared: 0.004446
F-statistic: 1.429 on 1 and 95 DF, p-value: 0.235

```
> plot(logpsa~weight, pch=16, cex=1.3,col="blue", main="plot of logpsa vs weight")
> abline(mod_weight, col="red", lwd=2)
```



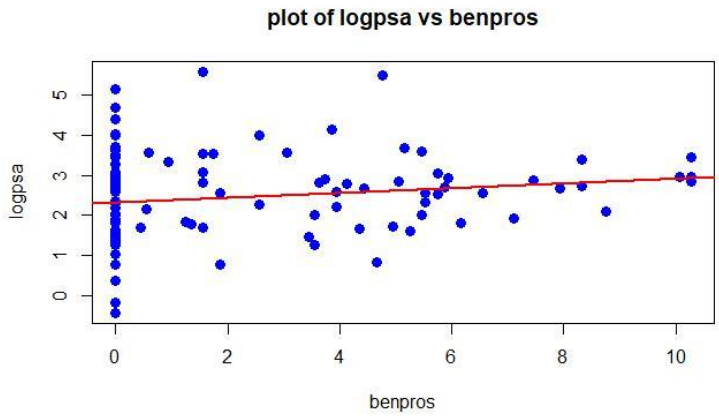

```
> mod_age<-lm(logpsa~age)
> summary(mod_age)
Call:
lm(formula = logpsa ~ age)

Residuals:
    Min      1Q  Median      3Q     Max
-2.90564 -0.71115 0.07247 0.66617 2.99249

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.79721    1.00729  0.791  0.4307
age         0.02633    0.01567  1.680  0.0961 .
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.143 on 95 degrees of freedom
Multiple R-squared: 0.02887,    Adjusted R-squared: 0.01865
F-statistic: 2.824 on 1 and 95 DF, p-value: 0.09615

> plot(logpsa~age, pch=16, cex=1.3,col="blue", main="plot of logpsa vs age")
> abline(mod_age, col="red", lwd=2)
```




```
> mod_benpros<-lm(logpsa~benpros)
> summary(mod_benpros)
Call:
lm(formula = logpsa ~ benpros)

Residuals:
    Min      1Q  Median      3Q     Max
-2.75607 -0.76149 -0.01686 0.63318 3.16016

Coefficients:
```

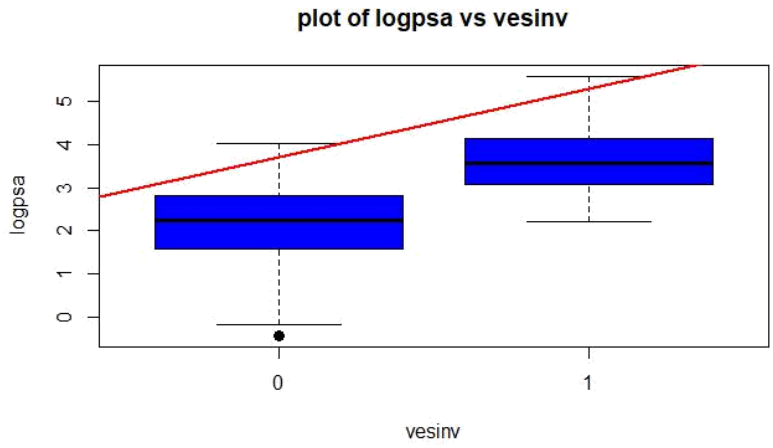
```
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.32682    0.15191 15.317    <2e-16 ***
benpros     0.05991    0.03856  1.554     0.124
---
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.145 on 95 degrees of freedom
Multiple R-squared: 0.02478,    Adjusted R-squared: 0.01451
F-statistic: 2.413 on 1 and 95 DF, p-value: 0.1236
```

```
> plot(logpsa~benpros, pch=16, cex=1.3,col="blue", main="plot of logpsa vs benpros")
> abline(mod_benpros, col="red", lwd=2)
```

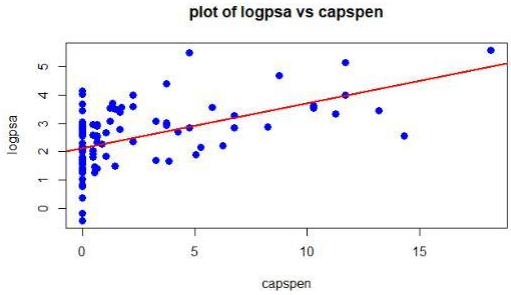



```
> mod_vesinv<-lm(logpsa~vesinv)
> summary(mod_vesinv)
Call:
lm(formula = logpsa ~ vesinv)

Residuals:
    Min      1Q  Median      3Q     Max
-2.56623 -0.63526 -0.00524 0.67302 1.89302

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.1370     0.1096 19.492 < 2e-16 ***
vesinv1      1.5783     0.2356  6.698 1.48e-09 ***
---
Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9558 on 95 degrees of freedom
Multiple R-squared: 0.3208,     Adjusted R-squared: 0.3136
F-statistic: 44.86 on 1 and 95 DF, p-value: 1.481e-09
```

```
> plot(logpsa~vesinv, pch=16, cex=1.3,col="blue", main="plot of logpsa vs vesinv")
> abline(mod_vesinv, col="red", lwd=2)
```



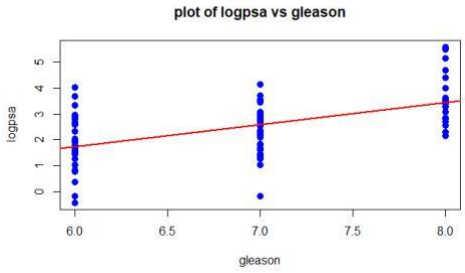

```
> mod_capspen<-lm(logpsa~capspen)
> summary(mod_capspen)
Call:
lm(formula = logpsa ~ capspen)

Residuals:
    Min      1Q  Median      3Q     Max
-2.5532 -0.6740  0.0071  0.6660  2.6043

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.12399    0.11728  18.110  < 2e-16 ***
capspen     0.15796    0.02676   5.903  5.5e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.992 on 95 degrees of freedom
Multiple R-squared: 0.2683,     Adjusted R-squared: 0.2606
F-statistic: 34.84 on 1 and 95 DF,  p-value: 5.503e-08

> plot(logpsa~capspen, pch=16, cex=1.3,col="blue", main="plot of logpsa vs capspen")
> abline(mod_capspen, col="red", lwd=2)
```




```
> mod_gleason<-lm(logpsa~gleason)
> summary(mod_gleason)
Call:
lm(formula = logpsa ~ gleason)
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-2.7428 -0.6134  0.0773  0.4773  2.2881

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -3.3026     0.9322  -3.543 0.000616 ***
gleason      0.8408     0.1348   6.237 1.23e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.9768 on 95 degrees of freedom
Multiple R-squared:  0.2905,         Adjusted R-squared:  0.2831
F-statistic:  38.9 on 1 and 95 DF,  p-value: 1.228e-08
```

```
> plot(logpsa~gleason, pch=16, cex=1.3,col="blue", main="plot of logpsa vs gleason")
> abline(mod_gleason, col="red", lwd=2)
```



It is seen that only *cancervol, vesinv, capspen and gleason* predictors are statistically significant in predicting the effect of psa level by rejecting null hypothesis since p-values<0.05. R squared values for weight, age and benpros have less than 2% indicating theses predictors explained only very low percent (below 2%)of variation in response psa.

For single predictors t-test and F-test are equivalent and hence they gave the same p-values in all separate models.

Residual standard error values indicate the average amount that the response will deviate from the true regression line even it the model is correct. RSE values for cancervol, vesinv, capspen and gleason are very low (less than 1 heare) indicating predictions obtained using the model are very close to the true outcome values. We can conclude that model fits the data well for theses predictors only.

d)Fit a multiple regression model to predict the response using all of the predictors. Describe your results. For which predictors can we reject the null hypothesis $H_0 : \beta_j = 0$?

```
> #d) fitting multiple regression model
> model_full<-lm(logpsa ~ cancervol+weight+age+benpros+vesinv+capspen+gleason)
> summary(model_full)
Call:
lm(formula = logpsa ~ cancervol + weight + age + benpros + vesinv
    + capspen + gleason)

Residuals:
     Min       1Q   Median       3Q      Max
-1.88309 -0.46629  0.08045  0.47380  1.53219

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.685796   0.998754  -0.687  0.49409
cancervol    0.069454   0.014624   4.749 7.77e-06 ***
weight       0.001380   0.001822   0.757  0.45079
age         -0.002799   0.011724  -0.239  0.81186
benpros      0.087470   0.029605   2.955  0.00401 **
vesinv1      0.782623   0.268339   2.917  0.00448 **
capspen     -0.026521   0.032860  -0.807  0.42177
gleason      0.358153   0.127976   2.799  0.00629 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7679 on 89 degrees of freedom
Multiple R-squared:  0.5893,    Adjusted R-squared:  0.557
F-statistic: 18.24 on 7 and 89 DF,  p-value: 7.694e-15

> sigma(model_full)/mean(logpsa)
[1] 0.3097914
```

F statistic>1 and it's corresponding p-values<0.05 indicating it rejects the null hypothesis and hence confirmed that at least one of the predictors are significant and associated with psa level . To see which predictor variables are significant, we can look the estimated regression coefficients and associated t-statistic p-values.

Here, we reject null hypothesis for these cancervol, benpros, vesinv and gleason and they are statistically significant and associated with psa level since their p-values<0.05. Other remaining predictors don't have a relationship with psa level as these failed to reject null hypothesis because their p-values>0.05 and hence estimated coefficient are not far from zero.

Negative values of coefficients indicate the average amount of psa level decreases by unit increase in corresponding predictors holding all other predictors fixed. Negative intercept means psa level is already negative when there are no predictors.

Benpros is not significant when regressed individually ignoring all other predictors in question (C) but here benpros is statistically significant while adjusting all other variables.

Also, Capspen is statistically significant when treating individually ignoring all other predictors in question (C) but it is classified as statistically less significant predictors while holding all other predictors fixed. Due to the estimated coefficient less than zero for capspen, average effect on psa level by unit increase in capspen is not significant while holding all other predictors fixed. This is due to multicollinearity between predictors.

Here, RSE is 0.7679 corresponding to 31% error rate.

(e) Build a reasonably good" multiple regression model for these data. Carefully justify all the choices you make in building the model. Be sure to verify the model assumptions.

We have four predictors i.e.*cancervol, benpros, vesinv and gleason* are significant on the basis of multiple linear regression model with considerations of all other predictors. To get a reasonably good model, we need to use partial F test using anova i.e. comparing full and reduced model so that we can remove some statistical insignificant predictors if possible.

```
> #choosing reasonably good predictors by using partial F-test
> fit7<-lm(logpsa~cancervol+benpros+vesinv+gleason+capspen+age+weight)
> fit4<-lm(logpsa~cancervol+benpros+vesinv+gleason)
> fit3<-lm(logpsa~cancervol+benpros+vesinv)
>
> fit3<-lm(logpsa~cancervol+benpros+gleason)
>
> fit333<-lm(logpsa~cancervol+vesinv+gleason)
> fit3333<-lm(logpsa~gleason+vesinv+benpros)

#checking if we can drop previously chosen insignificant variable using anova
> anova(fit7, fit4) Analysis
of Variance Table

Model 1: logpsa ~ cancervol + benpros + vesinv + gleason + capspen + age +
    weight
Model 2: logpsa ~ cancervol + benpros + vesinv +
  gleason Res.Df RSS Df Sum of Sq F Pr(>F)
1     89 52.477
2     92 53.229 -3 -0.75232 0.4253 0.7353
```

Since p-values>0.05, it failed to reject null hypothesis and it says that reduced model is equal to full model. Hence three predicotrs capspen, weight and age can be dropped from the model and we left only 4 significant predictors.

```
#checking if we can drop any predictors among 4 using anova
> anova(fit4, fit3) Analysis
of Variance Table

Model 1: logpsa ~ cancervol + benpros + vesinv + gleason
Model 2: logpsa ~ cancervol + benpros + gleason
  Res.Df   RSS Df Sum of Sq      F  Pr(>F)
1     92 53.229
```

```
2    93 58.075 -1  -4.8466 8.3767 0.004746 **
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> anova(fit4, fit33)
Analysis of Variance Table

Model 1: logpsa ~ cancervol + benpros + vesinv + gleason
Model 2: logpsa ~ cancervol + benpros + gleason
  Res.Df   RSS Df Sum of Sq      F  Pr(>F)
1    92 53.229
2    93 58.075 -1  -4.8466 8.3767 0.004746 **
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> anova(fit4, fit333)
Analysis of Variance Table

Model 1: logpsa ~ cancervol + benpros + vesinv + gleason
Model 2: logpsa ~ cancervol + vesinv + gleason
  Res.Df   RSS Df Sum of Sq      F    Pr(>F)
1    92 53.229
2    93 60.340 -1  -7.1115 12.291 0.0007054 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> anova(fit4,fit3333)
Analysis of Variance Table

Model 1: logpsa ~ cancervol + benpros + vesinv + gleason
Model 2: logpsa ~ gleason + vesinv + benpros
  Res.Df   RSS Df Sum of Sq      F  Pr(>F)
1    92 53.229
2    93 67.987 -1  -14.758 25.508 2.22e-06 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

All anova results showed that all their p-values<0.05, rejecting null hypothesis and hence all predictors are significant and we should keep it in the model.

#checking different interaction by using all possible model

```
> #lm(y~x1*x2) is same as lm(y~x1+x2 +x1:x2)
> #checking for intereaction
> fitI1<-lm(logpsa~cancervol*benpros)
> fitI2<-lm(logpsa~cancervol*vesinv)
> fitI3<-lm(logpsa~cancervol*gleason)
> fitI4<-lm(logpsa~benpros*vesinv)
> fitI5<-lm(logpsa~benpros*gleason)
> fitI6<-lm(logpsa~vesinv*gleason)
> anova(fitI1)
Analysis of Variance Table
Response: logpsa
                  Df Sum Sq Mean Sq F value    Pr(>F)
cancervol          1 55.164 55.164 79.251 4.293e-14 ***
benpros            1 7.803  7.803 11.211 0.001175 **
cancervol:benpros 1 0.068  0.068  0.098 0.754926
Residuals          93 64.733  0.696
---
```

```
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> anova(fitI2)
Analysis of Variance Table

Response: logpsa
                Df Sum Sq Mean Sq F value    Pr(>F)
cancervol        1 55.164 55.164 81.4680 2.352e-14 ***
vesinv           1 6.547   6.547 9.6686 0.002488 **
cancervol:vesinv  1  3.086  3.086  4.5576  0.035403 *
Residuals        93 62.972  0.677
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> anova(fitI3)
Analysis of Variance Table

Response: logpsa
                Df Sum Sq Mean Sq F value  Pr(>F)
cancervol        1 55.164 55.164 82.2830 1.889e-14 ***
gleason          1 8.247   8.247 12.3011 0.0006991 ***
cancervol:gleason  1  2.010  2.010  2.9977 0.0866986 .
Residuals        93 62.349  0.670
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> anova(fitI4)
Analysis of Variance Table

Response: logpsa
              Df Sum Sq Mean Sq F value    Pr(>F)
benpros        1 3.166  3.166  3.9035    0.05115 .
vesinv         1 44.387 44.387 54.7361 5.998e-11 ***
benpros:vesinv  1 4.799  4.799  5.9181    0.01690 *
Residuals      93 75.417  0.811
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> anova(fitI5)
Analysis of Variance Table

Response: logpsa
                Df Sum Sq Mean Sq F value    Pr(>F)
benpros          1 3.166  3.166 3.3982  0.06845 .
gleason          1 36.569 36.569 39.2575 1.144e-08 ***
benpros:gleason 1 1.404  1.404 1.5071  0.22268
Residuals        93 86.631  0.932
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
> anova(fitI6)
Analysis of Variance Table

Response: logpsa
              Df Sum Sq Mean Sq F value    Pr(>F)
vesinv         1 40.984 40.984 52.3553 1.294e-10 ***
gleason        1 13.740 13.740 17.5523 6.355e-05 ***
vesinv:gleason 1 0.243  0.243 0.3108       0.5785
Residuals      93 72.801  0.783
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It is seen that two interaction are significant i.e. cancervol:vesinv and benpros:vesinv. So we need to check these two interaction are significant or not in the presence of other four predictors.

```
> #multiple regression model using interaction
> mod3<-lm(logpsa~cancervol+benpros+vesinv+gleason+cancervol:vesinv)
> mod2<-lm(logpsa~cancervol+benpros+vesinv+gleason+benpros:vesinv)
> mod1<-lm(logpsa~cancervol+benpros+vesinv+gleason)
> anova(mod1,mod2)
Analysis of Variance Table

Model 1: logpsa ~ cancervol + benpros + vesinv + gleason
Model 2: logpsa ~ cancervol + benpros + vesinv + gleason + benpros:vesinv
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     92 53.229
2     91 51.291 1  1.9379 3.4383 0.06694 .
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> anova(mod1,mod3)
Analysis of Variance Table

Model 1: logpsa ~ cancervol + benpros + vesinv + gleason
Model 2: logpsa ~ cancervol + benpros + vesinv + gleason + cancervol:vesinv
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     92 53.229
2     91 51.417 1  1.8124 3.2077 0.07662 .
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It is seen that the both interaction terms are not significant in the presence of other four significant predictors and should not be included in the final model as all p-values>0.05 accepting null hypothesis. So, mod1 is final model a nd we need to verify the regression model assumptions now and do some diagnostic.

#Final model

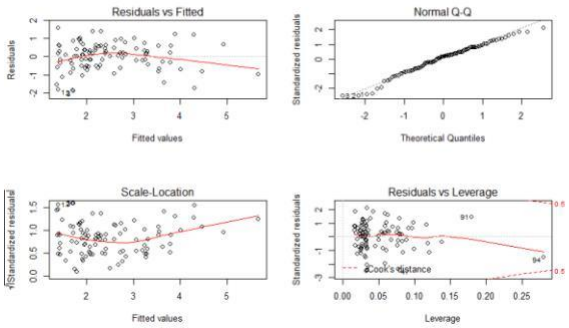
```
> #final model
> mod1<-lm(logpsa~cancervol+benpros+vesinv+gleason)
> summary(mod1)

Call:
lm(formula = logpsa ~ cancervol + benpros + vesinv + gleason)

Residuals:
    Min      1Q  Median      3Q     Max
-1.88531 -0.50276 0.09885 0.53687 1.56621

Coefficients:
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.65013    0.80999 -0.803 0.424253
cancervol    0.06488    0.01285  5.051 2.22e-06 ***
benpros      0.09136    0.02606  3.506 0.000705 ***
vesinv1      0.68421    0.23640  2.894 0.004746 **
gleason      0.33376    0.12331  2.707 0.008100 **
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7606 on 92 degrees of freedom
Multiple R-squared: 0.5834,     Adjusted R-squared: 0.5653
F-statistic: 32.21 on 4 and 92 DF, p-value: < 2.2e-16
```

```
> par(mfrow=c(2,2))
> plot(mod1)
> sigma(mod1)/mean(logpsa)
[1] 0.306875
```

p-value of the F-statistic is < 2.2e-16, which is highly significant. This means that, at least, one of the predictor variables is significantly related to the outcome variable. P-values<0.05 of t-statistics of all variables which means all reject null hypothesis and variables being significant and associated with response psa . Adjusted R squared =0.5653 , meaning that 56% of the variance in the measure of psa can be predicted by this regression model. Here RSE IS 0.7606 corresponding to 30% error rate.

#Residual diagnostics of final model and checking for validity of assumptions.



Residuals vs Fitted. This plot is used to check the linear relationship assumptions. A horizontal line, without distinct patterns is an indication for a linear relationship. There is no pattern in the residual plot. This suggests that we can assume linear relationship between the predictors and the psa levels.

Normal Q-Q. . to examine whether the residuals are normally distributed. It's good if residuals points follow the straight dashed line. Almost all data points lie on the Q-Q plot.

Scale-Location (or Spread-Location). Used to check the homogeneity of variance of the residuals (homoscedasticity). Horizontal line with equally spread points is a good indication of homoscedasticity. In our case, this looked slightly homoscedasticity.

Residuals vs Leverage. Used to identify influential cases, that is extreme values (#94, #91) that might influence the regression results when included or excluded from the analysis.

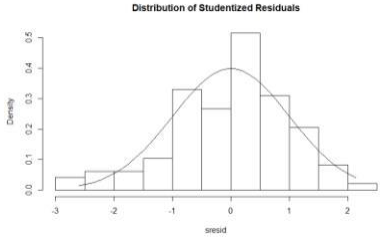
```
> #1) checking for normality

> shapiro.test(residuals(mod1))

        Shapiro-Wilk normality test

data: residuals(mod1)
```

W = 0.9919, p-value = 0.8281

p-values>0.05 accepting null hypothesis and hence confirming residuals are normally distributed.

```
> # distribution of studentized residuals
> #checking for normality
> shapiro.test(residuals(mod1))

        Shapiro-Wilk normality test

data: residuals(mod1)
W = 0.97912, p-value = 0.1251
```

```
> # distribution of studentized residuals
> sresid <- studres(mod1)
> hist(sresid, freq=FALSE,
+      main="Distribution of Studentized Residuals") >
xfit<-seq(min(sresid),max(sresid),length=40)
> yfit<-dnorm(xfit)
> lines(xfit, yfit)
```



As p-values>0.05 accepting null hypothesis and hence errors are normally distributed.

```
> #2) non-constant error variance test
> #Breush Pagan Test
> # non-constant error variance test
> #Breush Pagan Test
> lmtest::bptest(mod1)

        studentized Breusch-Pagan test

data: mod1
BP = 3.1199, df = 4, p-value = 0.538
```

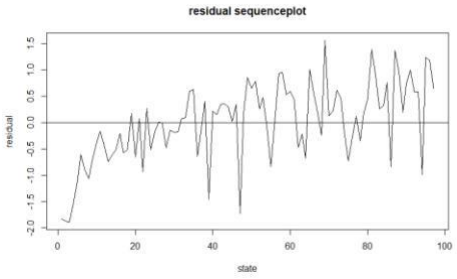
Both these tests have a p-value>0.05, therefore we accept the null hypothesis that the variance of the residuals is constant and infer that homoscedasticity is indeed present.

```
> #3) testing the independence assumptions
> #testing the independence assumptions
> library(car)
> dwt(mod1)
 lag Autocorrelation D-W Statistic p-value
   1      0.4333069     1.063381       0
 Alternative hypothesis: rho != 0
```

The Durbin Watson examines whether the errors are autocorrelated with themselves or not. The null states that they are not autocorrelated (what we want). Here p =0<0.05 rejecting null hypothesis.Hence we can't say the errors are not autocorrelated. It violated the independence assumption.

```
> #Testing independence
```

```
> par(mfrow=c(1,1))
> plot(residuals(mod1), xlab="state",ylab="residual",main="residual sequenceplot",type="l")
> abline(h=0)
```



f) Write the final model in equation form, being careful to handle qualitative predictors (if any) properly

Logpsa = -0.65 + 0.0649CancerVolume + 0.09136Benpros + 0.68421Vesinv1 + 0.3337Gleason

Here, vesinv is being factored.

Vesinv1 = 0 when vesinv =0

Vesinv1 = 1 when vesinv = 1

(g) Use the final model to predict the PSA level for a patient whose quantitative predictors are at the sample means of the variables and qualitative predictors (if any) are at the most frequent category.

Most frequent Vesinv :

```
> table(prost$vesinv)

 0  1
76 21
```

Vesinv 0 is more frequent.

```
> vesinv <- factor(vesinv)
> predict(mod1,data.frame(cancervol=mean(cancervol),benpros=mean(benpros),vesinv="0",gleason=mean(glea
son)))

2.330541

 psa<-exp(2.330541)
> psa
[1] 10.2835

> predict(mod1,data.frame(cancervol=mean(cancervol),benpros=mean(benpros),vesinv="0",gleason=mean(glea
son)), se.fit=T, interval="prediction")
$fit
      fit      lwr      upr
1 2.330541 0.8086761 3.852406
```

$se.fit
[1] 0.09265029

$df
[1] 92

$residual.scale
[1] 0.7606414

Predicted value = exp(2.330471) = 10.2827
This is the final predicted value.