Unit 2 Paper Technical Appendices

youngwoo Kwon

2021 3 4

#Summary In Appendix 1, the theoratical background for the data modificaiton was proven. Also, the hypothesis selection was done.

In Appendix 2, basic data analysis was done. The code calculated the proportion of missing values and displayed some scatter plots explaing the relationship between GMP and population size. The code also plotted other variables to find the connection with the previous relationship.

In Appendix 3, the code chose the linear model and plotted that model. It calculated the loss function outcome and residual variances.

In Appendix 4, the alternative models were written. Also, the loss function outcomes for those models were calculated.

In Appendix 5, original model and one comparable alternative model was compared with f-test.

#Appendix 1: Detail of Statistical models

1. If $Y \approx cN^b$ for some $c > 0, b > 1$, then $log(\frac{Y}{N}) \approx \beta_0 + \beta_1 log(N)$ for some $\beta_0 \in (-\infty, \infty), \beta_1 > 0$, and also $log(Y) \approx \beta_0 + (1 + \beta_1)log(N)$.

Let $Y \approx cN^b$. Then, $log(Y) \approx log(c) + blog(N)$. Therefore, for $\beta_0 = log(c), \beta_1 = b - 1$, $log(Y) \approx \beta_0 + (1 + \beta_1)log(N)$. Since $c > 0, b > 1$, we can say that $\beta_0 \in (-\infty, \infty)$ and $\beta_1 > 0$.

If we subtract log(N) in both sides, $log(Y) - log(N) \approx log(c) + (b - 1)log(N)$. So $log(\frac{Y}{N}) \approx \beta_0 + \beta_1 log(N)$ for $\beta_0 = log(c), \beta_1 = b - 1$.

2. Three hypothesis about how these other variables might influence per-capita GMP (pcgmp).

1) There is a linear relationship between Per-Capita GMP and population + fianace. (pcgmp ~ pop + finance)

2) There is a linear relationship between Per-Capita GMP and population + information, communication and technology. (pcgmp ~ pop + ict)

3) There is a linear relationship between GMP and population + information, communication and technologys. (gmp ~ pop + ict)

#Appendix 2: Exploratory analyses

1. Read and modify the data

```r
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
mydata = read.csv("http://dept.stat.lsa.umich.edu/~bbh/s485/data/gmp-2006.csv")
head(mydata)
```

```
##                                  MSA pcgmp    pop finance prof.tech     ict
## 1                          Akron, OH 32890 699300 0.12940   0.05440      NA
## 2                         Albany, GA 24270 163000 0.08217        NA 0.00708
## 3       Albany-Schenectady-Troy, NY 36840 850300 0.15780   0.09399 0.04511
## 4                    Albuquerque, NM 37660 816000 0.15990   0.09978 0.20500
## 5                     Alexandria, LA 25490 152200 0.09152   0.03790 0.01134
## 6 Allentown-Bethlehem-Easton, PA-NJ 30160 794400 0.13670        NA 0.03384
##   management
## 1   0.054310
## 2         NA
## 3         NA
## 4   0.006509
## 5   0.015210
## 6         NA
```

```r
newdata <- mydata
newdata$pcgmp <- as.double(newdata$pcgmp)
newdata$pop <- as.double(newdata$pop)
newdata$gmp <- newdata$pop * newdata$pcgmp
head(newdata)
```

```
##                                  MSA pcgmp    pop finance prof.tech     ict
## 1                          Akron, OH 32890 699300 0.12940   0.05440      NA
## 2                         Albany, GA 24270 163000 0.08217        NA 0.00708
## 3       Albany-Schenectady-Troy, NY 36840 850300 0.15780   0.09399 0.04511
## 4                    Albuquerque, NM 37660 816000 0.15990   0.09978 0.20500
## 5                     Alexandria, LA 25490 152200 0.09152   0.03790 0.01134
## 6 Allentown-Bethlehem-Easton, PA-NJ 30160 794400 0.13670        NA 0.03384
##   management         gmp
## 1   0.054310 22999977000
```

```
## 2          NA  3956010000
## 3          NA 31325052000
## 4    0.006509 30730560000
## 5    0.015210  3879578000
## 6          NA 23959104000
```

2. Missing Values

```r
nrow(newdata)
```

```
## [1] 244
```

```r
Finance_prop = nrow(newdata[4] %>% na.omit())/nrow(newdata)
Prof.tech_prop = nrow(newdata[5] %>% na.omit())/nrow(newdata)
ict_prop = nrow(newdata[6] %>% na.omit())/nrow(newdata)
management_prop = nrow(newdata[7] %>% na.omit())/nrow(newdata)
Finance_prof.tech_prop = nrow(newdata[4:5] %>% na.omit())/nrow(newdata)

Finance_prop
```

```
## [1] 0.9631148
```

```r
Prof.tech_prop
```

```
## [1] 0.6721311
```

```r
ict_prop
```

```
## [1] 0.8319672
```

```r
management_prop
```

```
## [1] 0.5532787
```

```r
Finance_prof.tech_prop
```

```
## [1] 0.6557377
```

```r
nrow(newdata %>% na.omit())/nrow(newdata)
```

```
## [1] 0.3729508
```

96.31148% of data have no missing value in finance section.

67.21311% of data have no missing value in professional and technical services section.

83.19672% of data have no missing value in information, communication and technology section.

55.32787% of data have no missing value in and enterprises section.

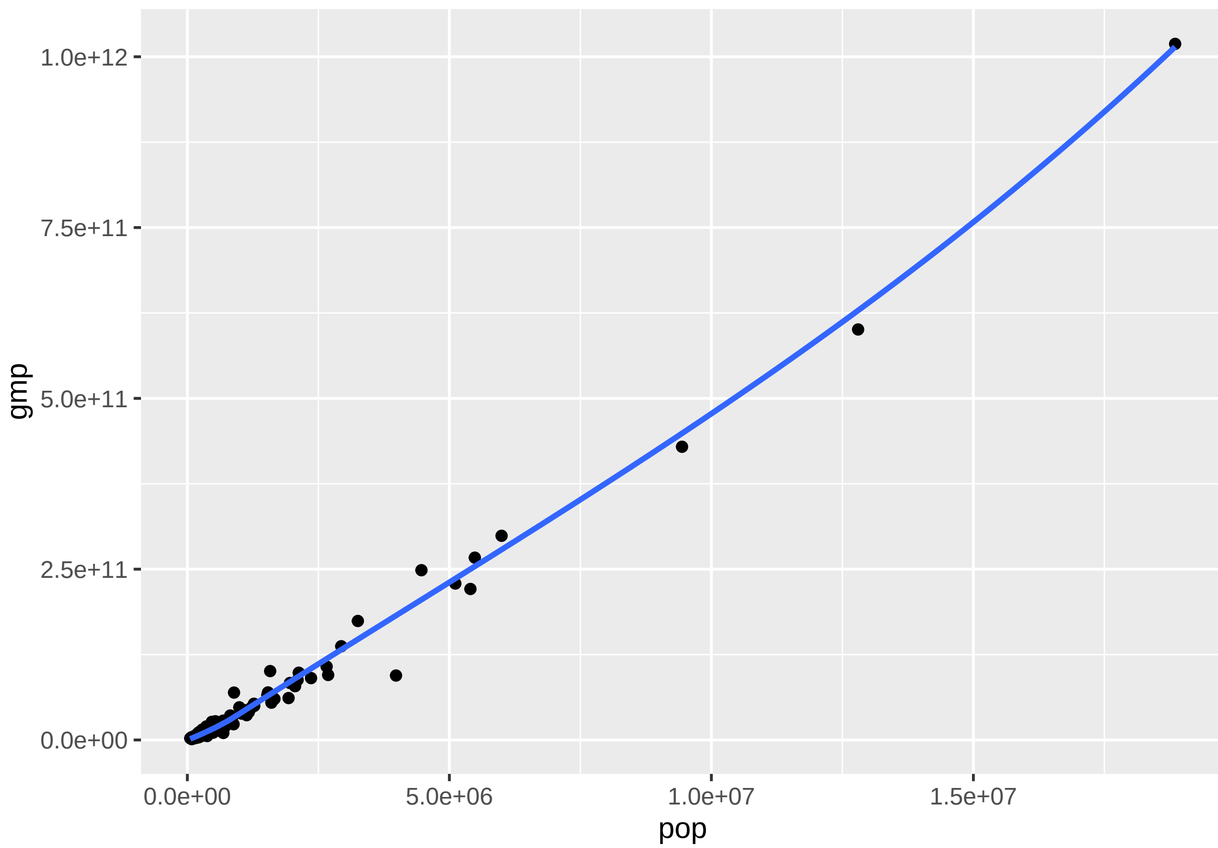
65.57377% of data have no missing value in finance and professional and technical services section.

37.29508% of data have no missing value.

3. Scatter plot

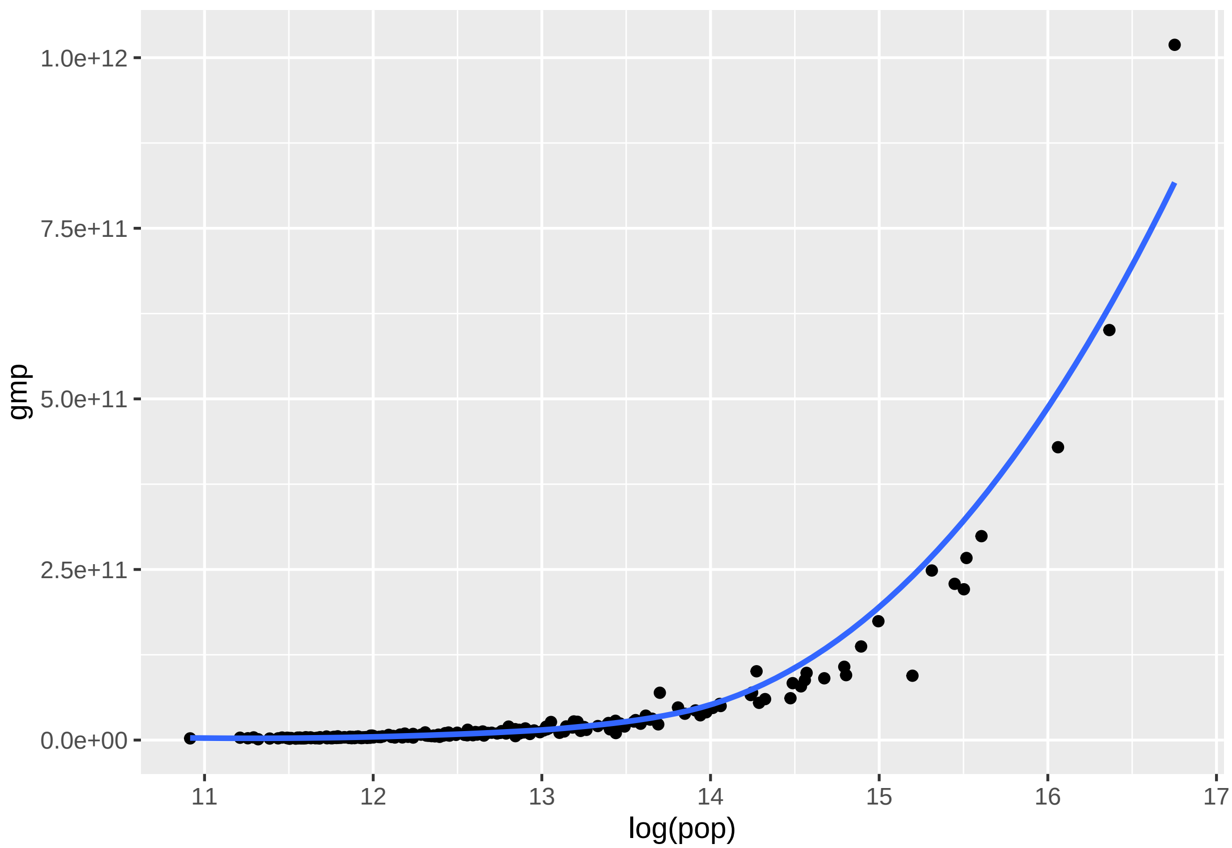
```r
gmp_pop = ggplot(newdata, aes(y=gmp, x=pop)) +
  geom_point() +
  geom_smooth(se=FALSE)
gmp_pop
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



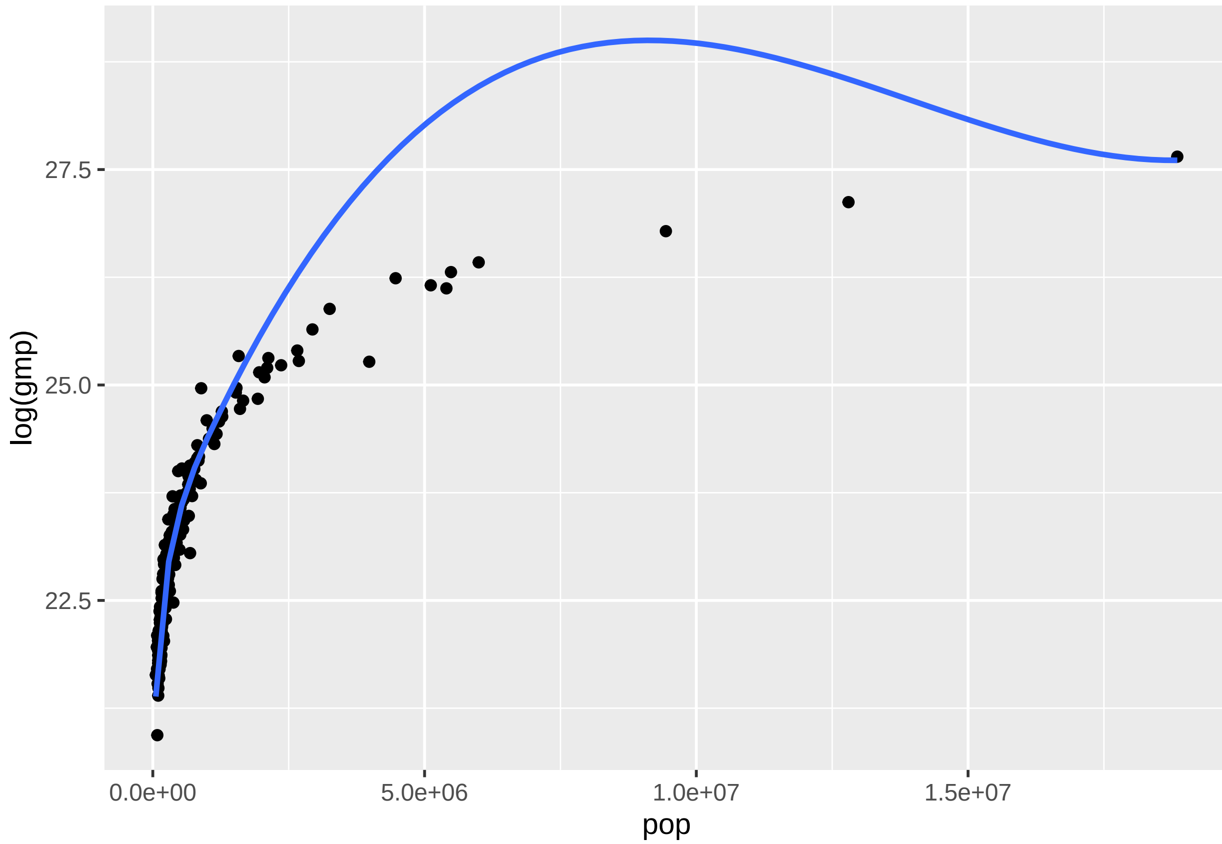

```r
loggmp_pop = ggplot(newdata, aes(y=gmp, x=log(pop))) +
  geom_point() +
  geom_smooth(se=FALSE)
loggmp_pop
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



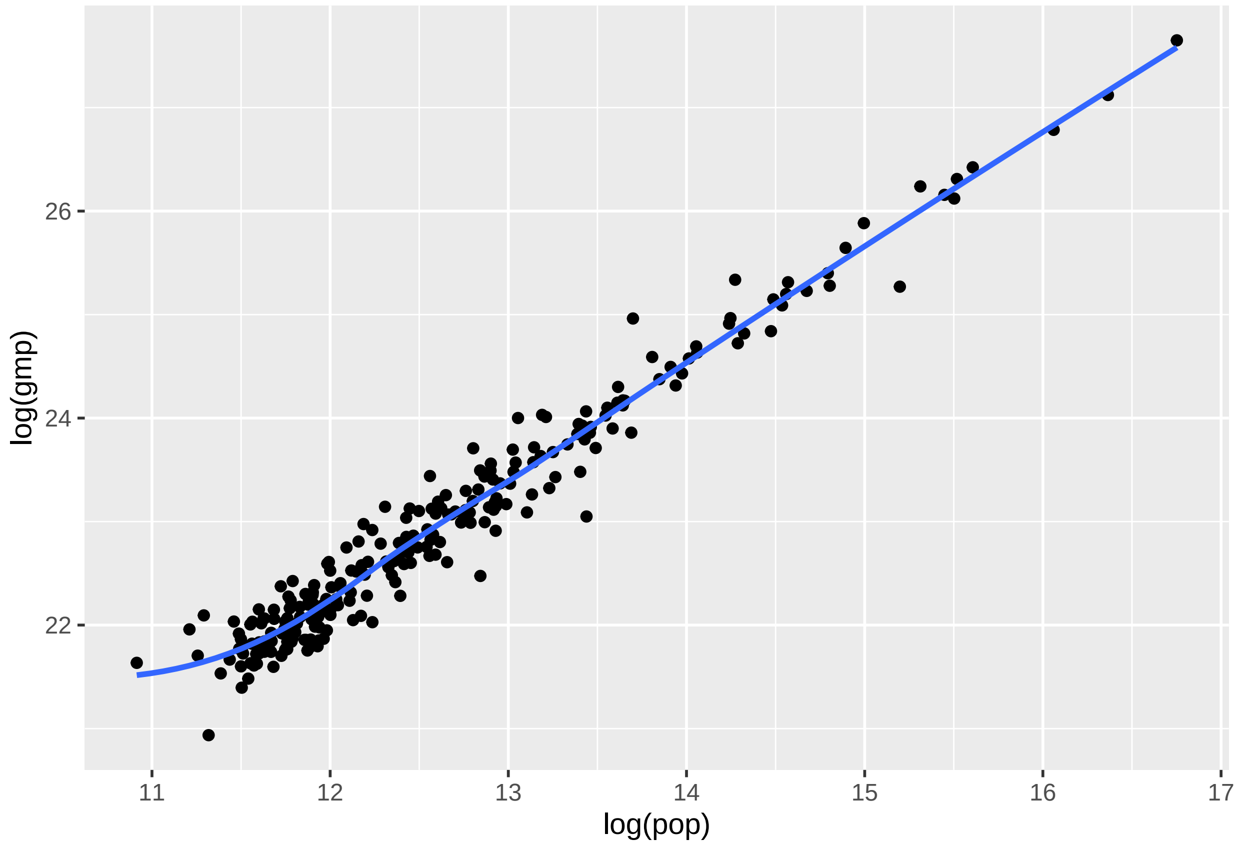

```
gmp_logpop = ggplot(newdata, aes(y=log(gmp), x=pop)) +
  geom_point() +
  geom_smooth(se=FALSE)
gmp_logpop
```

```
## ‘geom_smooth()‘ using method = ’loess’ and formula ’y ~ x’
```




```r
loggmp_logpop = ggplot(newdata, aes(y=log(gmp), x=log(pop))) +
  geom_point() +
  geom_smooth(se=FALSE)
loggmp_logpop
```

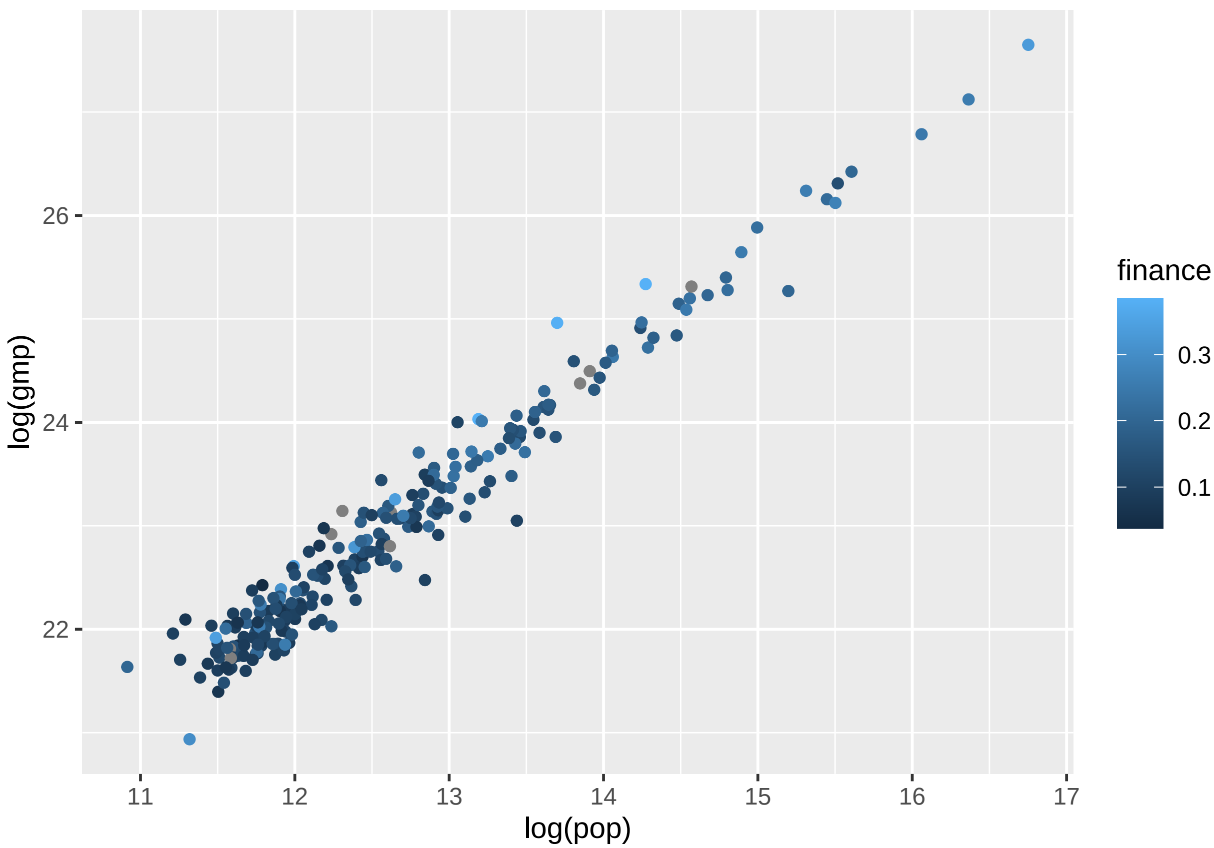
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



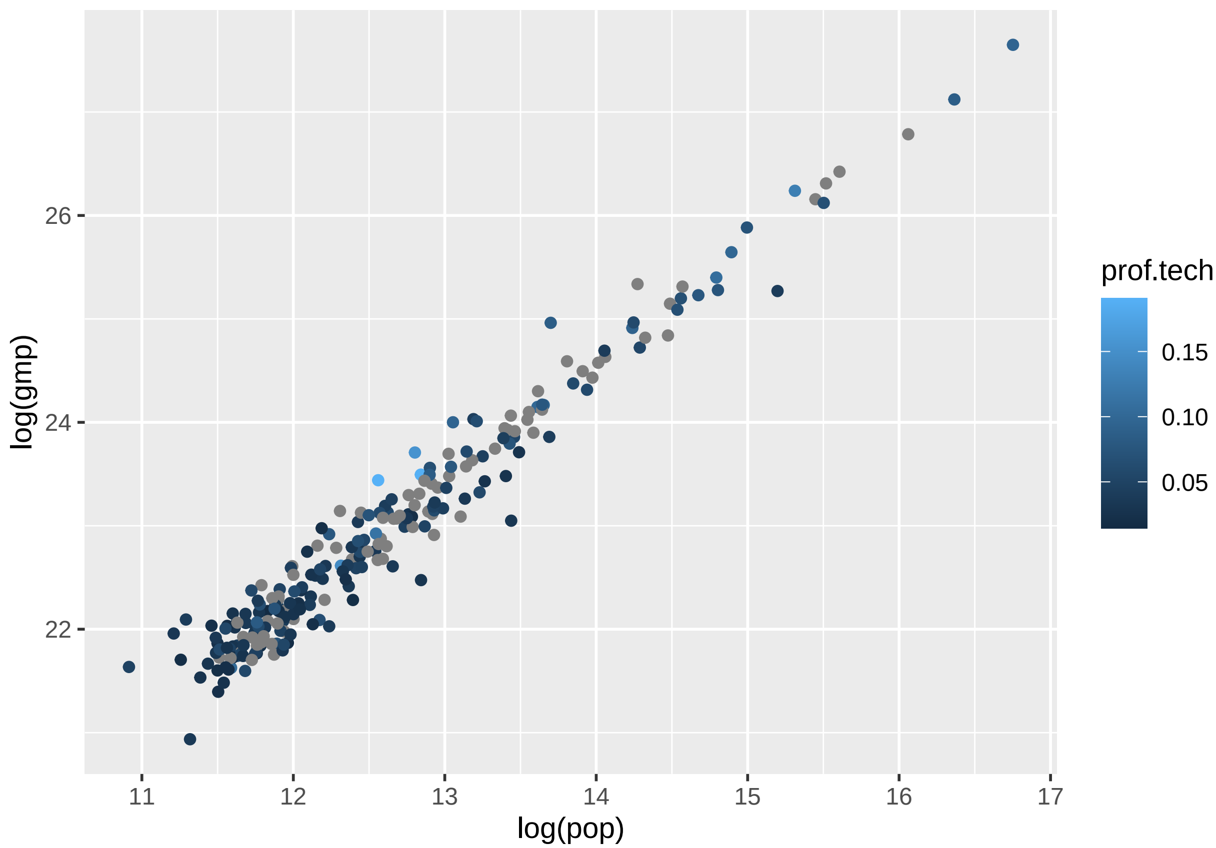
The results say that the log(y)~log(x) is better scale for capturing patterns.

4. Other variances and gmp~pop

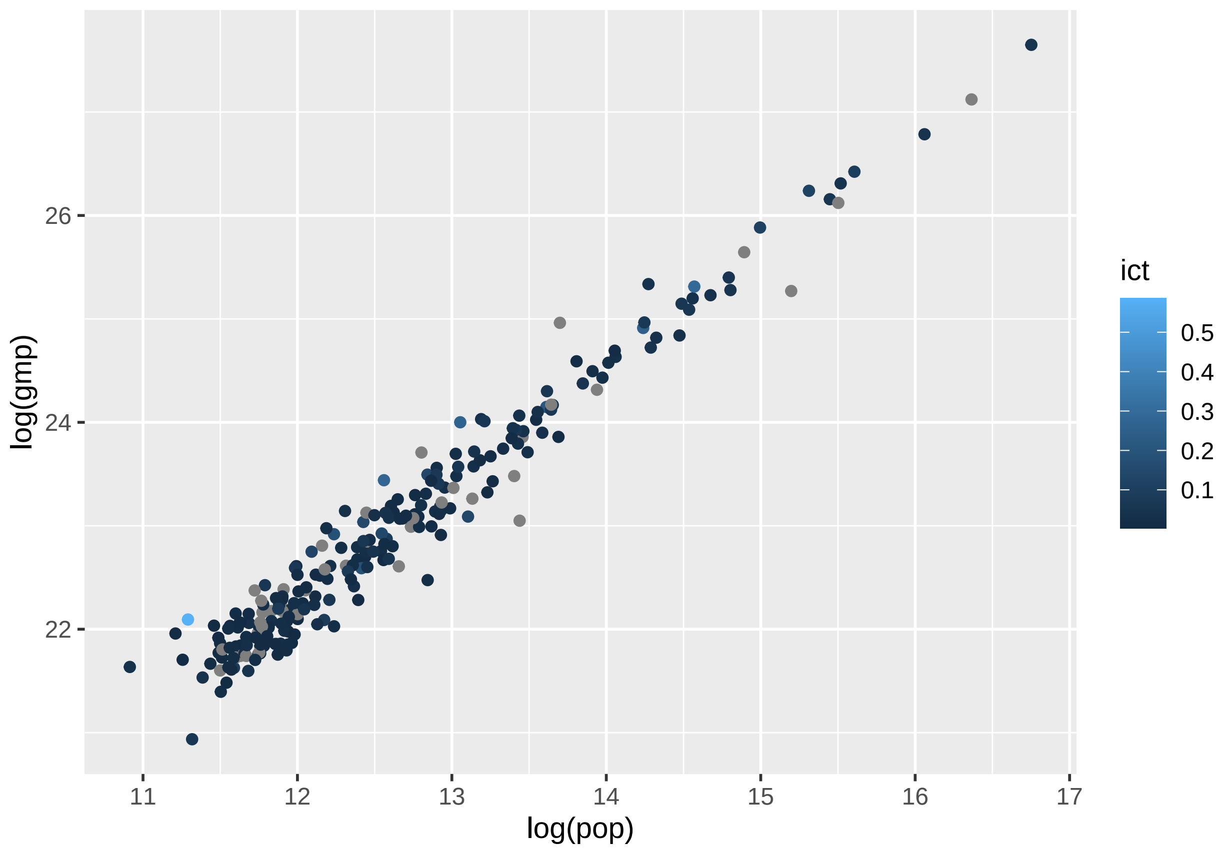
```
loggmp_logpop_finance = ggplot(newdata, aes(y=log(gmp), x=log(pop))) +
  geom_point(aes(colour = finance))
loggmp_logpop_finance
```




```r
loggmp_logpop_prof.tech = ggplot(newdata, aes(y=log(gmp), x=log(pop))) +
  geom_point(aes(colour = prof.tech))
loggmp_logpop_prof.tech
```




```r
loggmp_logpop_ict = ggplot(newdata, aes(y=log(gmp), x=log(pop))) +
  geom_point(aes(colour = ict))
loggmp_logpop_ict
```



I didn't remove the NA values because ggplot would automatically neglect and do not colour the data that have NA value

#Appendix 3: Fitting the power law model

1.Basic linear model

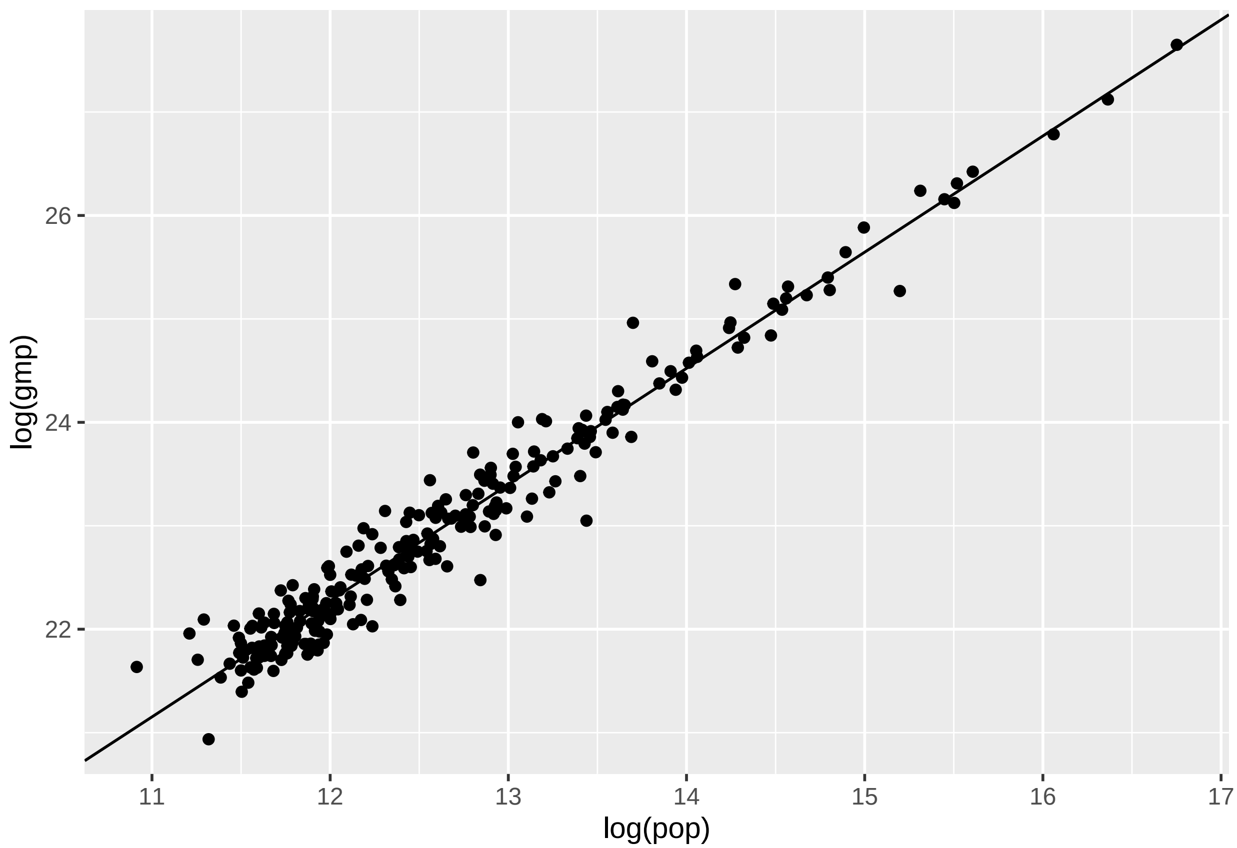
```
lm_loggmp_logpop = lm(log(gmp)~log(pop), data = newdata)
summary(lm_loggmp_logpop)
```

```
##
## Call:
## lm(formula = log(gmp) ~ log(pop), data = newdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.84226 -0.13993  0.00157  0.12942  0.77779
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.79623    0.18350   47.94   <2e-16 ***
## log(pop)     1.12326    0.01449   77.54   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.238 on 242 degrees of freedom
## Multiple R-squared:  0.9613, Adjusted R-squared:  0.9611
## F-statistic:  6012 on 1 and 242 DF,  p-value: < 2.2e-16
```

As we saw in the #Appendix 1, the log(c) = log(8.79623) equals to the β_0, and b-1 = 1.12326 - 1 = 0.012326 equals to the β_1. Since the Adjusted R-squared value is over 0.96 and t value for each estimate is large, we can say that this model supports the supra-linear power-law scaling hypothesis.

2. Plot the data, errors and residuals

```
ggplot(newdata, aes(y=log(gmp), x=log(pop))) +
  geom_point() +
  geom_abline(intercept = lm_loggmp_logpop$coefficients[1], slope = lm_loggmp_logpop$coefficients[2])
```




```
var(lm_loggmp_logpop$residuals)
```

```
## [1] 0.05642693
```

```
0.238^2 #From Residual standard error at linear model summary
```

```
## [1] 0.056644
```

So the variance of ghe residuals are almost equal to the variance of the regression. Sinde we got high t-value and small p-value for each coefficients and high adjusted R-sqaure value, we can trust the estimated coefficients.

3. Loss function, In-sample loss, estimated values of parameters

```
loss_log <-function(z, model){
  result = (log(z[1]) - predict(model, z[-1]))^2
  return(colMeans(result))
}

loss_log(newdata[c(8,3)], lm_loggmp_logpop)
```

```
##        gmp
## 0.05619567
```

(Used log_e instead of log_10. Essentially, $log_e(x) = rlog_{10}(x)$ where $r = log_e 10$, so nothing important changed.)

So the in-sample loss is 0.05619567. Since the in-sample loss is quite low, the expected values of the parameters make sense.

#Appendix 4: Fitting and assessment of alternate models

1, 2. Three alternate regression models & fit models

1) There is a linear relationship between Per-Capita GMP and population + fianace. (pcgmp ~ pop + finance)

2) There is a linear relationship between Per-Capita GMP and population + information, communication and technology. (pcgmp ~ pop + ict)

3) There is a linear relationship between Per-Capita GMP and population + professional and technical services. (pcgmp ~ pop + prof.tech)

```r
alt_model1 = lm(pcgmp~pop + finance, data = newdata)
alt_model2 = lm(pcgmp~pop + ict, data = newdata)
alt_model3 = lm(gmp~pop + ict, data = newdata)
```

```r
summary(alt_model1)
```

```
##
## Call:
## lm(formula = pcgmp ~ pop + finance, data = newdata)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -24223  -4509   -989   3878  33425
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.365e+04  1.306e+03  18.109  < 2e-16 ***
## pop         1.249e-03  3.080e-04   4.056 6.82e-05 ***
## finance     5.189e+04  8.425e+03   6.159 3.19e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7821 on 232 degrees of freedom
##   (9 observations deleted due to missingness)
## Multiple R-squared:  0.2678, Adjusted R-squared:  0.2615
## F-statistic: 42.43 on 2 and 232 DF,  p-value: < 2.2e-16
```

```r
summary(alt_model2)
```

```
##
## Call:
## lm(formula = pcgmp ~ pop + ict, data = newdata)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -17450.6  -4995.6   -801.7   4334.7  29372.5
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.915e+04  6.379e+02  45.703  < 2e-16 ***
```

```
## pop         1.990e-03  3.135e-04   6.350 1.42e-09 ***
## ict         5.236e+04  8.547e+03   6.126 4.70e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7378 on 200 degrees of freedom
##   (41 observations deleted due to missingness)
## Multiple R-squared:  0.3029, Adjusted R-squared:  0.296
## F-statistic: 43.46 on 2 and 200 DF,  p-value: < 2.2e-16
```

```
summary(alt_model3)
```

```
##
## Call:
## lm(formula = gmp ~ pop + ict, data = newdata)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -5.005e+10 -1.867e+09  2.108e+09  3.564e+09  5.547e+10
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.681e+09  7.786e+08  -8.581 2.62e-15 ***
## pop          5.145e+04  3.826e+02 134.478  < 2e-16 ***
## ict          4.220e+09  1.043e+10   0.405    0.686
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.006e+09 on 200 degrees of freedom
##   (41 observations deleted due to missingness)
## Multiple R-squared:  0.9892, Adjusted R-squared:  0.9891
## F-statistic:  9148 on 2 and 200 DF,  p-value: < 2.2e-16
```

All three models have very low adjusted r-sqaured value.

3. Evaluate the model based on the square-error loss function

```
loss <-function(z, model){
  result = (z[1] - predict(model, z[-1]))^2
  return(colMeans(result))
}

loss(newdata[c(2,3,4)] %>% na.omit(), alt_model1)
```

```
##    pcgmp
## 60380645
```

```
loss(newdata[c(2,3,6)] %>% na.omit(), alt_model2)
```

```
##    pcgmp
## 53635797
```

```r
loss(newdata[c(8,3,6)] %>% na.omit(), alt_model3)
```

```
##          gmp
## 7.990672e+19
```

```r
log(loss(newdata[c(2,3,4)] %>% na.omit(), alt_model1))
```

```
##    pcgmp
## 17.91618
```

```r
log(loss(newdata[c(2,3,6)] %>% na.omit(), alt_model2))
```

```
##    pcgmp
## 17.79773
```

```r
log(loss(newdata[c(8,3,6)] %>% na.omit(), alt_model3))
```

```
##      gmp
## 45.82739
```

All three models have very large loss function output. But we have to consider that 1) pcgmp and gmp has different scale, 2) the third alternative model considers gmp in normal scale, not a log scale.

#Appendix 5: Additional cacluation for version 2

To check the alternative model 3 more deeply, we need to make another loss function that is meaningful for the comparison. So, we made log_logloss function that compares the log of the residuals and log of predictions.

One problem is that we could not calculate the prediction if our prediction is less than zero. Although they are useful, I had to removed those data during the calculation.

```
log_logloss <- function(z,model){
  x = log(abs(predict(model, z[-1])))
  result = (log(z[1]) - x)^2 %>% na.omit()
  return(colMeans(result))
}

log_logloss(newdata[c(8,3,6)] %>% na.omit(), alt_model3)
```

```
##      gmp
## 1.388848
```

One another way to check the model is do the F-test.

```
renewed_data <- newdata
renewed_data$scaled_pop <- (newdata$pop)^lm_loggmp_logpop$coefficients[2]
renewed_model = lm(gmp~scaled_pop + 0, data=renewed_data)
summary(renewed_model)
```

```
##
## Call:
## lm(formula = gmp ~ scaled_pop + 0, data = renewed_data)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -7.861e+10 -8.720e+08 -4.594e+07  1.137e+09  5.177e+10
##
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## scaled_pop  6670.25      42.49     157   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.19e+09 on 243 degrees of freedom
## Multiple R-squared:  0.9902, Adjusted R-squared:  0.9902
## F-statistic: 2.465e+04 on 1 and 243 DF,  p-value: < 2.2e-16
```

```
compare_model1 = renewed_model
compare_model2 = lm(gmp~pop + ict, data = renewed_data)
summary(compare_model1)
```

```
##
## Call:
## lm(formula = gmp ~ scaled_pop + 0, data = renewed_data)
##
```

```
## Residuals:
##        Min         1Q     Median         3Q        Max
## -7.861e+10 -8.720e+08 -4.594e+07  1.137e+09  5.177e+10
##
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## scaled_pop  6670.25      42.49     157   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.19e+09 on 243 degrees of freedom
## Multiple R-squared:  0.9902, Adjusted R-squared:  0.9902
## F-statistic: 2.465e+04 on 1 and 243 DF,  p-value: < 2.2e-16
```

```r
summary(compare_model2)
```

```
##
## Call:
## lm(formula = gmp ~ pop + ict, data = renewed_data)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -5.005e+10 -1.867e+09  2.108e+09  3.564e+09  5.547e+10
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6.681e+09  7.786e+08  -8.581 2.62e-15 ***
## pop          5.145e+04  3.826e+02 134.478  < 2e-16 ***
## ict          4.220e+09  1.043e+10   0.405    0.686
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.006e+09 on 200 degrees of freedom
##   (41 observations deleted due to missingness)
## Multiple R-squared:  0.9892, Adjusted R-squared:  0.9891
## F-statistic:  9148 on 2 and 200 DF,  p-value: < 2.2e-16
```

We can check the F-statistic of two models. So we can check the p-value.

```r
pf(24650, 1, 243, lower.tail = FALSE)
```

```
## [1] 2.745241e-246
```

```r
pf(9148, 2, 200, lower.tail = FALSE)
```

```
## [1] 2.484379e-197
```

Our first model has a lower p-value from the F-statistic. So we can say that our first model is better, but those differences are not significiant since both p-values are so low.