

# Laporan KP

Erlin Shofiana

2024-12-25

```
library(readr)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
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
```

```
library(tidyr)
library(plm)
```

```
##
## Attaching package: 'plm'
```

```
## The following objects are masked from 'package:dplyr':
##
##   between, lag, lead
```

```
library(car)
```

```
## Warning: package 'car' was built under R version 4.3.3
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##   recode
```

```
library(lmtest)
```

```
## Warning: package 'lmtest' was built under R version 4.3.3
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##   as.Date, as.Date.numeric
```

```
#Load Data
library(readxl)
data <- read_excel("C:/Users/Lenovo/Downloads/downloaded_filnew3.xlsx")
prov = read_excel("C:/Users/Lenovo/Downloads/downloaded_filnew3.xlsx",sheet=4)
head(data,10)
```

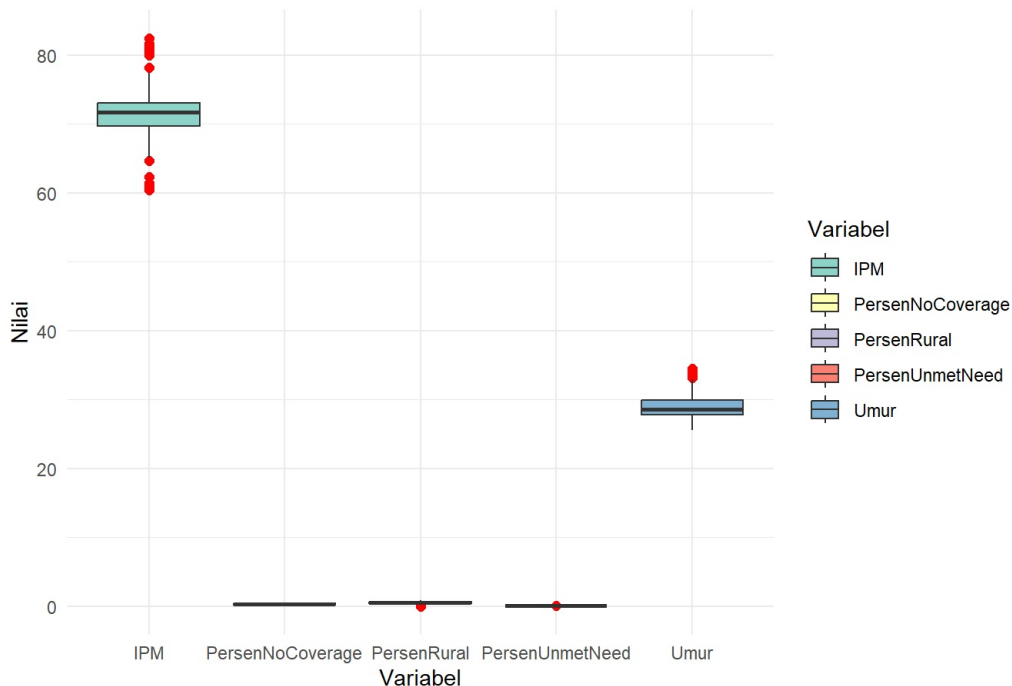
```
## # A tibble: 10 × 14
##   Provinsi Tahun PersenUnmetNeed PersenRural PersenTidakBekerja PersenLakiLaki
##   <chr>      <dbl>          <dbl>          <dbl>          <dbl>          <dbl>
## 1 Aceh      2019            0.0399          0.674            0.600            0.600
## 2 Bali      2019            0.0274          0.309            0.469            0.469
## 3 Banten    2019            0.0690          0.296            0.590            0.590
## 4 Bengkulu  2019            0.0500          0.665            0.523            0.523
## 5 D I Yogy... 2019            0.0425          0.276            0.469            0.469
## 6 Dki Jaka... 2019            0.0269          0          0.554            0.554
## 7 Gorontalo  2019            0.0752          0.582            0.558            0.558
## 8 Jambi      2019            0.0423          0.668            0.559            0.559
## 9 Jawa Bar... 2019            0.0508          0.222            0.582            0.582
## 10 Jawa Ten... 2019            0.0484          0.474            0.515            0.515
## # i 8 more variables: PersenNoCoverage <dbl>, Umur <dbl>,
## #   nonesensial_kapita <dbl>, gk <dbl>, PersenKelas3_15 <dbl>, IPM <dbl>,
## #   anggota <dbl>, under50 <dbl>
```

```
library(tidyr)
library(ggplot2)

# Asumsikan `data` adalah dataset Anda
data_long <- pivot_longer(data,
  cols = c(PersenUnmetNeed, Umur, PersenRural,
    PersenNoCoverage, IPM), # Pilih semua kolom yang akan digunakan
  names_to = "Variabel",
  values_to = "Nilai")

# Membuat boxplot
ggplot(data_long, aes(x = Variabel, y = Nilai, fill = Variabel)) +
  geom_boxplot(outlier.color = "red", outlier.shape = 16, outlier.size = 2) + # Mengatur outlier
  scale_fill_brewer(palette = "Set3") + # Palet warna
  labs(title = "Boxplot untuk Beberapa Variabel", x = "Variabel", y = "Nilai") +
  theme_minimal() # Tema minimalis
```

Boxplot untuk Beberapa Variabel



```
theme(legend.position = "none") # Hilangkan legend (opsional)
```

```
## List of 1
## $ legend.position: chr "none"
## - attr(*, "class")= chr [1:2] "theme" "gg"
## - attr(*, "complete")= logi FALSE
## - attr(*, "validate")= logi TRUE
```

```

data$logumur = log(data$Umur)
data$lognonesensial_kapita = log(data$nonessential_kapita)
data$loggk = log(data$gk)
data$logIPM = log(data$IPM)
#Multikolinearitas
m0 <- lm(PersenUnmetNeed ~logumur+PersenRural+
        PersenNoCoverage+lognonesensial_kapita+logIPM, data=data)
vif(m0)

```

```

##              logumur          PersenRural      PersenNoCoverage
##          1.589540          3.119121          1.016607
## lognonesensial_kapita          logIPM
##          3.006139          3.307490

```

```

gf1=plm(PersenUnmetNeed ~ logumur+PersenRural+
        PersenNoCoverage+lognonesensial_kapita+logIPM, data=data,model="within",index=c("Provinsi","Tahun"))
gc1=plm(PersenUnmetNeed ~ logumur+PersenRural+
        PersenNoCoverage+lognonesensial_kapita+logIPM, data=data,model="pooling",index=c("Provinsi","Tahun"))
gr1=plm(PersenUnmetNeed ~ logumur+PersenRural+
        PersenNoCoverage+lognonesensial_kapita+logIPM, data=data,model="random",index=c("Provinsi","Tahun"))

#Uji Chow
pooltest(gc1,gf1) #H0 = common

```

```

##
## F statistic
##
## data: PersenUnmetNeed ~ logumur + PersenRural + PersenNoCoverage + ...
## F = 7.656, df1 = 33, df2 = 131, p-value < 2.2e-16
## alternative hypothesis: unstability

```

```

#Uji Hausman
phtest(gf1,gr1) #H0 = random

```

```

##
## Hausman Test
##
## data: PersenUnmetNeed ~ logumur + PersenRural + PersenNoCoverage + ...
## chisq = 14.693, df = 5, p-value = 0.01176
## alternative hypothesis: one model is inconsistent

```

```

#Uji Breusch Pagan
plmtest(gf1,effect="twoways",type="bp") #H0 ditolak berarti ada efek dua arah

```

```

##
## Lagrange Multiplier Test - two-ways effects (Breusch-Pagan)
##
## data: PersenUnmetNeed ~ logumur + PersenRural + PersenNoCoverage + ...
## chisq = 116.39, df = 2, p-value < 2.2e-16
## alternative hypothesis: significant effects

```

```

plmtest(gf1,effect="individual",type="bp") #H0 ditolak berarti ada efek individu

```

```

##
## Lagrange Multiplier Test - (Breusch-Pagan)
##
## data: PersenUnmetNeed ~ logumur + PersenRural + PersenNoCoverage + ...
## chisq = 87.685, df = 1, p-value < 2.2e-16
## alternative hypothesis: significant effects

```

```

plmtest(gf1,effect="time",type="bp") #H0 ditolak berarti ada efek waktu

```

```

##
## Lagrange Multiplier Test - time effects (Breusch-Pagan)
##
## data: PersenUnmetNeed ~ logumur + PersenRural + PersenNoCoverage + ...
## chisq = 28.709, df = 1, p-value = 8.411e-08
## alternative hypothesis: significant effects

```

```
#Pembentukan model
gla=plm(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita+logIPM,data=data,model="with
in",index=c("Provinsi","Tahun"),effect='twoways')
```

```
#Homoskedastisitas Residual
bptest(gla,studentize = F)
```

```
##
## Breusch-Pagan test
##
## data: gla
## BP = 11.134, df = 5, p-value = 0.04878
```

```
#Diagnostic
#Normalitas residual
ks.test(gla$residuals, "pnorm",
        mean=mean(gla$residuals),
        sd=sd(gla$residuals))
```

```
##
## Asymptotic one-sample Kolmogorov-Smirnov test
##
## data: gla$residuals
## D = 0.043093, p-value = 0.9104
## alternative hypothesis: two-sided
```

```
#Autokorelasi Residual
pbgtest(gla) #H0 : tidak ada korelasi serial pada komponen gla
```

```
##
## Breusch-Godfrey/Wooldridge test for serial correlation in panel models
##
## data: PersenUnmetNeed ~ logumur + PersenRural + PersenNoCoverage + ...
## chisq = 26.908, df = 5, p-value = 5.944e-05
## alternative hypothesis: serial correlation in idiosyncratic errors
```

```
data$Provinsi = toupper(data$Provinsi)
#list(data$Provinsi)
# Menggunakan dplyr untuk memanipulasi data frame
library(dplyr)
data <- data %>%
  mutate(Provinsi = case_when(
    Provinsi == "D I YOGYAKARTA" ~ "DI YOGYAKARTA",
    Provinsi == "KEP. BANGKA BELITUNG" ~ "KEPULAUAN BANGKA BELITUNG",
    TRUE ~ Provinsi
  ))
library(jsonlite)
df_long <- fromJSON("https://raw.githubusercontent.com/yusufsyarifudin/wilayah-indonesia/master/data/list_of_area/
provinces.json")
df_long$Provinsi = df_long$name
data_gabungan <- merge(df_long, data, by = "Provinsi")
data_gab = data_gabungan[,-c(2:4)]
#Data Baru
library(GWmodel)
```

```
## Warning: package 'GWmodel' was built under R version 4.3.3
```

```
## Loading required package: robustbase
```

```
## Loading required package: sp
```

```
## Warning: package 'sp' was built under R version 4.3.3
```

```
## Loading required package: Rcpp
```

```
## Welcome to GWmodel version 2.3-2.
```

```
coordinates(data_gab)=c(2:3)
class(data_gab)
```

```
## [1] "SpatialPointsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
bwd.GWPR.bisquare.ad <- bw.gwr(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita+logIP
M
                                , data = data_gab, approach = "CV", kernel = "bisquare", adaptive=T)
```

```
## Adaptive bandwidth: 112 CV score: 0.03166405
## Adaptive bandwidth: 77 CV score: 0.0289413
## Adaptive bandwidth: 54 CV score: 0.02779505
## Adaptive bandwidth: 41 CV score: 0.02724518
## Adaptive bandwidth: 32 CV score: 0.03023192
## Adaptive bandwidth: 46 CV score: 0.0277654
## Adaptive bandwidth: 37 CV score: 0.02780959
## Adaptive bandwidth: 42 CV score: 0.02724518
```

```
hasil.GWPR.bisquare.ad <- gwr.basic(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita+
logIPM
                                , data = data_gab, bw = bwd.GWPR.bisquare.ad, kernel = "bisquare", adaptive=T
)
```

```
#adaptive gaussian
bwd.GWPR.gaussian.ad <- bw.gwr(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita+logIP
M
                                , data = data_gab, approach = "CV", kernel = "gaussian", adaptive=T)
```

```
## Adaptive bandwidth: 112 CV score: 0.03879362
## Adaptive bandwidth: 77 CV score: 0.03545217
## Adaptive bandwidth: 54 CV score: 0.0329948
## Adaptive bandwidth: 41 CV score: 0.03201582
## Adaptive bandwidth: 32 CV score: 0.03107996
## Adaptive bandwidth: 27 CV score: 0.03051749
## Adaptive bandwidth: 23 CV score: 0.02929321
## Adaptive bandwidth: 21 CV score: 0.02929321
```

```
hasil.GWPR.gaussian.ad <- gwr.basic(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita+
logIPM
                                , data = data_gab, bw = bwd.GWPR.gaussian.ad, kernel = "gaussian", adaptive=T
)
```

```
#adaptive exponential
bwd.GWPR.exponential.ad <- bw.gwr(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita+lo
gIPM
                                , data = data_gab, approach = "CV", kernel = "exponential", adaptive=T)
```

```
## Adaptive bandwidth: 112 CV score: 0.03556463
## Adaptive bandwidth: 77 CV score: 0.03301374
## Adaptive bandwidth: 54 CV score: 0.03123953
## Adaptive bandwidth: 41 CV score: 0.03026574
## Adaptive bandwidth: 32 CV score: 0.02925743
## Adaptive bandwidth: 27 CV score: 0.02868444
## Adaptive bandwidth: 23 CV score: 0.0278451
## Adaptive bandwidth: 21 CV score: 0.0278451
```

```
hasil.GWPR.exponential.ad <- gwr.basic(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapi
ta+logIPM
                                , data = data_gab, bw = bwd.GWPR.exponential.ad, kernel = "exponential", a
daptive=T)
```

```
#fixed bisquare
bwd.GWPR.bisquare.fix <- bw.gwr(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita+logI
PM
                                , data = data_gab, approach = "CV", kernel = "bisquare", adaptive=F)
```

```
## Fixed bandwidth: 26.05943 CV score: 0.03492284
## Fixed bandwidth: 16.10883 CV score: 0.03042554
## Fixed bandwidth: 9.959026 CV score: 0.02849991
## Fixed bandwidth: 6.158237 CV score: 0.03480682
## Fixed bandwidth: 12.30804 CV score: 0.02974504
## Fixed bandwidth: 8.507254 CV score: 0.03030893
## Fixed bandwidth: 10.85627 CV score: 0.0289531
## Fixed bandwidth: 9.404498 CV score: 0.02839337
## Fixed bandwidth: 9.061781 CV score: 0.02882084
## Fixed bandwidth: 9.616309 CV score: 0.02837465
```

```
hasil.GWPR.bisquare.fix <- gwr.basic(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita
+logIPM
                                , data = data_gab, bw = bwd.GWPR.bisquare.fix, kernel = "bisquare", adaptive
=F)

#fixed gaussian
bwd.GWPR.gaussian.fix <- bw.gwr(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita+logI
PM
                                , data = data_gab, approach = "CV", kernel = "gaussian", adaptive=F)
```

```
## Fixed bandwidth: 26.05943 CV score: 0.04131368
## Fixed bandwidth: 16.10883 CV score: 0.03764266
## Fixed bandwidth: 9.959026 CV score: 0.03348812
## Fixed bandwidth: 6.158237 CV score: 0.02975898
## Fixed bandwidth: 3.809221 CV score: 0.02754389
## Fixed bandwidth: 2.357449 CV score: 0.02904728
## Fixed bandwidth: 4.706465 CV score: 0.02847133
## Fixed bandwidth: 3.254693 CV score: 0.02713428
## Fixed bandwidth: 2.911976 CV score: 0.0272163
```

```
hasil.GWPR.gaussian.fix <- gwr.basic(PersenUnmetNeed ~ logumur+PersenRural+PersenNoCoverage+lognonesensial_kapita
+logIPM
                                , data = data_gab, bw = bwd.GWPR.gaussian.fix, kernel = "gaussian", adaptive
=F)

#fixed exponential
bwd.GWPR.exponential.fix <- bw.gwr(PersenUnmetNeed ~logumur+PersenRural+
                                PersenNoCoverage+lognonesensial_kapita+logIPM, data = data_gab, approach = "
CV", kernel = "exponential", adaptive=F)
```

```
## Fixed bandwidth: 26.05943 CV score: 0.03774969
## Fixed bandwidth: 16.10883 CV score: 0.03490001
## Fixed bandwidth: 9.959026 CV score: 0.03188255
## Fixed bandwidth: 6.158237 CV score: 0.02890819
## Fixed bandwidth: 3.809221 CV score: 0.02647164
## Fixed bandwidth: 2.357449 CV score: 0.02604746
## Fixed bandwidth: 1.460204 CV score: 0.02955233
## Fixed bandwidth: 2.911976 CV score: 0.02588523
## Fixed bandwidth: 3.254693 CV score: 0.02603325
## Fixed bandwidth: 2.700165 CV score: 0.02587138
```

```

hasil.GWPR.exponential.fix <- gwr.basic(PersenUnmetNeed ~ logumur+PersenRural+
                                     PersenNoCoverage+lognonesensial_kapita+logIPM, data = data_gab, bw = bw
d.GWPR.exponential.fix, kernel = "exponential", adaptive=F)

AIC <- c(hasil.GWPR.bisquare.ad$GW.diagnostic$AIC,
        hasil.GWPR.gaussian.ad$GW.diagnostic$AIC,
        hasil.GWPR.exponential.ad$GW.diagnostic$AIC,
        hasil.GWPR.bisquare.fix$GW.diagnostic$AIC,
        hasil.GWPR.gaussian.fix$GW.diagnostic$AIC,
        hasil.GWPR.exponential.fix$GW.diagnostic$AIC)
R2 <- c(hasil.GWPR.bisquare.ad$GW.diagnostic$gw.R2,
        hasil.GWPR.gaussian.ad$GW.diagnostic$gw.R2,
        hasil.GWPR.exponential.ad$GW.diagnostic$gw.R2,
        hasil.GWPR.bisquare.fix$GW.diagnostic$gw.R2,
        hasil.GWPR.gaussian.fix$GW.diagnostic$gw.R2,
        hasil.GWPR.exponential.fix$GW.diagnostic$gw.R2)

R2adj <- c(hasil.GWPR.bisquare.ad$GW.diagnostic$gwR2.adj,
           hasil.GWPR.gaussian.ad$GW.diagnostic$gwR2.adj,
           hasil.GWPR.exponential.ad$GW.diagnostic$gwR2.adj,
           hasil.GWPR.bisquare.fix$GW.diagnostic$gwR2.adj,
           hasil.GWPR.gaussian.fix$GW.diagnostic$gwR2.adj,
           hasil.GWPR.exponential.fix$GW.diagnostic$gwR2.adj)
RSS <- c(hasil.GWPR.bisquare.ad$GW.diagnostic$RSS.gw,
        hasil.GWPR.gaussian.ad$GW.diagnostic$RSS.gw,
        hasil.GWPR.exponential.ad$GW.diagnostic$RSS.gw,
        hasil.GWPR.bisquare.fix$GW.diagnostic$RSS.gw,
        hasil.GWPR.gaussian.fix$GW.diagnostic$RSS.gw,
        hasil.GWPR.exponential.fix$GW.diagnostic$RSS.gw)

tabel <- cbind(RSS,R2,R2adj,AIC)
rownames(tabel) <- c("Adaptive Bisquare", "Adaptive Gaussian",
                    "Adaptive Exponential", "Fixed Bisquare",
                    "Fixed Gaussian", "Fixed Exponential")
data.frame(tabel)

```

##		RSS	R2	R2adj	AIC
##	Adaptive Bisquare	0.01582504	0.6847840	0.5490207	-1052.107
##	Adaptive Gaussian	0.02286500	0.5445562	0.4562384	-1011.227
##	Adaptive Exponential	0.02151480	0.5714506	0.4755208	-1020.494
##	Fixed Bisquare	0.01971820	0.6072367	0.4967317	-1027.127
##	Fixed Gaussian	0.01782286	0.6449898	0.5199686	-1039.680
##	Fixed Exponential	0.01665860	0.6681803	0.5344176	-1049.321

```

comparison <- data.frame(R2=c(hasil.GWPR.exponential.fix$GW.diagnostic$gw.R2,summary(gla)$r.squared[1]),
                        R2adj=c(hasil.GWPR.exponential.fix$GW.diagnostic$gwR2.adj,summary(gla)$r.squared[2]),
                        RSS=c(hasil.GWPR.exponential.fix$GW.diagnostic$RSS.gw,sum(summary(gla)$residuals^2)))

rownames(comparison) <- c("GWPR", "Regresi Panel")
comparison

```

##		R2	R2adj	RSS
##	GWPR	0.66818034	0.5344176	0.01665860
##	Regresi Panel	0.03662226	-0.2819751	0.01052251

```

parameter_GWPR <- data.frame(
  Provinsi = data_gab$Provinsi,
  hasil.GWPR.exponential.fix$SDF[,c(1,2,3,4,5,6)]
)

input = parameter_GWPR[,c(1:7)]
srow <- seq(from = 1, to = nrow(input), by = 5)
s_data <- input[srow, ]
s_data

```

##	Provinsi	Intercept	logumur	PersenRural
## 1	ACEH	-0.700614803	0.1189702718	0.0367906181
## 6	BALI	1.329691962	0.0196267272	0.0198922951
## 11	BANTEN	0.566363748	0.0446592930	-0.0081898615
## 16	BENGKULU	0.002808142	0.0762031672	0.0177630553
## 21	DI YOGYAKARTA	0.509103220	0.1603384571	0.0247960878
## 26	DKI JAKARTA	0.239806363	0.0926976681	0.0295301289
## 31	GORONTALO	0.234861176	0.0002104620	0.0611572863
## 36	JAMBI	0.013432557	0.0667272823	0.0243536784
## 41	JAWA BARAT	0.675098469	0.0538932229	-0.0057407261
## 46	JAWA TENGAH	1.089040904	0.0722657630	-0.0038918394
## 51	JAWA TIMUR	1.271232326	0.0253663743	0.0146924785
## 56	KALIMANTAN BARAT	0.149363978	0.0457070280	0.0358913490
## 61	KALIMANTAN SELATAN	0.755469709	-0.0092689336	0.0223992581
## 66	KALIMANTAN TENGAH	0.515472477	0.0024966387	0.0265043980
## 71	KALIMANTAN TIMUR	0.314001598	0.0366275682	0.0466544336
## 76	KALIMANTAN UTARA	0.243964851	0.0865784554	0.0399262939
## 81	KEPULAUAN BANGKA BELITUNG	0.316546724	0.0803300291	0.0040787874
## 86	KEPULAUAN RIAU	-0.018278888	0.0765246184	0.0242400146
## 91	LAMPUNG	0.250904625	0.0831290988	-0.0007298808
## 96	MALUKU	-0.674746375	-0.0144120187	0.0489375468
## 101	MALUKU UTARA	-0.178363590	0.0051464108	0.0233160378
## 106	NUSA TENGGARA BARAT	0.982437513	-0.0175257571	0.0281782876
## 111	NUSA TENGGARA TIMUR	0.294764109	-0.0461477859	0.0555836926
## 116	PAPUA	-0.531718848	-0.0060537528	0.0194803617
## 121	PAPUA BARAT	-0.502632146	0.0004845747	0.0236961742
## 126	RIAU	-0.123327959	0.0537442423	0.0279394716
## 131	SULAWESI BARAT	0.072241889	-0.0410418128	0.0601725072
## 136	SULAWESI SELATAN	0.082386658	-0.0617781586	0.0620593592
## 141	SULAWESI TENGAH	0.072768173	-0.0493090657	0.0602261101
## 146	SULAWESI TENGGARA	-0.192885410	-0.0829967114	0.0750852968
## 151	SULAWESI UTARA	0.030413998	-0.0320013699	0.0695569188
## 156	SUMATERA BARAT	-0.188868718	0.0532674270	0.0270412057
## 161	SUMATERA SELATAN	0.055049843	0.0821870048	0.0115686679
## 166	SUMATERA UTARA	-0.352295108	0.0581050863	0.0272188786
##	PersenNoCoverage	lognonesensial_kapita	logIPM	
## 1	0.0058794323	-0.0137676484	0.119222690	
## 6	0.0360772139	0.0303463296	-0.412584759	
## 11	0.0290695041	0.0071716680	-0.179660283	
## 16	0.0088860536	0.0185141269	-0.110053556	
## 21	-0.0459573793	0.0177539968	-0.288186080	
## 26	-0.0107816812	0.0129305832	-0.160403429	
## 31	-0.0556376309	0.0109432062	-0.079522389	
## 36	-0.0006565541	0.0236098188	-0.121141100	
## 41	0.0133676041	0.0011021660	-0.192405637	
## 46	0.0128705574	0.0005538692	-0.301852994	
## 51	0.0248185344	0.0181177989	-0.364422589	
## 56	-0.0161899890	0.0133377018	-0.103532632	
## 61	-0.0042820674	0.0057470916	-0.176612465	
## 66	-0.0159418997	0.0007252125	-0.114285604	
## 71	-0.0332522801	0.0049671985	-0.108292231	
## 76	-0.0468265264	0.0036944338	-0.125491423	
## 81	0.0068329858	0.0004344810	-0.128642371	
## 86	0.0035713404	0.0167396642	-0.100348361	
## 91	0.0305941178	0.0107236357	-0.148652249	
## 96	-0.0669167766	-0.0209908212	0.244590797	
## 101	-0.0826593297	-0.0256931128	0.130936212	
## 106	0.0514167636	0.0480778048	-0.357656001	
## 111	0.0589294388	0.0436380813	-0.163727250	
## 116	0.0000809393	-0.0110570673	0.171800581	
## 121	-0.0337550197	-0.0225535738	0.197810891	
## 126	0.0054506594	0.0144208434	-0.051431539	
## 131	0.0233627790	0.0043419133	0.006597737	
## 136	0.0315102280	0.0081025771	0.008359893	
## 141	0.0128634731	0.0014868065	0.022632491	
## 146	0.0193569484	0.0056175914	0.096540247	
## 151	-0.0839143904	0.0114931351	-0.008032130	
## 156	0.0136323637	0.0153174292	-0.039450160	
## 161	0.0122351265	0.0141131508	-0.112962696	
## 166	0.0093456477	-0.0093035651	0.072304639	

```

pval <- gwr.t.adjust(hasil.GWPR.exponential.fix)$results$p
pvalue= data.frame("Provinsi"=data_gab$Provinsi,ifelse(pval<0.05, "Signifikan", "Tidak"))
selected_rows <- seq(from = 1, to = nrow(pvalue), by = 5)
selected_data <- pvalue[selected_rows, ]
selected_data

```



##	Provinsi	Intercept_p	logumur_p	PersenRural_p
## 1	ACEH	Tidak	Tidak	Tidak
## 6	BALI	Signifikan	Tidak	Tidak
## 11	BANTEN	Tidak	Tidak	Tidak
## 16	BENGKULU	Tidak	Tidak	Tidak
## 21	DI YOGYAKARTA	Tidak	Signifikan	Tidak
## 26	DKI JAKARTA	Tidak	Signifikan	Tidak
## 31	GORONTALO	Tidak	Tidak	Tidak
## 36	JAMBI	Tidak	Tidak	Tidak
## 41	JAWA BARAT	Tidak	Tidak	Tidak
## 46	JAWA TENGAH	Signifikan	Signifikan	Tidak
## 51	JAWA TIMUR	Signifikan	Tidak	Tidak
## 56	KALIMANTAN BARAT	Tidak	Tidak	Signifikan
## 61	KALIMANTAN SELATAN	Signifikan	Tidak	Tidak
## 66	KALIMANTAN TENGAH	Tidak	Tidak	Tidak
## 71	KALIMANTAN TIMUR	Tidak	Tidak	Tidak
## 76	KALIMANTAN UTARA	Tidak	Tidak	Tidak
## 81	KEPULAUAN BANGKA BELITUNG	Tidak	Signifikan	Tidak
## 86	KEPULAUAN RIAU	Tidak	Signifikan	Tidak
## 91	LAMPUNG	Tidak	Signifikan	Tidak
## 96	MALUKU	Tidak	Tidak	Tidak
## 101	MALUKU UTARA	Tidak	Tidak	Tidak
## 106	NUSA TENGGARA BARAT	Signifikan	Tidak	Tidak
## 111	NUSA TENGGARA TIMUR	Tidak	Tidak	Tidak
## 116	PAPUA	Tidak	Tidak	Tidak
## 121	PAPUA BARAT	Tidak	Tidak	Tidak
## 126	RIAU	Tidak	Tidak	Tidak
## 131	SULAWESI BARAT	Tidak	Tidak	Signifikan
## 136	SULAWESI SELATAN	Tidak	Tidak	Signifikan
## 141	SULAWESI TENGAH	Tidak	Tidak	Signifikan
## 146	SULAWESI TENGGARA	Tidak	Tidak	Signifikan
## 151	SULAWESI UTARA	Tidak	Tidak	Signifikan
## 156	SUMATERA BARAT	Tidak	Tidak	Tidak
## 161	SUMATERA SELATAN	Tidak	Signifikan	Tidak
## 166	SUMATERA UTARA	Tidak	Tidak	Tidak
##	PersenNoCoverage_p	lognonesensial_kapita_p	logIPM_p	
## 1	Tidak	Tidak	Tidak	
## 6	Tidak	Tidak	Signifikan	
## 11	Tidak	Tidak	Signifikan	
## 16	Tidak	Tidak	Tidak	
## 21	Signifikan	Tidak	Signifikan	
## 26	Tidak	Tidak	Signifikan	
## 31	Signifikan	Tidak	Tidak	
## 36	Tidak	Tidak	Tidak	
## 41	Tidak	Tidak	Signifikan	
## 46	Tidak	Tidak	Signifikan	
## 51	Tidak	Tidak	Signifikan	
## 56	Tidak	Tidak	Tidak	
## 61	Tidak	Tidak	Signifikan	
## 66	Tidak	Tidak	Tidak	
## 71	Tidak	Tidak	Tidak	
## 76	Signifikan	Tidak	Tidak	
## 81	Tidak	Tidak	Tidak	
## 86	Tidak	Tidak	Tidak	
## 91	Tidak	Tidak	Tidak	
## 96	Tidak	Tidak	Signifikan	
## 101	Signifikan	Tidak	Tidak	
## 106	Signifikan	Signifikan	Signifikan	
## 111	Signifikan	Signifikan	Tidak	
## 116	Tidak	Tidak	Tidak	
## 121	Tidak	Tidak	Tidak	
## 126	Tidak	Tidak	Tidak	
## 131	Tidak	Tidak	Tidak	
## 136	Tidak	Tidak	Tidak	
## 141	Tidak	Tidak	Tidak	
## 146	Tidak	Tidak	Tidak	
## 151	Signifikan	Tidak	Tidak	
## 156	Tidak	Tidak	Tidak	
## 161	Tidak	Tidak	Tidak	
## 166	Tidak	Tidak	Tidak	

```

kode_provinsi <- c(
  "KEPULAUAN BANGKA BELITUNG" = "A",
  "KEPULAUAN RIAU" = "A",
  "LAMPUNG" = "A",
  "SUMATERA SELATAN" = "A",
  "KALIMANTAN BARAT" = "B",
  "SULAWESI BARAT" = "B",
  "SULAWESI SELATAN" = "B",
  "SULAWESI TENGAH" = "B",
  "SULAWESI TENGGARA" = "B",
  "GORONTALO" = "C",
  "KALIMANTAN UTARA" = "C",
  "MALUKU UTARA" = "C",
  "BALI" = "D",
  "BANTEN" = "D",
  "JAWA BARAT" = "D",
  "JAWA TIMUR" = "D",
  "KALIMANTAN SELATAN" = "D",
  "MALUKU" = "D",
  "DKI JAKARTA" = "E",
  "JAWA TENGAH" = "E",
  "NUSA TENGGARA BARAT" = "F",
  "NUSA TENGGARA TIMUR" = "G",
  "DI YOGYAKARTA" = "H",
  "SULAWESI UTARA" = "I",
  "ACEH" = "J",
  "BENGKULU" = "J",
  "JAMBI" = "J",
  "KALIMANTAN TENGAH" = "J",
  "KALIMANTAN TIMUR" = "J",
  "PAPUA" = "J",
  "PAPUA BARAT" = "J",
  "RIAU" = "J",
  "SUMATERA BARAT" = "J",
  "SUMATERA UTARA" = "J"
)
data_gabungan$Kode <- kode_provinsi[data_gabungan$Provinsi]

```

```
library(dplyr)
```

```

avg_per_kode <- data_gabungan %>%
  group_by(Kode) %>%
  summarise(MeanPersenUnmetNeed = mean(PersenUnmetNeed, na.rm = TRUE)) %>%
  arrange(desc(MeanPersenUnmetNeed)) # Mengurutkan dari tertinggi ke terendah

```

```
# Membuat barplot
```

```

barplot_data <- avg_per_kode$MeanPersenUnmetNeed
names(barplot_data) <- avg_per_kode$Kode # Menambahkan nama pada barplot

```

```

barplot(
  barplot_data,
  col = "skyblue",
  ylim = c(0, 0.1), # Membatasi skala vertikal dari 0 hingga 1
  main = "Barplot Rata-rata Persen Unmet Need Setiap Kelompok",
  xlab = "Kelompok",
  ylab = "Rata-rata Persen Unmet Need"
)

```

```
# Menghitung rata-rata keseluruhan
```

```
avg_overall <- mean(data_gabungan$PersenUnmetNeed, na.rm = TRUE)
```

```
# Menambahkan garis rata-rata keseluruhan
```

```
abline(h = avg_overall, col = "red", lty = 2)
```

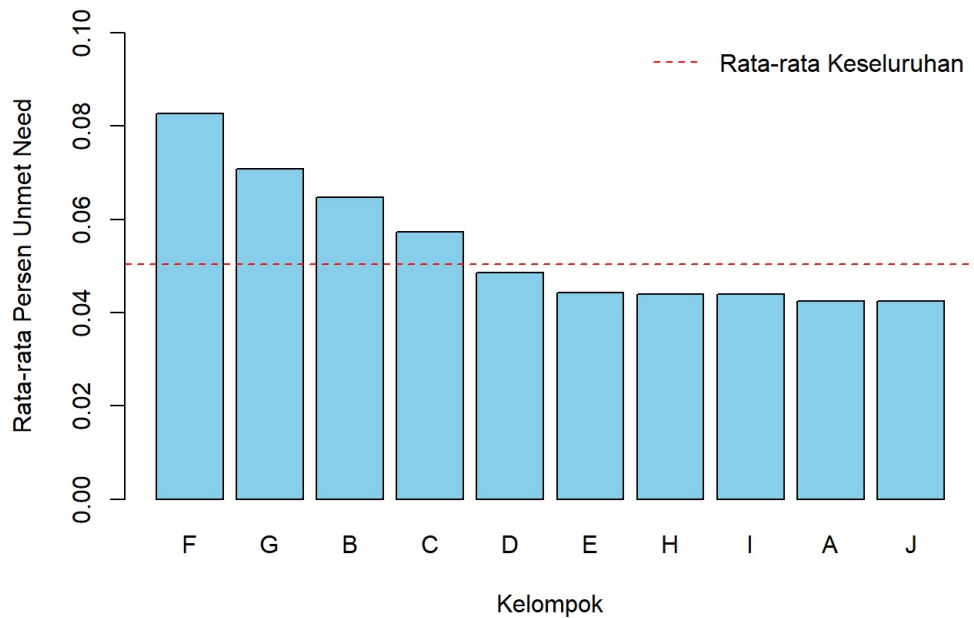
```
# Menambahkan legenda
```

```

legend(
  "topright",
  legend = c("Rata-rata Keseluruhan"),
  col = "red",
  lty = 2,
  bty = "n"
)

```

**Barplot Rata-rata Persen Unmet Need Setiap Kelompok**



```
avg_per_kode <- data_gabungan %>%
  group_by(Kode) %>%
  summarise(MeanPersenRural= mean(nonesensial_kapita, na.rm = TRUE)) %>%
  arrange(desc(MeanPersenRural)) # Mengurutkan dari tertinggi ke terendah

# Membuat barplot
barplot_data <- avg_per_kode$MeanPersenRural
names(barplot_data) <- avg_per_kode$Kode # Menambahkan nama pada barplot

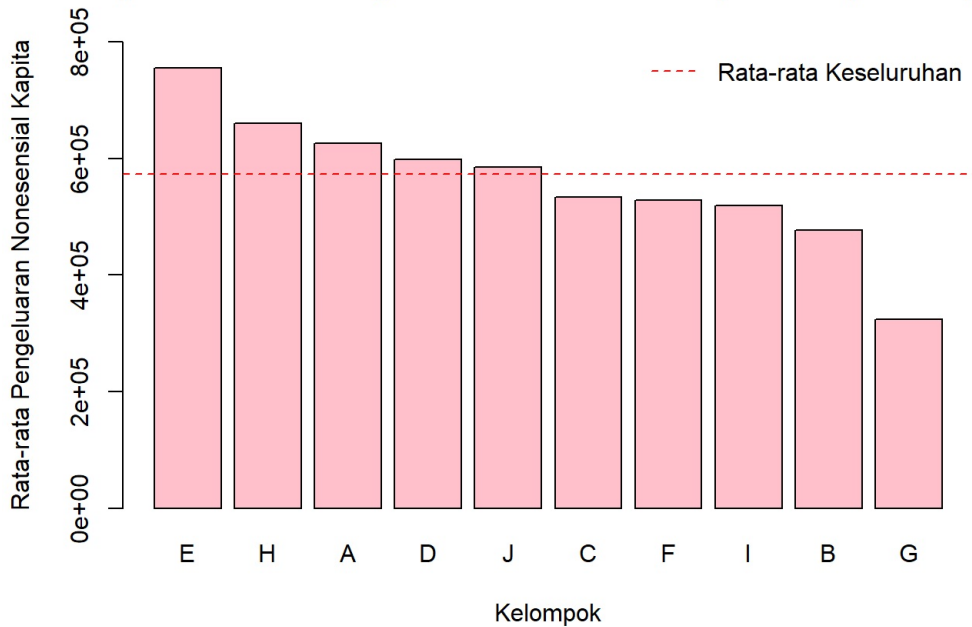
barplot(
  barplot_data,
  col = "pink",
  ylim = c(0, 800000), # Membatasi skala vertikal dari 0 hingga 1
  main = "Barplot Rata-rata Pengeluaran Nonesensial Kapita Setiap Kelompok ",
  xlab = "Kelompok",
  ylab = "Rata-rata Pengeluaran Nonesensial Kapita"
)

# Menghitung rata-rata keseluruhan
avg_overall <- mean(data_gabungan$nonesensial_kapita, na.rm = TRUE)

# Menambahkan garis rata-rata keseluruhan
abline(h = avg_overall, col = "red", lty = 2)

# Menambahkan legenda
legend(
  "topright",
  legend = c("Rata-rata Keseluruhan"),
  col = "red",
  lty = 2,
  bty = "n"
)
```

## Barplot Rata-rata Pengeluaran Nonesensial Kapita Setiap Kelompok



```
avg_per_kode <- data_gabungan %>%
  group_by(Kode) %>%
  summarise(MeanPersenRural= mean(PersenNoCoverage, na.rm = TRUE)) %>%
  arrange(desc(MeanPersenRural)) # Mengurutkan dari tertinggi ke terendah

# Membuat barplot
barplot_data <- avg_per_kode$MeanPersenRural
names(barplot_data) <- avg_per_kode$Kode # Menambahkan nama pada barplot

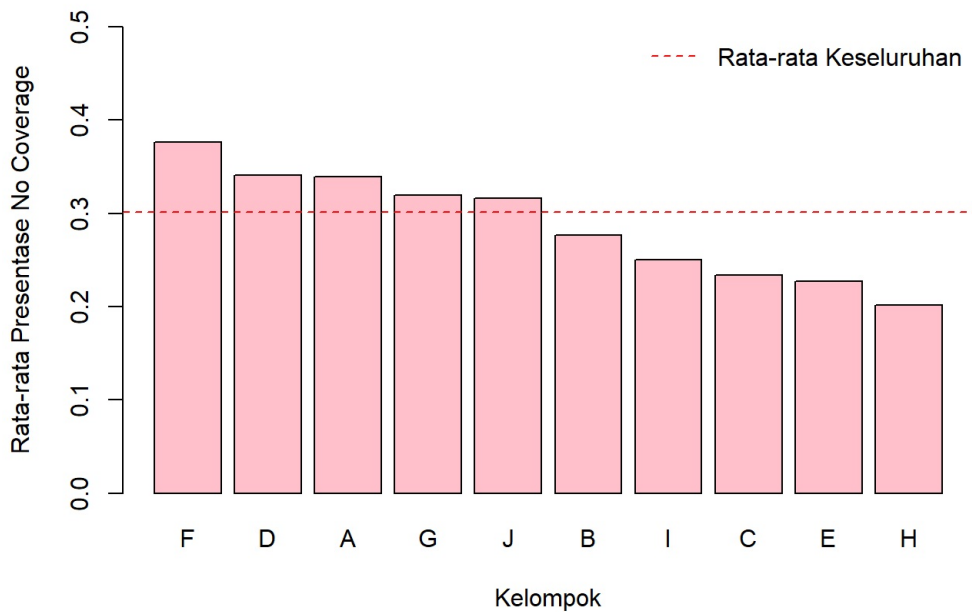
barplot(
  barplot_data,
  col = "pink",
  ylim = c(0, 0.5), # Membatasi skala vertikal dari 0 hingga 1
  main = "Barplot Rata-rata Presentase No Coverage Setiap Kelompok ",
  xlab = "Kelompok",
  ylab = "Rata-rata Presentase No Coverage"
)

# Menghitung rata-rata keseluruhan
avg_overall <- mean(data_gabungan$PersenNoCoverage, na.rm = TRUE)

# Menambahkan garis rata-rata keseluruhan
abline(h = avg_overall, col = "red", lty = 2)

# Menambahkan legenda
legend(
  "topright",
  legend = c("Rata-rata Keseluruhan"),
  col = "red",
  lty = 2,
  bty = "n"
)
```

## Barplot Rata-rata Presentase No Coverage Setiap Kelompok



```
avg_per_kode <- data_gabungan %>%
  group_by(Kode) %>%
  summarise(MeanPersenRural= mean(Umur, na.rm = TRUE)) %>%
  arrange(desc(MeanPersenRural)) # Mengurutkan dari tertinggi ke terendah

# Membuat barplot
barplot_data <- avg_per_kode$MeanPersenRural
names(barplot_data) <- avg_per_kode$Kode # Menambahkan nama pada barplot

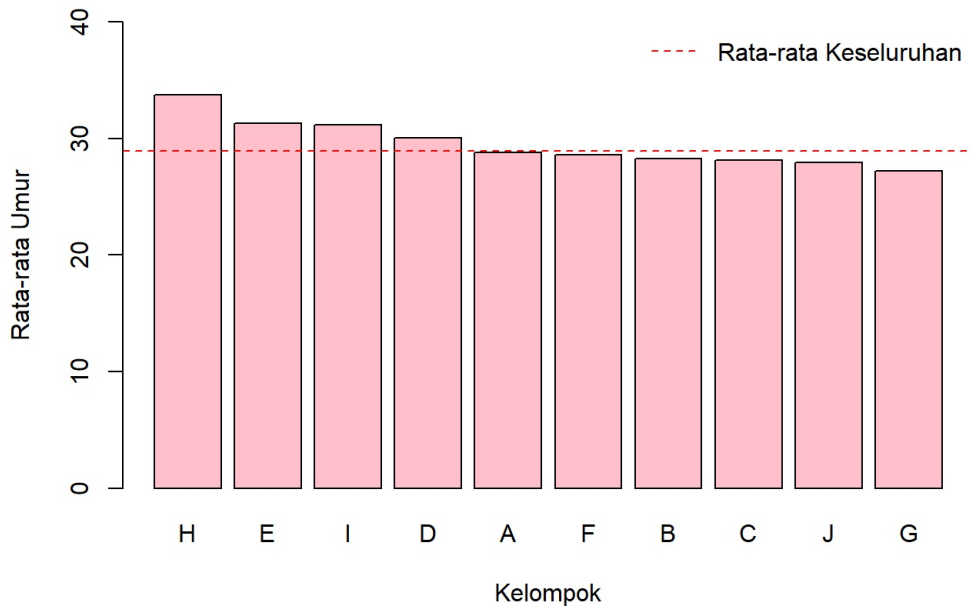
barplot(
  barplot_data,
  col = "pink",
  ylim = c(0, 40), # Membatasi skala vertikal dari 0 hingga 1
  main = "Barplot Rata-rata Umur Setiap Kelompok ",
  xlab = "Kelompok",
  ylab = "Rata-rata Umur"
)

# Menghitung rata-rata keseluruhan
avg_overall <- mean(data_gabungan$Umur, na.rm = TRUE)

# Menambahkan garis rata-rata keseluruhan
abline(h = avg_overall, col = "red", lty = 2)

# Menambahkan legenda
legend(
  "topright",
  legend = c("Rata-rata Keseluruhan"),
  col = "red",
  lty = 2,
  bty = "n"
)
```

## Barplot Rata-rata Umur Setiap Kelompok



```
avg_per_kode <- data_gabungan %>%
  group_by(Kode) %>%
  summarise(MeanPersenRural= mean(PersenRural, na.rm = TRUE)) %>%
  arrange(desc(MeanPersenRural)) # Mengurutkan dari tertinggi ke terendah

# Membuat barplot
barplot_data <- avg_per_kode$MeanPersenRural
names(barplot_data) <- avg_per_kode$Kode # Menambahkan nama pada barplot

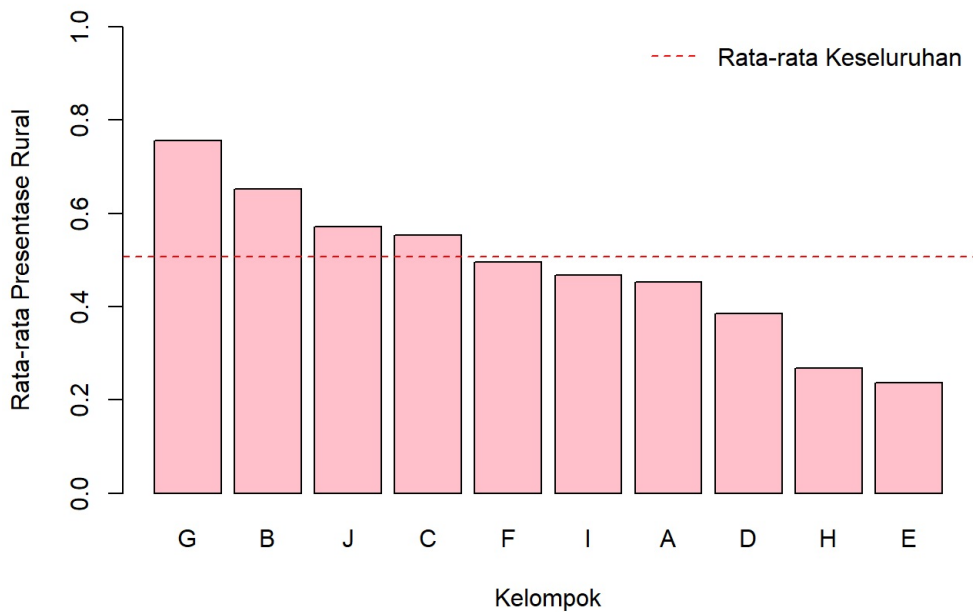
barplot(
  barplot_data,
  col = "pink",
  ylim = c(0, 1), # Membatasi skala vertikal dari 0 hingga 1
  main = "Barplot Rata-rata Presentase Rural Setiap Kelompok ",
  xlab = "Kelompok",
  ylab = "Rata-rata Presentase Rural"
)

# Menghitung rata-rata keseluruhan
avg_overall <- mean(data_gabungan$PersenRural, na.rm = TRUE)

# Menambahkan garis rata-rata keseluruhan
abline(h = avg_overall, col = "red", lty = 2)

# Menambahkan legenda
legend(
  "topright",
  legend = c("Rata-rata Keseluruhan"),
  col = "red",
  lty = 2,
  bty = "n"
)
```

## Barplot Rata-rata Presentase Rural Setiap Kelompok



```
avg_per_kode <- data_gabungan %>%
  group_by(Kode) %>%
  summarise(MeanPersenRural= mean(IPM, na.rm = TRUE)) %>%
  arrange(desc(MeanPersenRural)) # Mengurutkan dari tertinggi ke terendah

# Membuat barplot
barplot_data <- avg_per_kode$MeanPersenRural
names(barplot_data) <- avg_per_kode$Kode # Menambahkan nama pada barplot

barplot(
  barplot_data,
  col = "pink",
  ylim = c(0, 100), # Membatasi skala vertikal dari 0 hingga 1
  main = "Barplot Rata-rata IPM Setiap Kelompok ",
  xlab = "Kelompok",
  ylab = "Rata-rata IPM"
)

# Menghitung rata-rata keseluruhan
avg_overall <- mean(data_gabungan$IPM, na.rm = TRUE)

# Menambahkan garis rata-rata keseluruhan
abline(h = avg_overall, col = "red", lty = 2)

# Menambahkan legenda
legend(
  "topright",
  legend = c("Rata-rata Keseluruhan"),
  col = "red",
  lty = 2,
  bty = "n"
)
```

Barplot Rata-rata IPM Setiap Kelompok

