Master Thesis Round Two

Patrick Sssonzi

1/29/2021

# SPEED Vs DENSITY

## Enter Data

Bombo <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Bombo-Clean")

## New names:  
## \* `` -> ...3

Entebbe <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Entebbe-Clean")

## New names:  
## \* `` -> ...3

Hoima <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Hoima-Clean")

## New names:  
## \* `` -> ...3

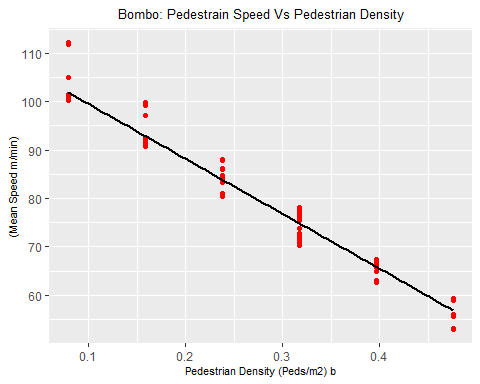
Jinja <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Jinja-Clean")

## New names:  
## \* `` -> ...3

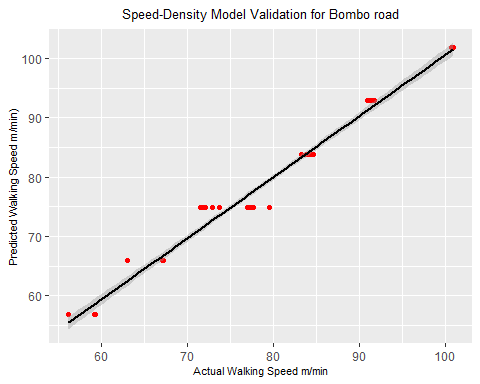
## Split data into training data set and validation data set

n=dim(Bombo)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Bombo\_train = Bombo[id,]  
Bombo\_validation = Bombo[-id,]  
  
  
n=dim(Entebbe)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Entebbe\_train = Entebbe[id,]  
Entebbe\_validation = Entebbe[-id,]  
  
  
n=dim(Hoima)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Hoima\_train = Hoima[id,]  
Hoima\_validation = Hoima[-id,]  
  
  
n=dim(Jinja)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Jinja\_train = Jinja[id,]  
Jinja\_validation = Jinja[-id,]

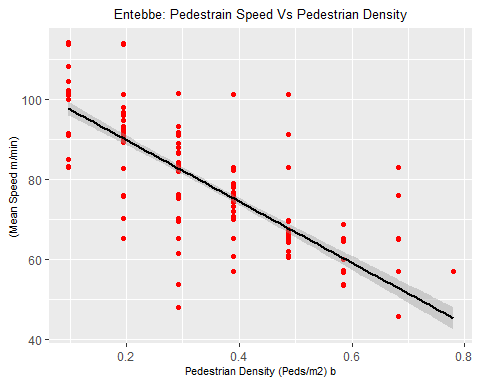
## Bombo: graph and linear regression and validation



##   
## Call:  
## lm(formula = `(Mean Speed m/min)` ~ `Pedestrian Density (Peds/m2) b`,   
## data = Bombo\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -4.483 -1.280 0.064 1.379 10.303   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 110.8444 0.3818 290.29 <2e-16 \*\*\*  
## `Pedestrian Density (Peds/m2) b` -113.3677 1.3344 -84.96 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.409 on 299 degrees of freedom  
## Multiple R-squared: 0.9602, Adjusted R-squared: 0.9601   
## F-statistic: 7218 on 1 and 299 DF, p-value: < 2.2e-16

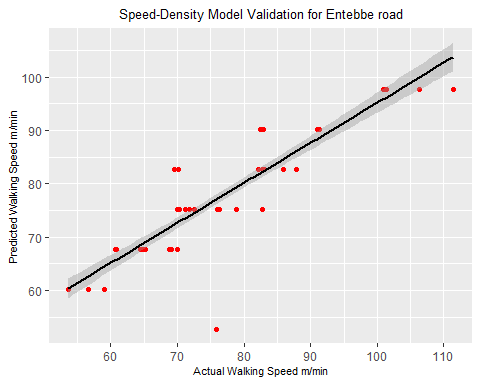


## Entebbe: graph and linear regression and validation

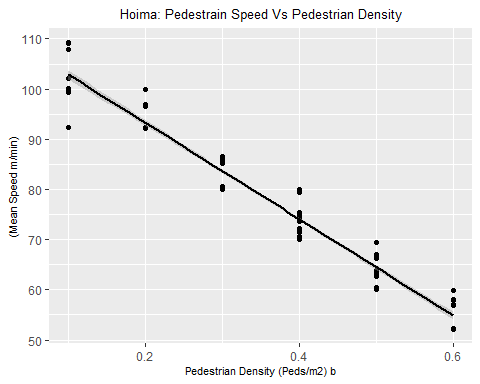


##   
## Call:  
## lm(formula = `(Mean Speed m/min)` ~ `Pedestrian Density (Peds/m2) b`,   
## data = Entebbe\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -34.785 -3.284 0.260 1.353 33.483   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 105.246 1.134 92.83 <2e-16 \*\*\*  
## `Pedestrian Density (Peds/m2) b` -76.919 3.060 -25.14 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7.591 on 318 degrees of freedom  
## (8 observations deleted due to missingness)  
## Multiple R-squared: 0.6652, Adjusted R-squared: 0.6642   
## F-statistic: 631.9 on 1 and 318 DF, p-value: < 2.2e-16

## [1] 26.79966

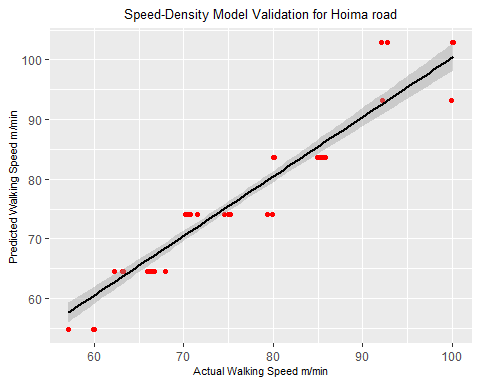


## Hoima: graph and linear regression and validation

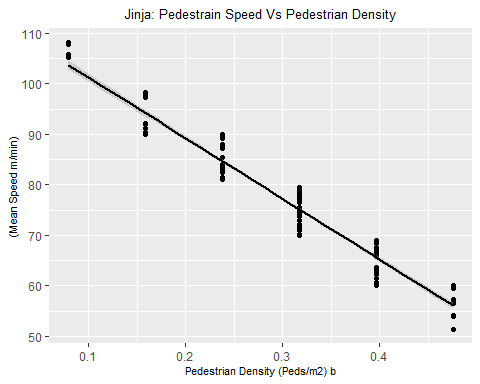


##   
## Call:  
## lm(formula = `(Mean Speed m/min)` ~ `Pedestrian Density (Peds/m2) b`,   
## data = Hoima\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -10.5763 -3.4370 0.6427 2.1058 6.7101   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 112.4546 0.5973 188.3 <2e-16 \*\*\*  
## `Pedestrian Density (Peds/m2) b` -95.9902 1.5142 -63.4 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 3.127 on 319 degrees of freedom  
## (7 observations deleted due to missingness)  
## Multiple R-squared: 0.9265, Adjusted R-squared: 0.9262   
## F-statistic: 4019 on 1 and 319 DF, p-value: < 2.2e-16

## [1] 12.76322

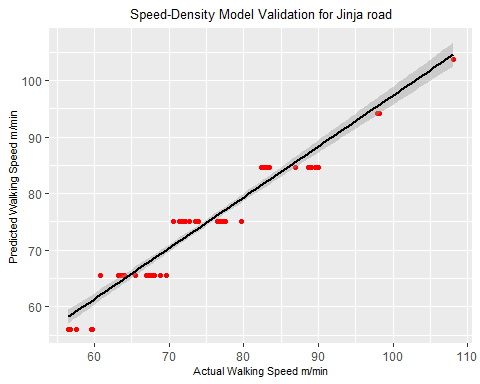


## Jinja: graph and linear regression and validation



##   
## Call:  
## lm(formula = `(Mean Speed m/min)` ~ `Pedestrian Density (Peds/m2) b`,   
## data = Jinja\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.5829 -2.3325 0.4846 2.0333 5.3324   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 113.2197 0.5767 196.32 <2e-16 \*\*\*  
## `Pedestrian Density (Peds/m2) b` -120.0446 1.7431 -68.87 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.815 on 310 degrees of freedom  
## Multiple R-squared: 0.9386, Adjusted R-squared: 0.9384   
## F-statistic: 4743 on 1 and 310 DF, p-value: < 2.2e-16

## [1] 7.527353



## Confidence intervals

#### Bombo  
confint(lm(data = Bombo\_train, formula = `(Mean Speed m/min)`~ `Pedestrian Density (Peds/m2) b`), level = .95)

## 2.5 % 97.5 %  
## (Intercept) 110.0929 111.5958  
## `Pedestrian Density (Peds/m2) b` -115.9938 -110.7417

#### Entebbe  
confint(lm(data = Entebbe\_train, formula = `(Mean Speed m/min)`~ `Pedestrian Density (Peds/m2) b`))

## 2.5 % 97.5 %  
## (Intercept) 103.01529 107.47632  
## `Pedestrian Density (Peds/m2) b` -82.93947 -70.89885

#### Hoima  
confint(lm(data = Hoima\_train, formula = `(Mean Speed m/min)`~ `Pedestrian Density (Peds/m2) b`))

## 2.5 % 97.5 %  
## (Intercept) 111.2795 113.62972  
## `Pedestrian Density (Peds/m2) b` -98.9692 -93.01118

#### Jinja  
confint(lm(data = Jinja\_train, formula = `(Mean Speed m/min)`~ `Pedestrian Density (Peds/m2) b`))

## 2.5 % 97.5 %  
## (Intercept) 112.0849 114.3544  
## `Pedestrian Density (Peds/m2) b` -123.4745 -116.6148

# FLOW Vs DENSITY

## Enter Data

Bombo <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Bombo-Reduced")

## New names:  
## \* `` -> ...3

Entebbe <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Entebbe-Reduced")

## New names:  
## \* `` -> ...3

Hoima <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Hoima-Reduced")

## New names:  
## \* `` -> ...3

Jinja <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Jinja-Reduced")

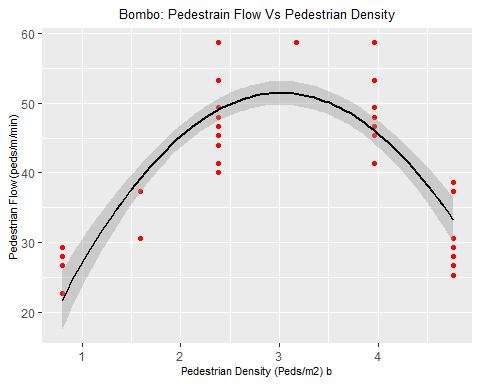
## New names:  
## \* `` -> ...3

### Split data into training data set and validation data set

Bombo[,12] <- Bombo[,12]\*10  
n=dim(Bombo)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Bombo\_train = Bombo[id,]  
Bombo\_validation = Bombo[-id,]  
  
  
  
Entebbe[,12] <- Entebbe[,12]\*10  
n=dim(Entebbe)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Entebbe\_train = Entebbe[id,]  
Entebbe\_validation = Entebbe[-id,]  
  
  
  
Hoima[,12] <- Hoima[,12]\*10  
n=dim(Hoima)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Hoima\_train = Hoima[id,]  
Hoima\_validation = Hoima[-id,]  
  
  
  
Jinja[,12] <- Jinja[,12]\*10  
n=dim(Jinja)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Jinja\_train = Jinja[id,]  
Jinja\_validation = Jinja[-id,]

## Bombo

ggplot(data = Bombo\_train,   
 aes(x = `Pedestrian Density (Peds/m2) b`,   
 y = `Pedestrian Flow (peds/m/min)`)) +   
 geom\_point(color='red') +  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y ~ poly(x, 2, raw = T))+  
 labs(title="Bombo: Pedestrain Flow Vs Pedestrian Density")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(axis.title.y = element\_text(size = 8))



### Bombo training summary

summary(Bombo\_poly <- lm(data = Bombo\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`Pedestrian Density (Peds/m2) b`,2, raw = T)))

##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`Pedestrian Density (Peds/m2) b`,   
## 2, raw = T), data = Bombo\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.0499 -4.5296 -0.8892 4.4441 12.8037   
##   
## Coefficients:  
## Estimate Std. Error t value  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 33.9974 0.7729 43.98  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -5.6549 0.2088 -27.09  
## Pr(>|t|)   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 <2e-16 \*\*\*  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.92 on 87 degrees of freedom  
## Multiple R-squared: 0.9832, Adjusted R-squared: 0.9828   
## F-statistic: 2550 on 2 and 87 DF, p-value: < 2.2e-16

coefficients(Bombo\_poly)

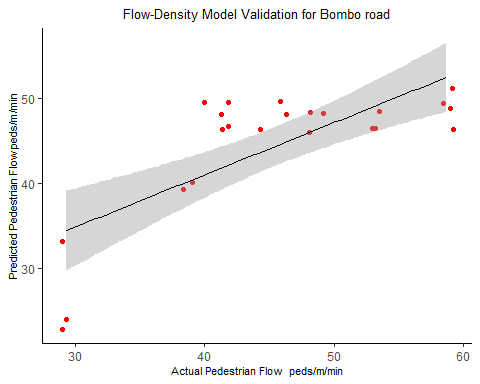
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1   
## 33.997441   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2   
## -5.654872

# coef(Bombo\_poly)  
  
confint(Bombo\_poly, level = 0.95)

## 2.5 % 97.5 %  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 32.461161 35.533720  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -6.069829 -5.239916

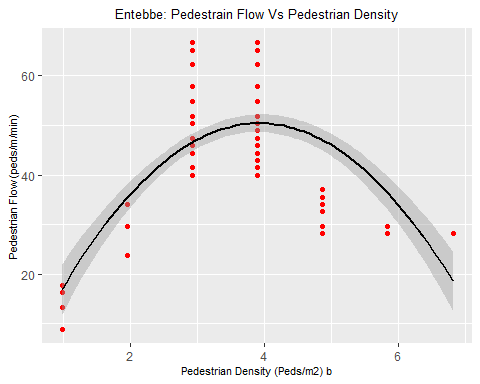
### Model Validation on validation data set - MSE and RMSE

predict\_Bombo\_poly <- predict(Bombo\_poly,   
 newdata = Bombo\_validation)  
  
### Model Validation Bombo  
  
ggplot()+  
 aes(x = Bombo\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Bombo\_poly)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow-Density Model Validation for Bombo road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Entebbe

ggplot(data = Entebbe\_train,   
 aes(x = `Pedestrian Density (Peds/m2) b`,   
 y = `Pedestrian Flow (peds/m/min)`)) +   
 geom\_point(color='red') +  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y ~ poly(x, 2, raw = T))+  
 labs(title="Entebbe: Pedestrain Flow Vs Pedestrian Density")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(axis.title.y = element\_text(size = 8))



### Entebbe training summary

summary(Entebbe\_poly <- lm(data = Entebbe\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`Pedestrian Density (Peds/m2) b`,2, raw = T)))

##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`Pedestrian Density (Peds/m2) b`,   
## 2, raw = T), data = Entebbe\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -18.151 -6.854 -3.474 7.868 19.859   
##   
## Coefficients:  
## Estimate Std. Error t value  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 25.7700 0.6975 36.95  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -3.3385 0.1523 -21.91  
## Pr(>|t|)   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 <2e-16 \*\*\*  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 9.158 on 145 degrees of freedom  
## Multiple R-squared: 0.9612, Adjusted R-squared: 0.9606   
## F-statistic: 1794 on 2 and 145 DF, p-value: < 2.2e-16

coefficients(Entebbe\_poly)

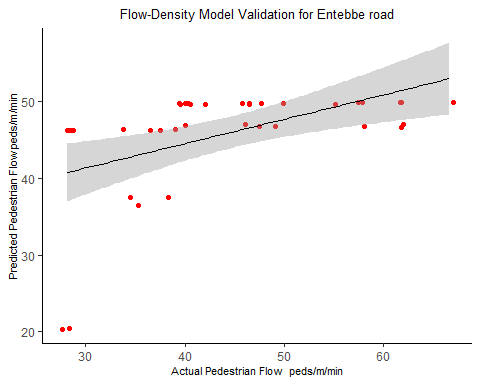
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1   
## 25.769977   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2   
## -3.338476

# coef(Entebbe\_poly)  
  
confint(Entebbe\_poly, level = 0.95)

## 2.5 % 97.5 %  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 24.391379 27.148575  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -3.639573 -3.037378

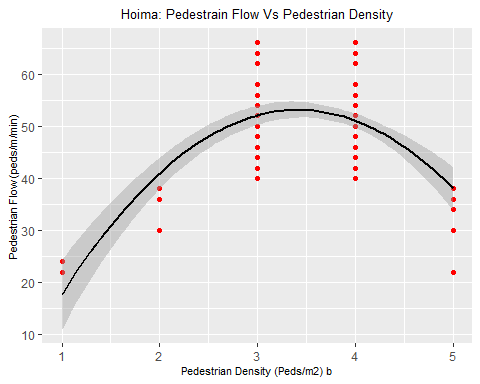
### Model Validation on validation data set - MSE and RMSE

predict\_Entebbe\_poly <- predict(Entebbe\_poly,   
 newdata = Entebbe\_validation)  
  
### Model Validation Entebbe  
  
ggplot()+  
 aes(x = Entebbe\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Entebbe\_poly)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow-Density Model Validation for Entebbe road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Hoima

ggplot(data = Hoima\_train,   
 aes(x = `Pedestrian Density (Peds/m2) b`,   
 y = `Pedestrian Flow (peds/m/min)`)) +   
 geom\_point(color='red') +  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y ~ poly(x, 2, raw = T))+  
 labs(title="Hoima: Pedestrain Flow Vs Pedestrian Density")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(axis.title.y = element\_text(size = 8))



### Hoima training summary

summary(Hoima\_poly <- lm(data = Hoima\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`Pedestrian Density (Peds/m2) b`,2, raw = T)))

##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`Pedestrian Density (Peds/m2) b`,   
## 2, raw = T), data = Hoima\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -18.747 -6.756 -1.729 5.215 15.215   
##   
## Coefficients:  
## Estimate Std. Error t value  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 30.8835 1.1688 26.42  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -4.5468 0.2976 -15.28  
## Pr(>|t|)   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 <2e-16 \*\*\*  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7.784 on 158 degrees of freedom  
## (8 observations deleted due to missingness)  
## Multiple R-squared: 0.9765, Adjusted R-squared: 0.9762   
## F-statistic: 3289 on 2 and 158 DF, p-value: < 2.2e-16

coefficients(Hoima\_poly)

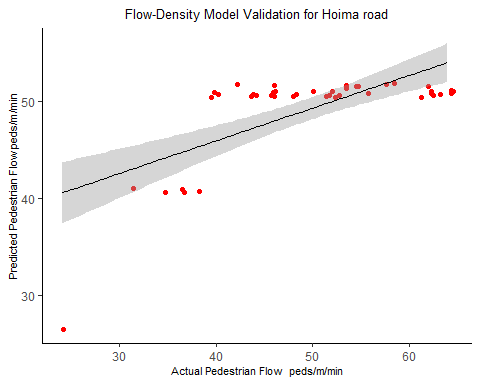
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1   
## 30.883524   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2   
## -4.546836

# coef(Hoima\_poly)  
  
confint(Hoima\_poly, level = 0.95)

## 2.5 % 97.5 %  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 28.575016 33.19203  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -5.134661 -3.95901

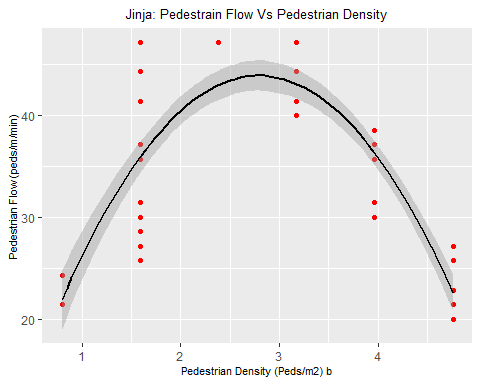
### Model Validation on validation data set - MSE and RMSE

predict\_Hoima\_poly <- predict(Hoima\_poly,   
 newdata = Hoima\_validation)  
  
### Model Validation Hoima  
  
ggplot()+  
 aes(x = Hoima\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Hoima\_poly)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow-Density Model Validation for Hoima road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Jinja

ggplot(data = Jinja\_train,   
 aes(x = `Pedestrian Density (Peds/m2) b`,   
 y = `Pedestrian Flow (peds/m/min)`)) +   
 geom\_point(color='red') +  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y ~ poly(x, 2, raw = T))+  
 labs(title="Jinja: Pedestrain Flow Vs Pedestrian Density")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(axis.title.y = element\_text(size = 8))



### Jinja training summary

summary(Jinja\_poly <- lm(data = Jinja\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`Pedestrian Density (Peds/m2) b`,2,raw = T)))

##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`Pedestrian Density (Peds/m2) b`,   
## 2, raw = T), data = Jinja\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -10.0463 -1.9775 -0.0162 3.2574 11.3823   
##   
## Coefficients:  
## Estimate Std. Error t value  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 31.4358 0.6536 48.09  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -5.6112 0.1601 -35.06  
## Pr(>|t|)   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 <2e-16 \*\*\*  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.238 on 78 degrees of freedom  
## Multiple R-squared: 0.9861, Adjusted R-squared: 0.9857   
## F-statistic: 2761 on 2 and 78 DF, p-value: < 2.2e-16

coefficients(Jinja\_poly)

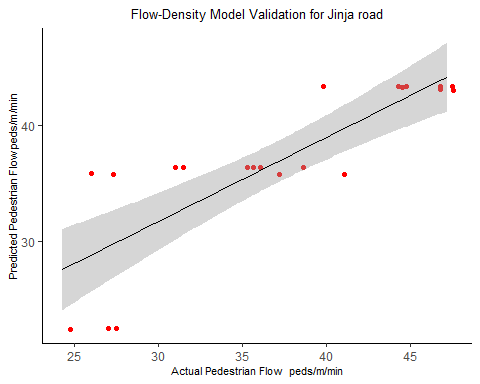
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1   
## 31.435792   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2   
## -5.611167

confint(Jinja\_poly, level = 0.95)

## 2.5 % 97.5 %  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 30.134535 32.73705  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -5.929825 -5.29251

### Model Validation on validation data set - MSE and RMSE

predict\_Jinja\_poly <- predict(Jinja\_poly,   
 newdata = Jinja\_validation)  
  
### Model Validation Jinja  
  
ggplot()+  
 aes(x = Jinja\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Jinja\_poly)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow-Density Model Validation for Jinja road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Maximum values

#### Bombo  
#### Max Flow  
max(Bombo$`Pedestrian Flow (peds/m/min)`)

## [1] 58.66667

#### Max Density  
Bombo$`Pedestrian Density (Peds/m2) b`[  
 which.max(Bombo$`Pedestrian Flow (peds/m/min)`)]

## [1] 2.380952

#### Entebbe  
#### Max Flow  
max(Entebbe$`Pedestrian Flow (peds/m/min)`)

## [1] 66.66667

#### Max Density  
Entebbe$`Pedestrian Density (Peds/m2) b`[  
 which.max(Entebbe$`Pedestrian Flow (peds/m/min)`)]

## [1] 2.923977

#### Hoima  
#### Max Flow  
max(Hoima$`Pedestrian Flow (peds/m/min)`, na.rm = TRUE)

## [1] 66

#### Max Density  
Hoima$`Pedestrian Density (Peds/m2) b`[  
 which.max(Hoima$`Pedestrian Flow (peds/m/min)`)]

## [1] 3

#### Jinja  
#### Max Flow  
max(Jinja$`Pedestrian Flow (peds/m/min)`, na.rm = TRUE)

## [1] 47.14286

#### Max Density  
Jinja$`Pedestrian Density (Peds/m2) b`[  
 which.max(Jinja$`Pedestrian Flow (peds/m/min)`)]

## [1] 1.587302

# FLOW Vs SPEED

## Enter Data

Bombo <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Bombo-Reduced")

## New names:  
## \* `` -> ...3

Entebbe <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Entebbe-Reduced")

## New names:  
## \* `` -> ...3

Hoima <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Hoima-Reduced")

## New names:  
## \* `` -> ...3

Jinja <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Jinja-Reduced")

## New names:  
## \* `` -> ...3

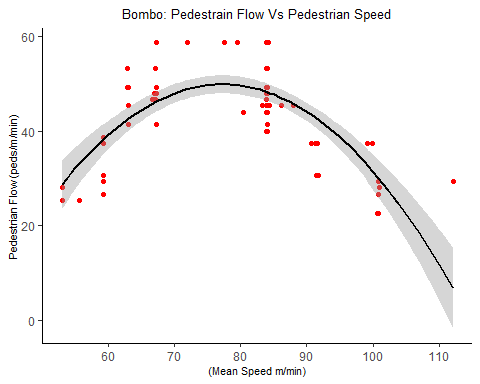
## Split data into training data set and validation data set

n=dim(Bombo)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Bombo\_train = Bombo[id,]  
Bombo\_validation = Bombo[-id,]  
  
  
n=dim(Entebbe)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Entebbe\_train = Entebbe[id,]  
Entebbe\_validation = Entebbe[-id,]  
  
  
n=dim(Hoima)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Hoima\_train = Hoima[id,]  
Hoima\_validation = Hoima[-id,]  
  
  
n=dim(Jinja)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Jinja\_train = Jinja[id,]  
Jinja\_validation = Jinja[-id,]

## Bombo

### Graph

ggplot(data = Bombo\_train,   
 aes(x = `(Mean Speed m/min)`,   
 y = `Pedestrian Flow (peds/m/min)`)) +   
 geom\_point(color='red') +  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y ~ poly(x, 2, raw = T))+  
 labs(title="Bombo: Pedestrain Flow Vs Pedestrian Speed")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



### Model Summary

summary(Bombo\_poly <- lm(data = Bombo\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`(Mean Speed m/min)`,2, raw = T)))

##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`(Mean Speed m/min)`,   
## 2, raw = T), data = Bombo\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -19.3015 -4.7899 0.1734 4.5314 13.9460   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## poly(`(Mean Speed m/min)`, 2, raw = T)1 1.329432 0.082457 16.123 < 2e-16  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 -0.009482 0.000980 -9.676 1.85e-15  
##   
## poly(`(Mean Speed m/min)`, 2, raw = T)1 \*\*\*  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 8.756 on 87 degrees of freedom  
## Multiple R-squared: 0.9633, Adjusted R-squared: 0.9625   
## F-statistic: 1142 on 2 and 87 DF, p-value: < 2.2e-16

coefficients(Bombo\_poly)

## poly(`(Mean Speed m/min)`, 2, raw = T)1 poly(`(Mean Speed m/min)`, 2, raw = T)2   
## 1.329432443 -0.009482288

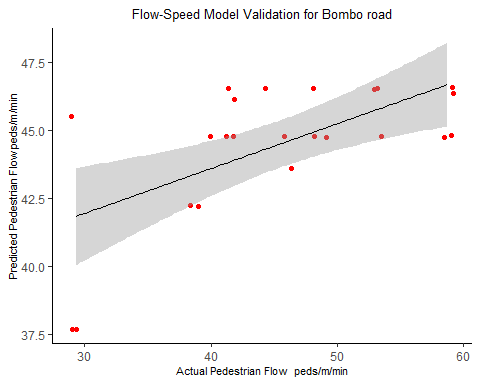
# coef(Bombo\_poly)  
  
confint(Bombo\_poly, level = 0.95)

## 2.5 % 97.5 %  
## poly(`(Mean Speed m/min)`, 2, raw = T)1 1.16553954 1.493325346  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 -0.01143009 -0.007534485

predict\_Bombo\_poly <- predict(Bombo\_poly,   
 newdata = Bombo\_validation)

### Model Validation Bombo

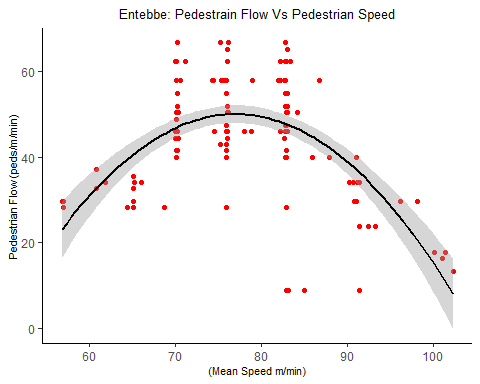
ggplot()+  
 aes(x = Bombo\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Bombo\_poly)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow-Speed Model Validation for Bombo road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Entebbe

### Graph

ggplot(data = Entebbe\_train,   
 aes(x = `(Mean Speed m/min)`,   
 y = `Pedestrian Flow (peds/m/min)`)) +   
 geom\_point(color='red') +  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y ~ poly(x, 2, raw = T))+  
 labs(title="Entebbe: Pedestrain Flow Vs Pedestrian Speed")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



### Model Summary

summary(Entebbe\_poly <- lm(data = Entebbe\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`(Mean Speed m/min)`,2, raw = T)))

##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`(Mean Speed m/min)`,   
## 2, raw = T), data = Entebbe\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -34.605 -6.757 0.225 10.059 23.149   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## poly(`(Mean Speed m/min)`, 2, raw = T)1 1.447280 0.113788 12.719 < 2e-16  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 -0.011129 0.001415 -7.867 7.67e-13  
##   
## poly(`(Mean Speed m/min)`, 2, raw = T)1 \*\*\*  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 12.63 on 145 degrees of freedom  
## Multiple R-squared: 0.9262, Adjusted R-squared: 0.9251   
## F-statistic: 909.3 on 2 and 145 DF, p-value: < 2.2e-16

coefficients(Entebbe\_poly)

## poly(`(Mean Speed m/min)`, 2, raw = T)1 poly(`(Mean Speed m/min)`, 2, raw = T)2   
## 1.44727971 -0.01112888

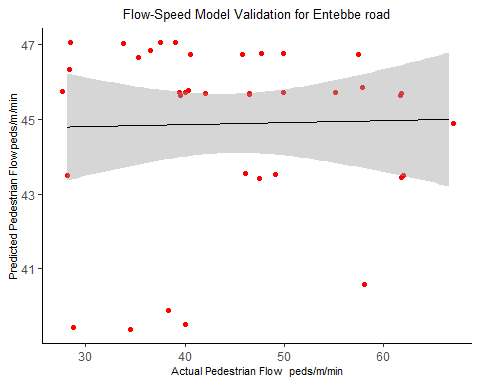
# coef(Bombo\_poly)  
  
confint(Entebbe\_poly, level = 0.95)

## 2.5 % 97.5 %  
## poly(`(Mean Speed m/min)`, 2, raw = T)1 1.22238257 1.672176851  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 -0.01392496 -0.008332802

predict\_Entebbe\_poly <- predict(Entebbe\_poly,   
 newdata = Entebbe\_validation)

### Model Validation Entebbe

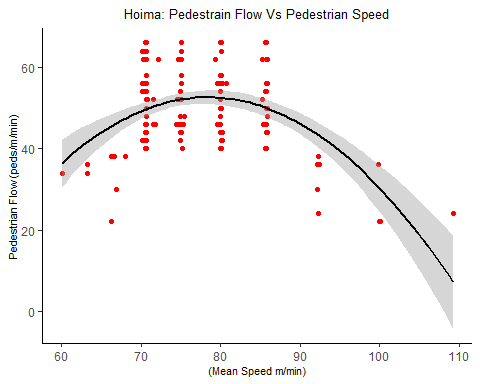
ggplot()+  
 aes(x = Entebbe\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Entebbe\_poly)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow-Speed Model Validation for Entebbe road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Hoima

### Graph

ggplot(data = Hoima\_train,   
 aes(x = `(Mean Speed m/min)`,   
 y = `Pedestrian Flow (peds/m/min)`)) +   
 geom\_point(color='red') +  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y ~ poly(x, 2, raw = T))+  
 labs(title="Hoima: Pedestrain Flow Vs Pedestrian Speed")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



### Model Summary

summary(Hoima\_poly <- lm(data = Hoima\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`(Mean Speed m/min)`,2, raw = T)))

##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`(Mean Speed m/min)`,   
## 2, raw = T), data = Hoima\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -29.3089 -6.6546 0.4802 6.3967 18.5355   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## poly(`(Mean Speed m/min)`, 2, raw = T)1 1.524562 0.086622 17.60 <2e-16  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 -0.011322 0.001097 -10.32 <2e-16  
##   
## poly(`(Mean Speed m/min)`, 2, raw = T)1 \*\*\*  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 9.155 on 158 degrees of freedom  
## (8 observations deleted due to missingness)  
## Multiple R-squared: 0.9676, Adjusted R-squared: 0.9671   
## F-statistic: 2356 on 2 and 158 DF, p-value: < 2.2e-16

coefficients(Hoima\_poly)

## poly(`(Mean Speed m/min)`, 2, raw = T)1 poly(`(Mean Speed m/min)`, 2, raw = T)2   
## 1.52456168 -0.01132193

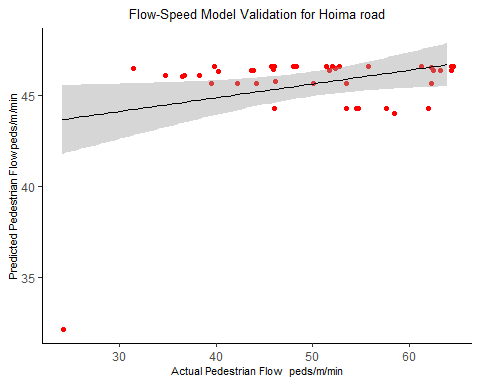
# coef(Bombo\_poly)  
  
confint(Hoima\_poly, level = 0.95)

## 2.5 % 97.5 %  
## poly(`(Mean Speed m/min)`, 2, raw = T)1 1.35347525 1.695648124  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 -0.01348927 -0.009154594

predict\_Hoima\_poly <- predict(Bombo\_poly,   
 newdata = Hoima\_validation)

### Model Validation Hoima

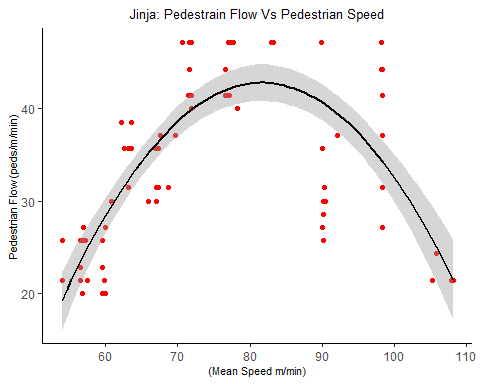
ggplot()+  
 aes(x = Hoima\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Hoima\_poly)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow-Speed Model Validation for Hoima road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Jinja

### Graph

ggplot(data = Jinja\_train,   
 aes(x = `(Mean Speed m/min)`,   
 y = `Pedestrian Flow (peds/m/min)`)) +   
 geom\_point(color='red') +  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y ~ poly(x, 2, raw = T))+  
 labs(title="Jinja: Pedestrain Flow Vs Pedestrian Speed")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



### Model Summary

summary(Jinja\_poly <- lm(data = Jinja\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`(Mean Speed m/min)`,2, raw = T)))

##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`(Mean Speed m/min)`,   
## 2, raw = T), data = Jinja\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -13.038 -7.030 1.163 6.218 12.113   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## poly(`(Mean Speed m/min)`, 2, raw = T)1 0.8398796 0.0618335 13.583 < 2e-16  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 -0.0048696 0.0007416 -6.566 5.21e-09  
##   
## poly(`(Mean Speed m/min)`, 2, raw = T)1 \*\*\*  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 8.217 on 78 degrees of freedom  
## Multiple R-squared: 0.9476, Adjusted R-squared: 0.9463   
## F-statistic: 705.7 on 2 and 78 DF, p-value: < 2.2e-16

coefficients(Jinja\_poly)

## poly(`(Mean Speed m/min)`, 2, raw = T)1 poly(`(Mean Speed m/min)`, 2, raw = T)2   
## 0.839879570 -0.004869641

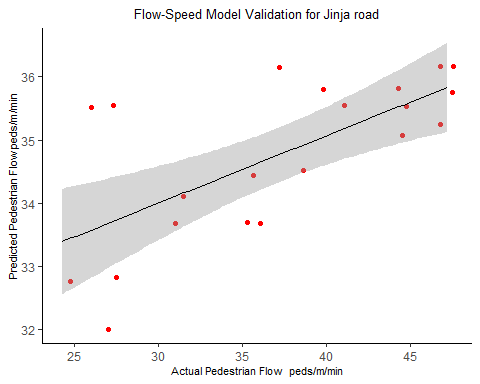
# coef(Bombo\_poly)  
  
confint(Jinja\_poly, level = 0.95)

## 2.5 % 97.5 %  
## poly(`(Mean Speed m/min)`, 2, raw = T)1 0.716778625 0.962980515  
## poly(`(Mean Speed m/min)`, 2, raw = T)2 -0.006346099 -0.003393183

predict\_Jinja\_poly <- predict(Jinja\_poly,   
 newdata = Jinja\_validation)

### Model Validation Jinja

ggplot()+  
 aes(x = Jinja\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Jinja\_poly)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow-Speed Model Validation for Jinja road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



# FLOW Vs DENSITY + SPEED (THREE DIMENSIONAL)

## Enter Data

Bombo <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Bombo-Reduced")

## New names:  
## \* `` -> ...3

Entebbe <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Entebbe-Reduced")

## New names:  
## \* `` -> ...3

Hoima <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Hoima-Reduced")

## New names:  
## \* `` -> ...3

Jinja <- read\_excel("Speed Density Flow.xlsx",   
 sheet = "Jinja-Reduced")

## New names:  
## \* `` -> ...3

Combined\_2 = rbind(Bombo,Entebbe,Hoima,Jinja)  
Combined\_2[,12] <- Combined\_2[,12]\*10

## Split data into training data set and validation data set

n=dim(Bombo)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Bombo\_train = Bombo[id,]  
Bombo\_validation = Bombo[-id,]  
  
  
n=dim(Entebbe)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Entebbe\_train = Entebbe[id,]  
Entebbe\_validation = Entebbe[-id,]  
  
  
n=dim(Hoima)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Hoima\_train = Hoima[id,]  
Hoima\_validation = Hoima[-id,]  
  
  
n=dim(Jinja)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Jinja\_train = Jinja[id,]  
Jinja\_validation = Jinja[-id,]

## Bombo

### Model Summary three dimension

summary(Bombo\_unknown <- lm(data = Bombo\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`Pedestrian Density (Peds/m2) b`,2, raw = T) +  
 poly(1/(log10(`(Mean Speed m/min)`)),2, raw = T)))

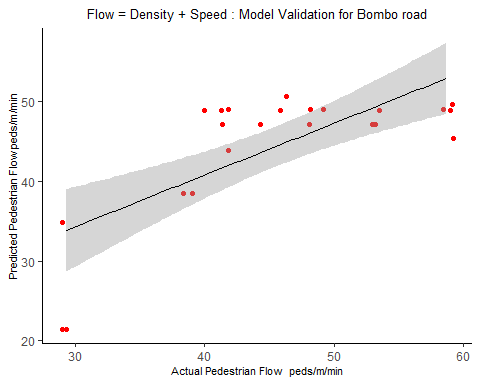
##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`Pedestrian Density (Peds/m2) b`,   
## 2, raw = T) + poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T),   
## data = Bombo\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -8.9575 -4.5889 -0.5173 4.4279 11.5854   
##   
## Coefficients:  
## Estimate Std. Error t value  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 379.85 29.78 12.757  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -521.22 61.01 -8.543  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 286.86 145.89 1.966  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 -596.28 295.36 -2.019  
## Pr(>|t|)   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 < 2e-16 \*\*\*  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 4.54e-13 \*\*\*  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 0.0525 .   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 0.0467 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5.824 on 85 degrees of freedom  
## Multiple R-squared: 0.9841, Adjusted R-squared: 0.9834   
## F-statistic: 1318 on 4 and 85 DF, p-value: < 2.2e-16

coefficients(Bombo\_unknown)

## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1   
## 379.8505   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2   
## -521.2236   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1   
## 286.8554   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2   
## -596.2808

### Model Validation Bombo three dimensions

predict\_Bombo\_poly\_3 <- predict(Bombo\_unknown,   
 newdata = Bombo\_validation)  
  
ggplot()+  
 aes(x = Bombo\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Bombo\_poly\_3)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow = Density + Speed : Model Validation for Bombo road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Entebbe

### Model Summary

summary(Entebbe\_unknown <- lm(data = Entebbe\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`Pedestrian Density (Peds/m2) b`,2, raw = T) +  
 poly(1/(log10(`(Mean Speed m/min)`)),2, raw = T)))

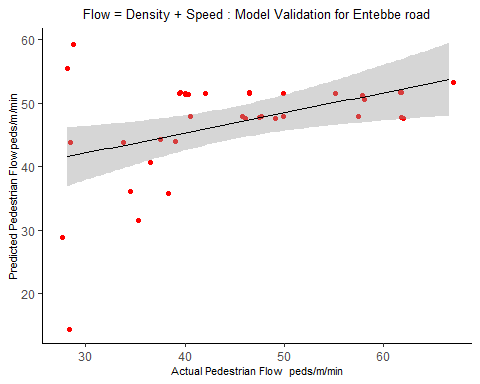
##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`Pedestrian Density (Peds/m2) b`,   
## 2, raw = T) + poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T),   
## data = Entebbe\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -15.601 -6.660 -1.586 6.585 19.069   
##   
## Coefficients:  
## Estimate Std. Error t value  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 352.87 25.23 13.986  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -401.37 27.91 -14.379  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 269.13 75.33 3.573  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 -594.52 153.25 -3.879  
## Pr(>|t|)   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 < 2e-16 \*\*\*  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 < 2e-16 \*\*\*  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 0.000482 \*\*\*  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 0.000159 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 8.608 on 143 degrees of freedom  
## Multiple R-squared: 0.9662, Adjusted R-squared: 0.9652   
## F-statistic: 1021 on 4 and 143 DF, p-value: < 2.2e-16

coefficients(Entebbe\_unknown)

## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1   
## 352.8658   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2   
## -401.3689   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1   
## 269.1315   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2   
## -594.5181

### Model Validation Entebbe three dimensions

predict\_Entebbe\_poly\_3 <- predict(Entebbe\_unknown,   
 newdata = Entebbe\_validation)  
  
ggplot()+  
 aes(x = Entebbe\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Entebbe\_poly\_3)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow = Density + Speed : Model Validation for Entebbe road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Hoima

### Model Summary three dimensions

summary(Hoima\_unknown <- lm(data = Hoima\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`Pedestrian Density (Peds/m2) b`,2, raw = T) +  
 poly(1/(log10(`(Mean Speed m/min)`)),2, raw = T)))

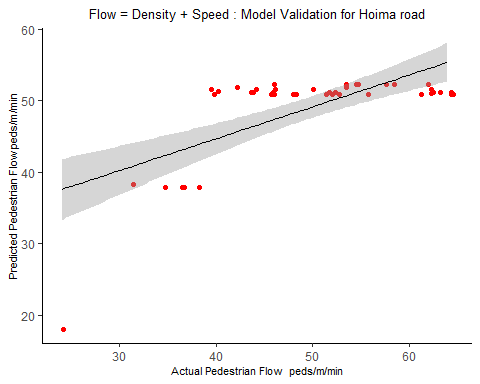
##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`Pedestrian Density (Peds/m2) b`,   
## 2, raw = T) + poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T),   
## data = Hoima\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -16.2355 -5.5850 -0.6401 5.1163 15.1216   
##   
## Coefficients:  
## Estimate Std. Error t value  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 416.61 42.27 9.856  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -599.07 66.90 -8.954  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 -26.30 105.55 -0.249  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 -19.43 210.19 -0.092  
## Pr(>|t|)   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 < 2e-16 \*\*\*  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 9.66e-16 \*\*\*  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 0.804   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 0.926   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7.663 on 156 degrees of freedom  
## (8 observations deleted due to missingness)  
## Multiple R-squared: 0.9776, Adjusted R-squared: 0.977   
## F-statistic: 1699 on 4 and 156 DF, p-value: < 2.2e-16

coefficients(Hoima\_unknown)

## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1   
## 416.60992   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2   
## -599.06522   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1   
## -26.30018   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2   
## -19.43448

### Model Validation Hoima three dimensions

predict\_Hoima\_poly\_3 <- predict(Hoima\_unknown,   
 newdata = Hoima\_validation)  
  
ggplot()+  
 aes(x = Hoima\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Hoima\_poly\_3)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow = Density + Speed : Model Validation for Hoima road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



## Jinja

### Model Summary three dimensions

summary(Jinja\_unknown <- lm(data = Jinja\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`Pedestrian Density (Peds/m2) b`,2, raw = T) +  
 poly(1/(log10(`(Mean Speed m/min)`)),2, raw = T)))

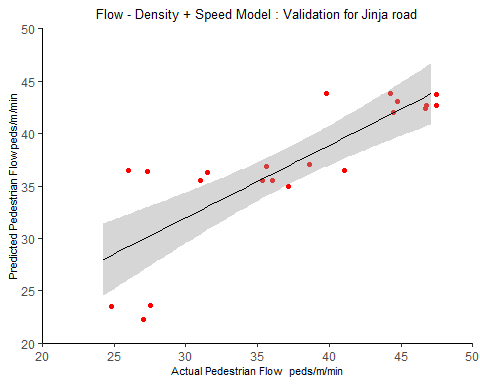
##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`Pedestrian Density (Peds/m2) b`,   
## 2, raw = T) + poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T),   
## data = Jinja\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -9.3460 -2.3509 -0.1666 3.2759 10.6725   
##   
## Coefficients:  
## Estimate Std. Error t value  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 322.03 21.80 14.770  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -518.55 38.25 -13.555  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 148.99 81.86 1.820  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 -303.04 167.62 -1.808  
## Pr(>|t|)   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 <2e-16 \*\*\*  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 <2e-16 \*\*\*  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 0.0727 .   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 0.0746 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.202 on 76 degrees of freedom  
## Multiple R-squared: 0.9867, Adjusted R-squared: 0.986   
## F-statistic: 1405 on 4 and 76 DF, p-value: < 2.2e-16

coefficients(Jinja\_unknown)

## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1   
## 322.0342   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2   
## -518.5534   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1   
## 148.9883   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2   
## -303.0390

### Model Validation Jinja three dimensions

predict\_Jinja\_poly\_3 <- predict(Jinja\_unknown,   
 newdata = Jinja\_validation)  
  
ggplot()+  
 aes(x = Jinja\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Jinja\_poly\_3)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow - Density + Speed Model : Validation for Jinja road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))+  
 scale\_x\_continuous(expand = c(0,0), limits = c(20,50),   
 breaks = seq(20, 50, by = 5))+  
 scale\_y\_continuous(expand = c(0,0), limits = c(20,50),   
 breaks = seq(20, 50, by = 5))



## Combined

## Split data into training data set and validation data set

## Split data into training data set and validation data set  
  
n=dim(Combined\_2)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Combined\_2\_train = Combined\_2[id,]  
Combined\_2\_validation = Combined\_2[-id,]

### Model Summary three dimensions

summary(Combined\_2\_unknown <- lm(data = Combined\_2\_train,   
 formula = `Pedestrian Flow (peds/m/min)` ~   
 0 + poly(`Pedestrian Density (Peds/m2) b`,2, raw = T) +  
 poly(1/(log10(`(Mean Speed m/min)`)),2, raw = T)))

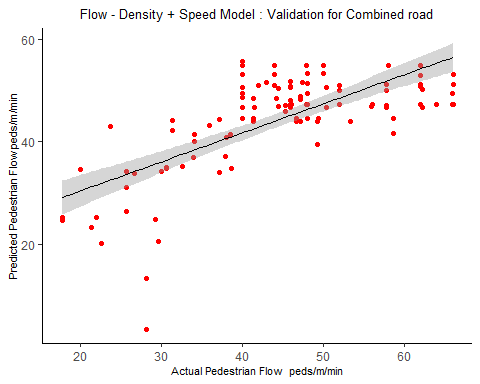
##   
## Call:  
## lm(formula = `Pedestrian Flow (peds/m/min)` ~ 0 + poly(`Pedestrian Density (Peds/m2) b`,   
## 2, raw = T) + poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T),   
## data = Combined\_2\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -32.043 -5.821 -0.574 5.615 24.759   
##   
## Coefficients:  
## Estimate Std. Error  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 34.3133 1.5236  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -4.1080 0.1998  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 417.8295 32.8749  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 -855.3052 67.2317  
## t value Pr(>|t|)   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1 22.52 <2e-16 \*\*\*  
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2 -20.56 <2e-16 \*\*\*  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1 12.71 <2e-16 \*\*\*  
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2 -12.72 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 8.16 on 474 degrees of freedom  
## (7 observations deleted due to missingness)  
## Multiple R-squared: 0.9689, Adjusted R-squared: 0.9686   
## F-statistic: 3691 on 4 and 474 DF, p-value: < 2.2e-16

coefficients(Combined\_2\_unknown)

## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)1   
## 34.313278   
## poly(`Pedestrian Density (Peds/m2) b`, 2, raw = T)2   
## -4.108047   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)1   
## 417.829471   
## poly(1/(log10(`(Mean Speed m/min)`)), 2, raw = T)2   
## -855.305219

### Model Validation Jinja three dimensions

predict\_Combined\_2\_poly\_3 <- predict(Combined\_2\_unknown,   
 newdata = Combined\_2\_validation)  
  
ggplot()+  
 aes(x = Combined\_2\_validation$`Pedestrian Flow (peds/m/min)`,   
 y = as.vector(predict\_Combined\_2\_poly\_3)) +   
 geom\_point(color="red", position = "jitter")+  
 geom\_smooth(method = "lm", se = TRUE, color="black",   
 formula = y~x, size=0.5)+  
 labs(title="Flow - Density + Speed Model : Validation for Combined road",   
 x="Actual Pedestrian Flow peds/m/min",   
 y="Predicted Pedestrian Flow peds/m/min")+  
 theme(plot.title = element\_text(hjust = 0.5, size = 10))+  
 theme(axis.title.y = element\_text(size = 8))+  
 theme(axis.title.x = element\_text(size = 8))+  
 theme(panel.grid.major = element\_blank(),   
 panel.grid.minor = element\_blank(),  
 panel.background = element\_blank(),   
 axis.line = element\_line(colour = "black"))



# PRINCIPAL COMPONENT ANALYSIS

## Enter Data

Bombo\_Clean = read\_excel("Speed Density Flow.xlsx",   
 sheet = "Bombo-Clean")

## New names:  
## \* `` -> ...3

Entebbe\_Clean = read\_excel("Speed Density Flow.xlsx",   
 sheet = "Entebbe-Clean")

## New names:  
## \* `` -> ...3

Hoima\_Clean = read\_excel("Speed Density Flow.xlsx",   
 sheet = "Hoima-Clean")

## New names:  
## \* `` -> ...3

Jinja\_Clean = read\_excel("Speed Density Flow.xlsx",   
 sheet = "Jinja-Clean")

## New names:  
## \* `` -> ...3

Combined\_3 = rbind(Bombo\_Clean,Entebbe\_Clean,Hoima\_Clean,Jinja\_Clean)  
Combined\_3[,12] <- Combined\_3[,12]\*10

## Split data into training data set and validation data set

n=dim(Bombo)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Bombo\_train = Bombo[id,]  
Bombo\_validation = Bombo[-id,]  
  
  
n=dim(Entebbe)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Entebbe\_train = Entebbe[id,]  
Entebbe\_validation = Entebbe[-id,]  
  
  
n=dim(Hoima)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Hoima\_train = Hoima[id,]  
Hoima\_validation = Hoima[-id,]  
  
  
n=dim(Jinja)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Jinja\_train = Jinja[id,]  
Jinja\_validation = Jinja[-id,]  
  
  
n=dim(Combined\_3)[1]  
set.seed(12345)  
id=sample(1:n, floor(n\*0.8))  
Combined\_3\_train = Combined\_3[id,]  
Combined\_3\_validation = Combined\_3[-id,]

## Creating many new variables to be reduced by principal component analysis

new\_variables = cbind.data.frame(  
 flow = Bombo\_train$`Pedestrian Flow (peds/m/min)`,  
 mean\_speed = Bombo\_train$`(Mean Speed m/min)`,  
 mean\_speed\_2 = Bombo\_train$`(Mean Speed m/min)`^2,  
 mean\_speed\_3 = Bombo\_train$`(Mean Speed m/min)`^3,  
 log\_speed = log(Bombo\_train$`(Mean Speed m/min)`),  
 inv\_speed = Bombo\_train$`(Mean Speed m/min)`^(-1),  
 inv\_log\_speed = log(Bombo\_train$`(Mean Speed m/min)`)^(-1),  
 density = Bombo\_train$`Pedestrian Density (Peds/m2) b`,  
 density\_2 = Bombo\_train$`Pedestrian Density (Peds/m2) b`,  
 density\_3 = Bombo\_train$`Pedestrian Density (Peds/m2) b`^3,  
 log\_density = log(Bombo\_train$`Pedestrian Flow (peds/m/min)`),  
 inv\_density = Bombo\_train$`Pedestrian Flow (peds/m/min)`^(-1),  
 inv\_log\_density = log(Bombo\_train$`Pedestrian Density (Peds/m2) b`)^(-1)  
 )  
  
  
head(new\_variables, 10)

## flow mean\_speed mean\_speed\_2 mean\_speed\_3 log\_speed inv\_speed  
## 1 37.33333 99.88109 9976.233 996437.1 4.603980 0.01001190  
## 2 48.00000 83.97201 7051.298 592111.7 4.430484 0.01190873  
## 3 41.33333 67.25380 4523.074 304193.9 4.208474 0.01486905  
## 4 48.00000 67.16418 4511.027 302979.4 4.207140 0.01488889  
## 5 48.00000 67.14628 4508.623 302737.3 4.206874 0.01489286  
## 6 40.00000 83.97201 7051.298 592111.7 4.430484 0.01190873  
## 7 53.33333 84.00000 7056.000 592704.0 4.430817 0.01190476  
## 8 48.00000 67.02128 4491.852 301049.6 4.205010 0.01492063  
## 9 58.66667 72.00000 5184.000 373248.0 4.276666 0.01388889  
## 10 49.33333 67.23586 4520.661 303950.5 4.208207 0.01487302  
## inv\_log\_speed density density\_2 density\_3 log\_density inv\_density  
## 1 0.2172034 0.1587302 0.1587302 0.003999248 3.619887 0.02678571  
## 2 0.2257090 0.2380952 0.2380952 0.013497462 3.871201 0.02083333  
## 3 0.2376158 0.3968254 0.3968254 0.062488252 3.721669 0.02419355  
## 4 0.2376912 0.3968254 0.3968254 0.062488252 3.871201 0.02083333  
## 5 0.2377062 0.3968254 0.3968254 0.062488252 3.871201 0.02083333  
## 6 0.2257090 0.2380952 0.2380952 0.013497462 3.688879 0.02500000  
## 7 0.2256920 0.2380952 0.2380952 0.013497462 3.976562 0.01875000  
## 8 0.2378116 0.3968254 0.3968254 0.062488252 3.871201 0.02083333  
## 9 0.2338270 0.3174603 0.3174603 0.031993985 4.071872 0.01704545  
## 10 0.2376309 0.3968254 0.3968254 0.062488252 3.898600 0.02027027  
## inv\_log\_density  
## 1 -0.5433160  
## 2 -0.6968231  
## 3 -1.0819479  
## 4 -1.0819479  
## 5 -1.0819479  
## 6 -0.6968231  
## 7 -0.6968231  
## 8 -1.0819479  
## 9 -0.8715338  
## 10 -1.0819479

new\_variables.pca <- prcomp(new\_variables[,-1], center = TRUE,scale. = TRUE)  
  
summary(new\_variables.pca)

## Importance of components:  
## PC1 PC2 PC3 PC4 PC5 PC6 PC7  
## Standard deviation 3.1188 1.4569 0.32207 0.17800 0.09598 0.07385 0.02280  
## Proportion of Variance 0.8106 0.1769 0.00864 0.00264 0.00077 0.00045 0.00004  
## Cumulative Proportion 0.8106 0.9875 0.99609 0.99873 0.99950 0.99995 1.00000  
## PC8 PC9 PC10 PC11 PC12  
## Standard deviation 0.005987 0.0009206 2.282e-05 1.454e-07 5.842e-17  
## Proportion of Variance 0.000000 0.0000000 0.000e+00 0.000e+00 0.000e+00  
## Cumulative Proportion 1.000000 1.0000000 1.000e+00 1.000e+00 1.000e+00

# str(new\_variables.pca)  
  
# library(devtools)  
# install\_github("vqv/ggbiplot")

## Sample of Principle Components

head(new\_variables.pca$x[,1:2])

## PC1 PC2  
## [1,] -4.289069 -0.7398230  
## [2,] -1.209006 0.7566013  
## [3,] 3.090346 -0.1759684  
## [4,] 3.127033 0.5883152  
## [5,] 3.129809 0.5882911  
## [6,] -1.237015 -0.1826733

head(new\_variables.pca$rotation[, 1:2])

## PC1 PC2  
## mean\_speed -0.3194979 -0.02450822  
## mean\_speed\_2 -0.3161757 -0.07820626  
## mean\_speed\_3 -0.3094820 -0.12928373  
## log\_speed -0.3196373 0.02936449  
## inv\_speed 0.3169874 -0.08128147  
## inv\_log\_speed 0.3187277 -0.05372920

## Visualising principal component analysis

library(ggbiplot)

## Loading required package: plyr

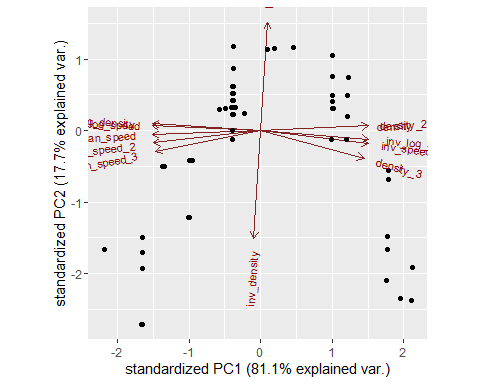
##   
## Attaching package: 'plyr'

## The following objects are masked from 'package:plotly':  
##   
## arrange, mutate, rename, summarise

## Loading required package: scales

## Loading required package: grid

ggbiplot(new\_variables.pca)



## Applying principal component analysis

new\_variables.pca <- prcomp(new\_variables[,-1], center = TRUE,scale. = TRUE)  
  
summary(new\_variables.pca)

## Importance of components:  
## PC1 PC2 PC3 PC4 PC5 PC6 PC7  
## Standard deviation 3.1188 1.4569 0.32207 0.17800 0.09598 0.07385 0.02280  
## Proportion of Variance 0.8106 0.1769 0.00864 0.00264 0.00077 0.00045 0.00004  
## Cumulative Proportion 0.8106 0.9875 0.99609 0.99873 0.99950 0.99995 1.00000  
## PC8 PC9 PC10 PC11 PC12  
## Standard deviation 0.005987 0.0009206 2.282e-05 1.454e-07 5.842e-17  
## Proportion of Variance 0.000000 0.0000000 0.000e+00 0.000e+00 0.000e+00  
## Cumulative Proportion 1.000000 1.0000000 1.000e+00 1.000e+00 1.000e+00

str(new\_variables.pca)

## List of 5  
## $ sdev : num [1:12] 3.119 1.457 0.322 0.178 0.096 ...  
## $ rotation: num [1:12, 1:12] -0.319 -0.316 -0.309 -0.32 0.317 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:12] "mean\_speed" "mean\_speed\_2" "mean\_speed\_3" "log\_speed" ...  
## .. ..$ : chr [1:12] "PC1" "PC2" "PC3" "PC4" ...  
## $ center : Named num [1:12] 7.96e+01 6.48e+03 5.40e+05 4.36 1.29e-02 ...  
## ..- attr(\*, "names")= chr [1:12] "mean\_speed" "mean\_speed\_2" "mean\_speed\_3" "log\_speed" ...  
## $ scale : Named num [1:12] 1.26e+01 1.98e+03 2.41e+05 1.65e-01 2.22e-03 ...  
## ..- attr(\*, "names")= chr [1:12] "mean\_speed" "mean\_speed\_2" "mean\_speed\_3" "log\_speed" ...  
## $ x : num [1:89, 1:12] -4.29 -1.21 3.09 3.13 3.13 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : NULL  
## .. ..$ : chr [1:12] "PC1" "PC2" "PC3" "PC4" ...  
## - attr(\*, "class")= chr "prcomp"

## Using PCA1 and PCA2 as variables in regression

summary(flow\_pca\_prediction <-   
 lm(formula = new\_variables$flow ~   
 new\_variables.pca$x[,1] +   
 new\_variables.pca$x[,2]))

##   
## Call:  
## lm(formula = new\_variables$flow ~ new\_variables.pca$x[, 1] +   
## new\_variables.pca$x[, 2])  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.6667 -1.6600 -0.8846 1.3417 5.6351   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 44.1049 0.2357 187.120 <2e-16 \*\*\*  
## new\_variables.pca$x[, 1] 0.1859 0.0760 2.446 0.0165 \*   
## new\_variables.pca$x[, 2] 6.6144 0.1627 40.653 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 2.224 on 86 degrees of freedom  
## Multiple R-squared: 0.9507, Adjusted R-squared: 0.9496   
## F-statistic: 829.3 on 2 and 86 DF, p-value: < 2.2e-16

summary(flow\_pca\_prediction <-   
 lm(formula = new\_variables$flow ~   
 0 + poly(new\_variables.pca$x[,1],2, raw = T) +  
 poly(new\_variables.pca$x[,2],2, raw = T) ))

##   
## Call:  
## lm(formula = new\_variables$flow ~ 0 + poly(new\_variables.pca$x[,   
## 1], 2, raw = T) + poly(new\_variables.pca$x[, 2], 2, raw = T))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -76.184 -3.353 11.876 21.885 36.574   
##   
## Coefficients:  
## Estimate Std. Error t value  
## poly(new\_variables.pca$x[, 1], 2, raw = T)1 -3.4658 0.7751 -4.471  
## poly(new\_variables.pca$x[, 1], 2, raw = T)2 2.9928 0.2713 11.031  
## poly(new\_variables.pca$x[, 2], 2, raw = T)1 30.2656 1.9991 15.139  
## poly(new\_variables.pca$x[, 2], 2, raw = T)2 2.6446 0.9588 2.758  
## Pr(>|t|)   
## poly(new\_variables.pca$x[, 1], 2, raw = T)1 2.39e-05 \*\*\*  
## poly(new\_variables.pca$x[, 1], 2, raw = T)2 < 2e-16 \*\*\*  
## poly(new\_variables.pca$x[, 2], 2, raw = T)1 < 2e-16 \*\*\*  
## poly(new\_variables.pca$x[, 2], 2, raw = T)2 0.00711 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 20.58 on 85 degrees of freedom  
## Multiple R-squared: 0.8019, Adjusted R-squared: 0.7925   
## F-statistic: 86 on 4 and 85 DF, p-value: < 2.2e-16

summary(flow\_pca\_prediction <-   
 lm(formula = new\_variables$flow ~   
 poly(new\_variables.pca$x[,1],2, raw = T) +  
 poly(new\_variables.pca$x[,2],2, raw = T) ))

##   
## Call:  
## lm(formula = new\_variables$flow ~ poly(new\_variables.pca$x[,   
## 1], 2, raw = T) + poly(new\_variables.pca$x[, 2], 2, raw = T))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.09096 -0.29027 0.05971 0.17772 0.99466   
##   
## Coefficients:  
## Estimate Std. Error t value  
## (Intercept) 41.396920 0.086310 479.633  
## poly(new\_variables.pca$x[, 1], 2, raw = T)1 0.057774 0.016610 3.478  
## poly(new\_variables.pca$x[, 1], 2, raw = T)2 0.101120 0.007971 12.686  
## poly(new\_variables.pca$x[, 2], 2, raw = T)1 8.559631 0.059365 144.186  
## poly(new\_variables.pca$x[, 2], 2, raw = T)2 0.826906 0.018812 43.957  
## Pr(>|t|)   
## (Intercept) < 2e-16 \*\*\*  
## poly(new\_variables.pca$x[, 1], 2, raw = T)1 0.000802 \*\*\*  
## poly(new\_variables.pca$x[, 1], 2, raw = T)2 < 2e-16 \*\*\*  
## poly(new\_variables.pca$x[, 2], 2, raw = T)1 < 2e-16 \*\*\*  
## poly(new\_variables.pca$x[, 2], 2, raw = T)2 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3956 on 84 degrees of freedom  
## Multiple R-squared: 0.9985, Adjusted R-squared: 0.9984   
## F-statistic: 1.376e+04 on 4 and 84 DF, p-value: < 2.2e-16