Lab 2

Sahil Jain

September 23, 2017

Reading the csv file into R console

```
BikeShare <- read.csv("/Users/sahiljain/Downloads/bike_share.csv")
```

Initializing variables.

```
y <- BikeShare$count
x1 <- BikeShare$temp
x2 <- BikeShare$humidity
x3 <- BikeShare$windspeed
x4 <- BikeShare$season
x5 <- BikeShare$weather
```

A. Fit a simple linear regression model relating count to temp. Formally test Beta_1 = 0 and beta_1 != 0.

Linear model of count to temp.

```
model1 <- lm(y ~ x1)
s1 <- summary(model1)
s1
```

```
##
## Call:
## lm(formula = y ~ x1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -293.32 -112.36  -33.36   78.98  741.44
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -156.9856     7.9451  -19.76   <2e-16 ***
## x1             5.0947     0.1138   44.78   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 166.5 on 10884 degrees of freedom
## Multiple R-squared:  0.1556, Adjusted R-squared:  0.1555
## F-statistic:  2006 on 1 and 10884 DF,  p-value: < 2.2e-16
```

Calculating the parameters manually.

```
beta1a_hat <- cor(x1,y) * sd(y) / sd(x1)
beta1a_hat
```

```
## [1] 5.094745
```

```
beta0a_hat <- mean(y) - beta1a_hat * mean(x1)
beta0a_hat
```

```
## [1] -156.9856
```

H0 : beta_1 = 0 versus Ha : beta_1 != 0

```
se_beta1a <- s1$coefficients[2,2]
t <- beta1a_hat / se_beta1a
p_val1 <- 2*pt(q = abs(t), df = 10884, lower.tail = FALSE)
print(paste("The p-value associated with H0 in count vs temprature : ", p_val1, sep =
""))
```

```
## [1] "The p-value associated with H0 in count vs temprature : 0"
```

95% Confidence interval for beta1

```
critc_val <- qt(p = 0.975, df = 10884, lower.tail = TRUE)

low_CI <- beta1a_hat - critc_val * se_beta1a
upp_CI <- beta1a_hat + critc_val * se_beta1a

print(paste("The 95% confidence interval for beta1 is : ", low_CI, ",", upp_CI, sep =
""))
```

```
## [1] "The 95% confidence interval for beta1 is : 4.87174499335949,5.31774443044745"
```

Interpretaion : From the p-value which is 0, we will not reject the null hypothesis and the level of confidence in 99.5%. This means that variable windspeed is highly significant and bike rentals are significantly influenced by the temprature.

B. Fit a simple linear regression model relating count to humidity. Formally test Beta_1 = 0 and beta_1 != 0.

Linear model of count to humidity

```
model2 <- lm(y ~ x2)
s2 <- summary(model2)
s2
```

```
##
## Call:
## lm(formula = y ~ x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -375.45 -120.49  -41.86   82.15  734.73
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 376.44561    5.54494   67.89   <2e-16 ***
## x2           -2.98727    0.08556  -34.91   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 171.8 on 10884 degrees of freedom
## Multiple R-squared:  0.1007, Adjusted R-squared:  0.1006
## F-statistic:  1219 on 1 and 10884 DF,  p-value: < 2.2e-16
```

Calculating the parameters manually.

```
beta1b_hat <- cor(x2,y) * sd(y) / sd(x2)
beta1b_hat
```

```
## [1] -2.987269
```

```
beta0b_hat <- mean(y) - beta1b_hat * mean(x2)
beta0b_hat
```

```
## [1] 376.4456
```

H0 : beta_1 = 0 versus Ha : beta_1 != 0

```
se_beta1b <- s2$coefficients[2,2]
t <- beta1b_hat / se_beta1b
p_val2 <- 2*pt(q = abs(t), df = 10884, lower.tail = FALSE)
print(paste("The p-value associated with Ho: beta0 = 0 is ", p_val2, sep = ""))
```

```
## [1] "The p-value associated with Ho: beta0 = 0 is 2.9215416637178e-253"
```

95% Confidece interval of beta1

```
critc_val <- qt(p = 0.975, df = 10884, lower.tail = TRUE)

low_CI <- beta1b_hat - critc_val * se_beta1b
upp_CI <- beta1b_hat + critc_val * se_beta1b

print(paste("The 95% confidence interval for beta1 is : ", low_CI, ",", upp_CI, sep =
""))
```

```
## [1] "The 95% confidence interval for beta1 is : -3.15497698856335,-2.8195601685055
1"
```

Interpretation : From the p-value which is 2.92 * 10^-253 < 0, we will not reject the null hypothesis and the level of confidence in 99.5%. This means that variable humidity is highly significant and bike rentals are significantly influenced by the humidity.

C. Fit a simple linear regression model relating count to Windspeed. Formally test Beta_1 = 0 and beta_1 != 0.

Linear model between count vs windspeed.

```
model3 <- lm(y ~ x3)
s3 <- summary(model3)
s3
```

```
##
## Call:
## lm(formula = y ~ x3)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -274.74 -145.29  -48.53   92.48  807.21
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 162.7876     3.2120   50.68   <2e-16 ***
## x3            2.2491     0.2116   10.63   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 180.2 on 10884 degrees of freedom
## Multiple R-squared:  0.01028,    Adjusted R-squared:  0.01018
## F-statistic:   113 on 1 and 10884 DF,  p-value: < 2.2e-16
```

Calculating the parameters manually

```
beta1c_hat <- cor(x3,y) * sd(y) / sd(x3)
beta1c_hat
```

```
## [1] 2.249058
```

```
beta0c_hat <- mean(y) - beta1c_hat * mean(x3)
beta0c_hat
```

```
## [1] 162.7876
```

H0 : beta_1 = 0 versus Ha : beta_1 != 0

```
se_beta1c <- s3$coefficients[2,2]
t <- beta1c_hat / se_beta1c
p_val3 <- 2*pt(q = abs(t), df = 10884, lower.tail = FALSE)
print(paste("The p-value associated with Ho: beta0 = 0 is ", p_val3, sep = ""))
```

```
## [1] "The p-value associated with Ho: beta0 = 0 is 2.89840720315406e-26"
```

95% Confidece interval of beta1

```
critc_val <- qt(p = 0.975, df = 10884, lower.tail = TRUE)

low_CI <- beta1c_hat - critc_val * se_beta1c
upp_CI <- beta1c_hat + critc_val * se_beta1c

print(paste("The 95% confidence interval for beta1 is : ", low_CI, ",", upp_CI, sep =
""))
```

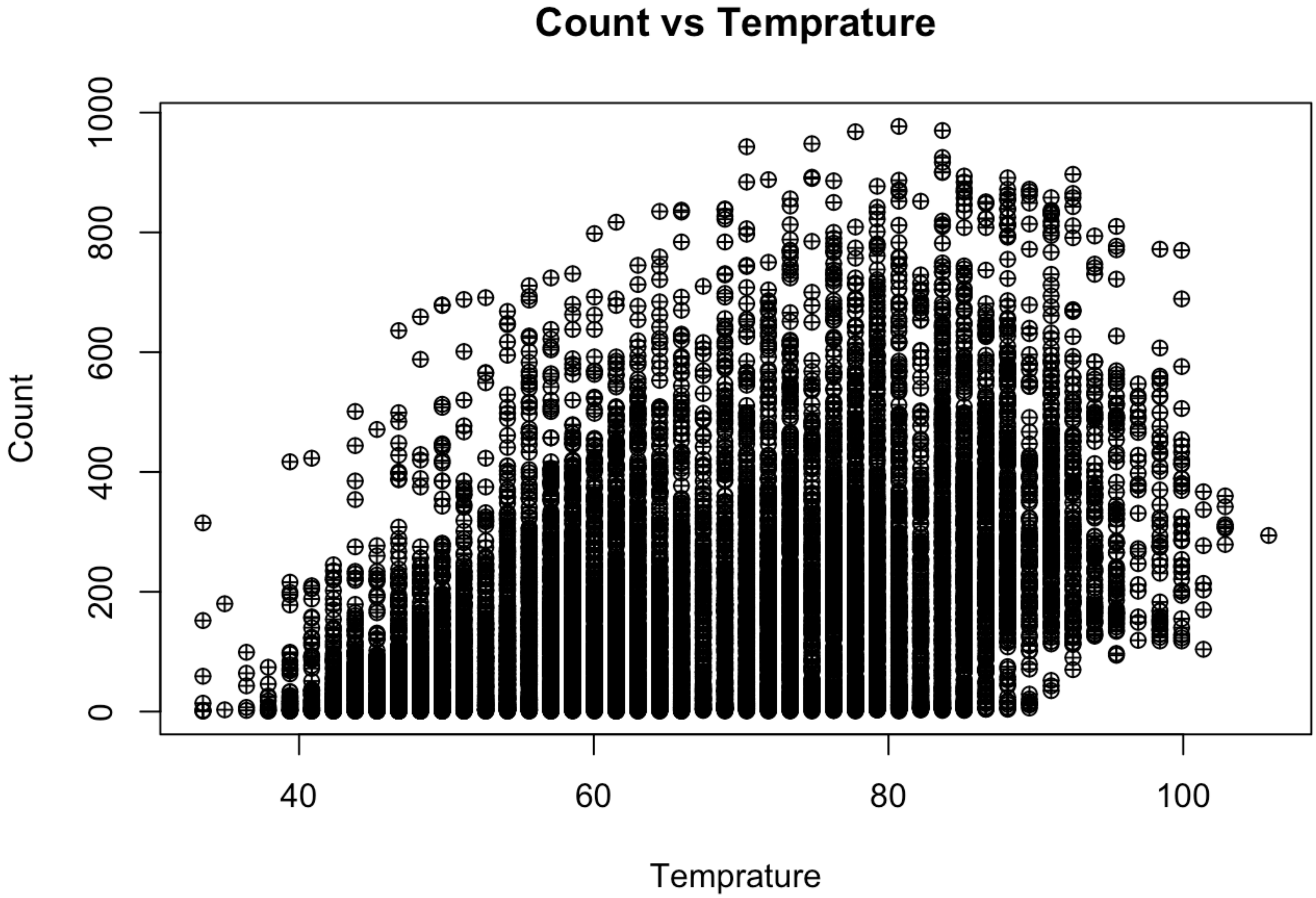
```
## [1] "The 95% confidence interval for beta1 is : 1.83434010656766,2.66377572810539"
```

Interpretation : From the p-value which is 2.89 * 10^-26 < 0, we will not reject the null hypothesis and the level of confidence in 99.5%. This means that variable windspeed is highly significant and bike rentals are significantly influenced by the Windspeed.

D. Construct three Scatter plots : (1) Count vs Temp (2) Count vs Humidity and (3) Count vs Windspeed. On all of these, plot the least squares line-of-best-fit, the 95% confindense interval and the 95% prediction interval.

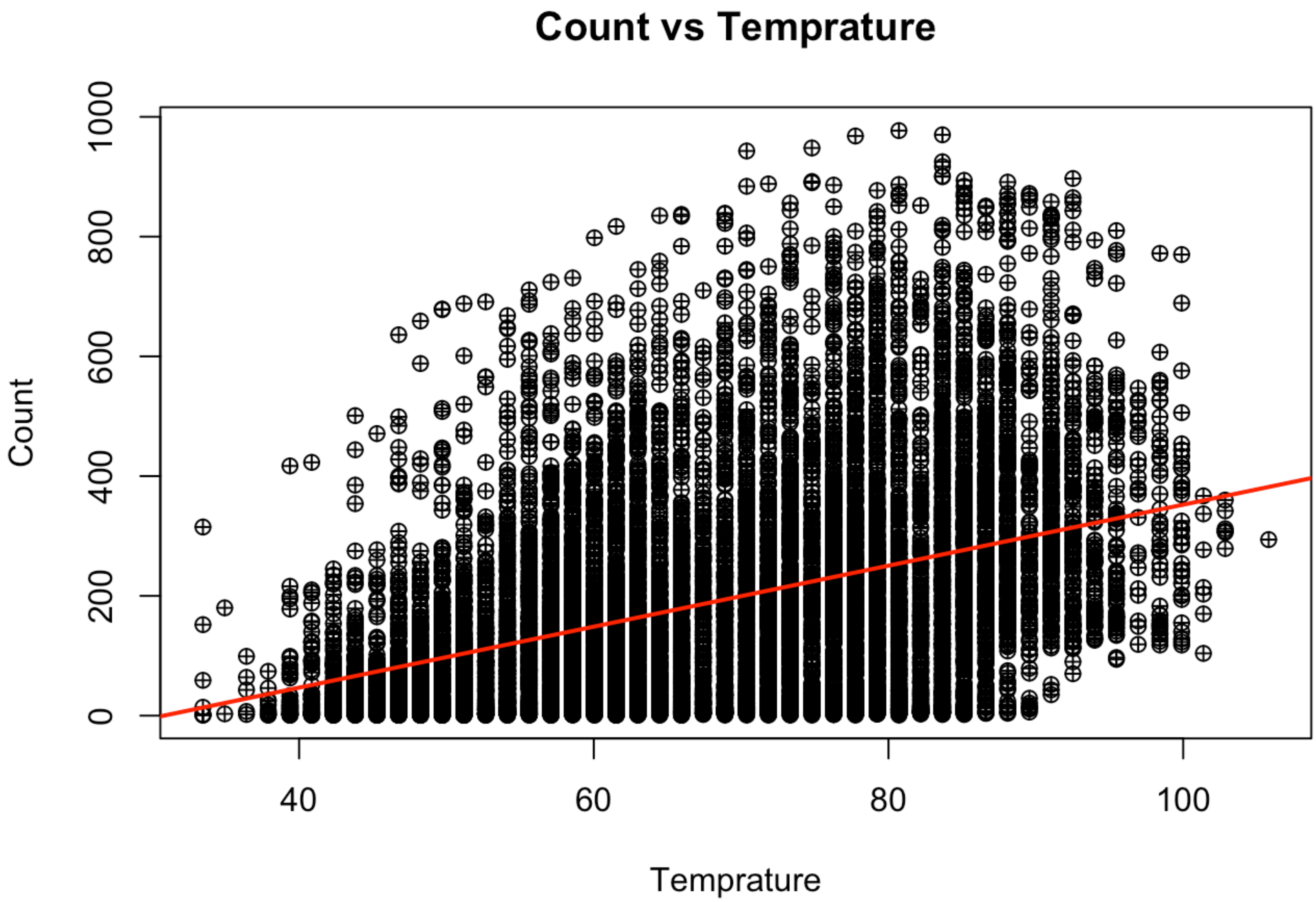
1(a) Scatter plot of count vs temp.

```
plot(x1, y, ylab = "Count", xlab = "Temprature", main = "Count vs Temprature", pch =
10)
```



1(b) Scatter plot of count vs temp with line of best fit

```
plot(x1, y, ylab = "Count", xlab = "Temprature", main = "Count vs Temprature", pch =
10)
abline(model1, col = "red", lwd = 2)
```



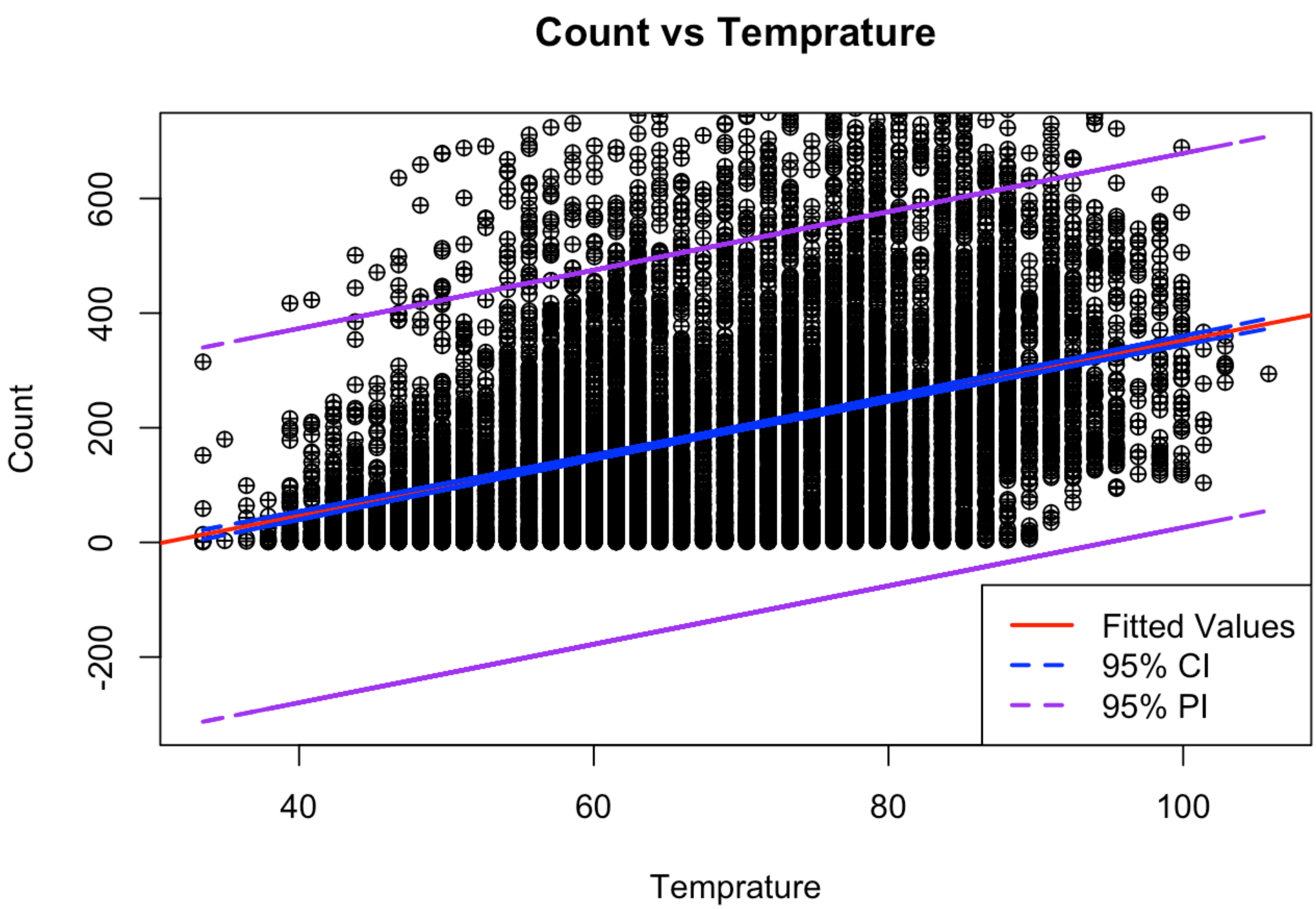
1(c) 95% Confidence interval and 95% prediction interval and plotting them on them scatter plot.

Count vs Temp

```
xp <- seq(from = min(x1), to = max(x1), length.out = 100)
CI <- predict(lm(y~x1), newData = data.frame(x1 = xp), interval = "confidence", level
= 0.95)
PI <- predict(lm(y~x1), newData = data.frame(x1 = xp), interval = "prediction", level
= 0.95)
```

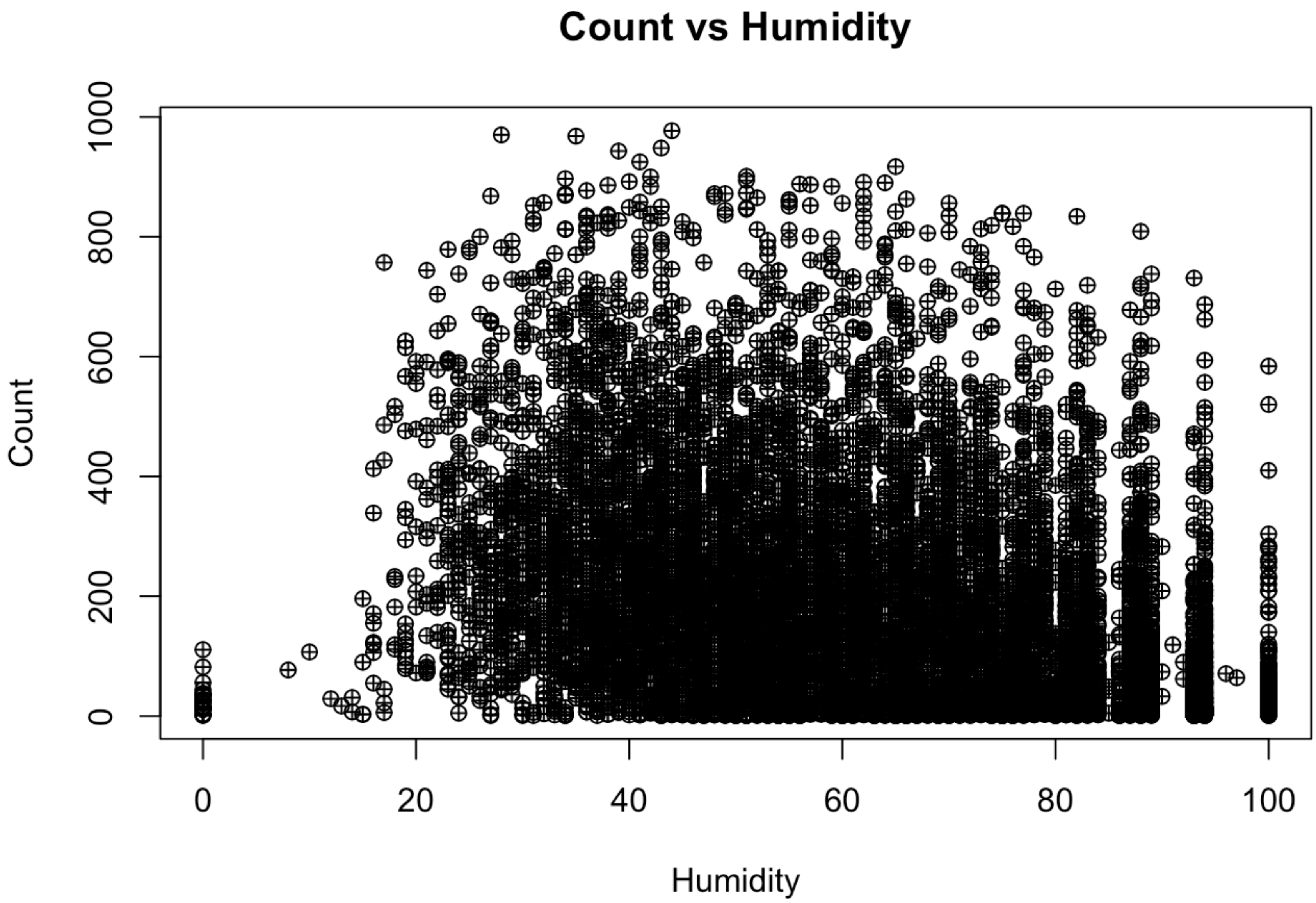
```
## Warning in predict.lm(lm(y ~ x1), newData = data.frame(x1 = xp), interval = "predi
ction", : predictions on current data refer to _future_ responses
```

```
ci_low <- CI[,2]
ci_hi <- CI[,3]
pi_low <- PI[,2]
pi_hi <- PI[,3]
plot(x1,y, ylab = "Count", xlab = "Temprature", main = "Count vs Temprature", pch = 1
0, ylim = c(min(pi_low), max(pi_hi)))
abline(lm(y~x1), col = "red", lwd = 2)
lines(x = x1, y = ci_low, col = "blue", lty = 2, lwd = 2)
lines(x = x1, y = ci_hi, col = "blue", lty = 2, lwd = 2)
lines(x = x1, y = pi_low, col = "purple", lty = 2, lwd = 2)
lines(x = x1, y = pi_hi, col = "purple", lty = 2, lwd = 2)
legend("bottomright", legend = c("Fitted Values", "95% CI", "95% PI"), lwd = c(2,2,2)
, lty = c(1,2,2), col = c("red", "blue", "purple"))
```



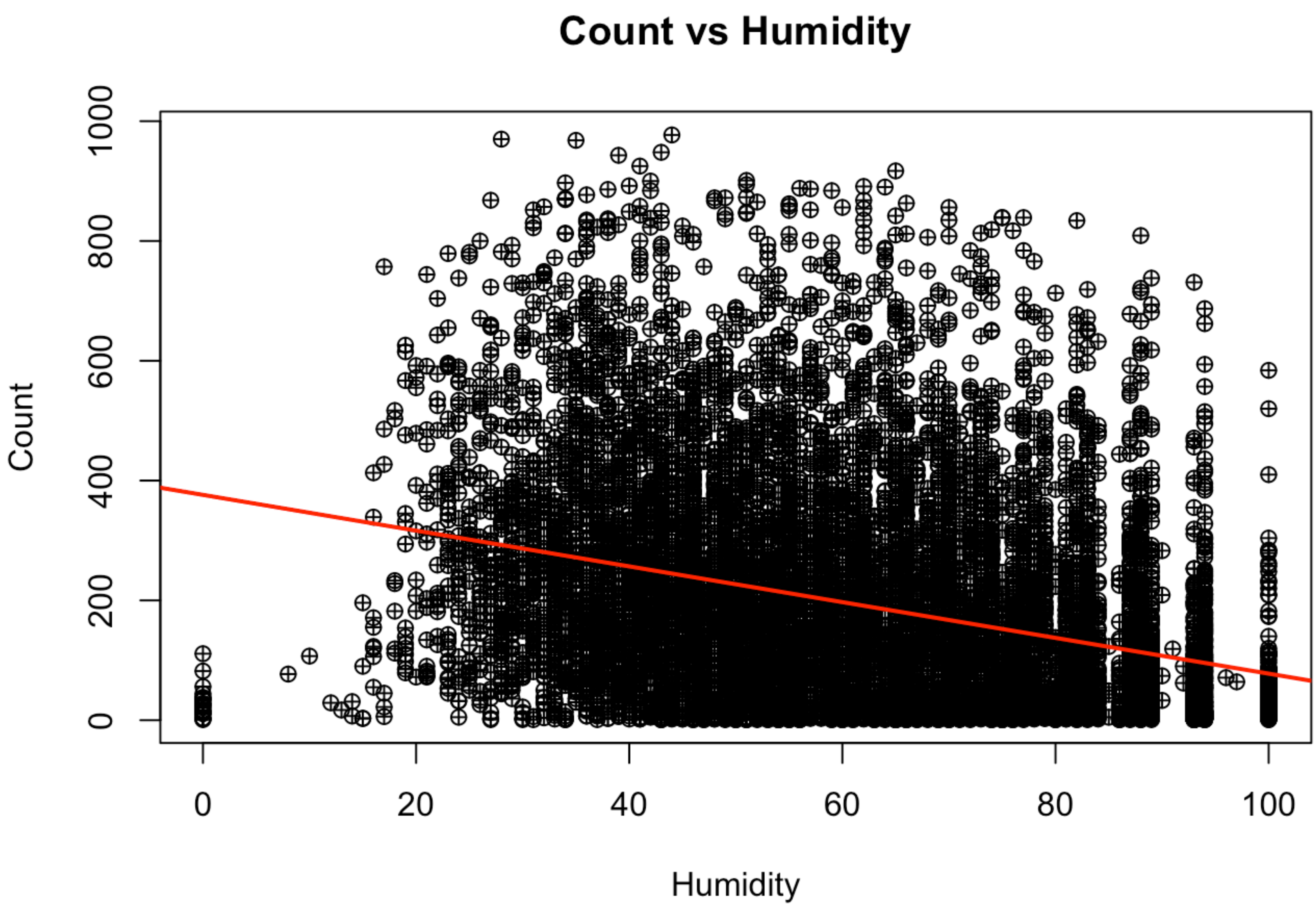
2(a) Scatter plot of Count vs Humidity

```
plot(x2,y, xlab = "Humidity" , ylab = "Count", main = "Count vs Humidity", pch = 10)
```



2(b) Line of the best fit

```
plot(x2,y, xlab = "Humidity" , ylab = "Count", main = "Count vs Humidity", pch = 10)
abline(model2, col = "red", lwd = 2)
```



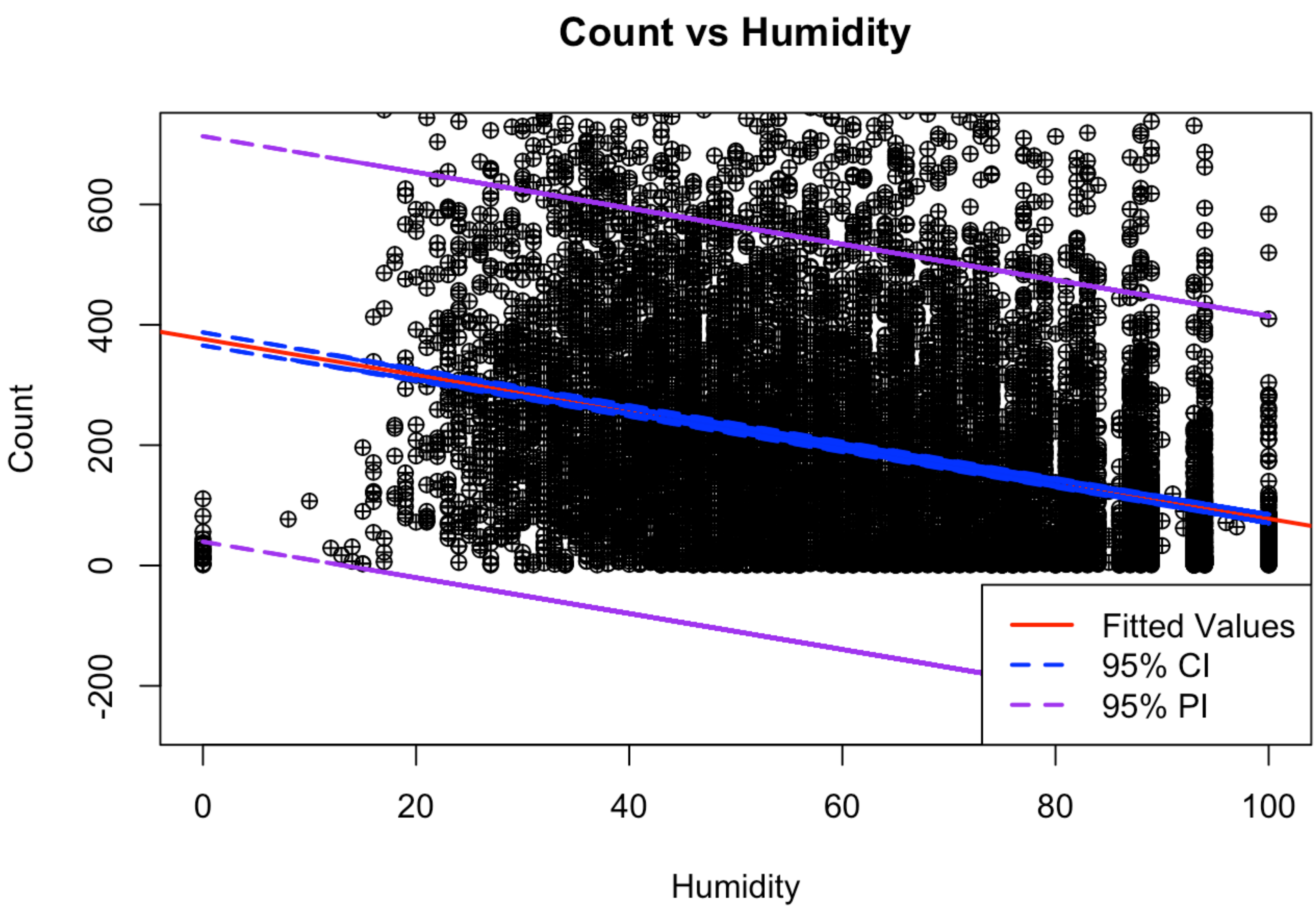
2(c) 95% Confidence and prediction interval for Count vs Humidity

Count vs Humidity

```
xp <- seq(from = min(x2), to = max(x2), length.out = 100)
CI <- predict(lm(y~x2), newData = data.frame(x2 = xp), interval = "confidence", level
= 0.95)
PI <- predict(lm(y~x2), newData = data.frame(x2 = xp), interval = "prediction", level
= 0.95)
```

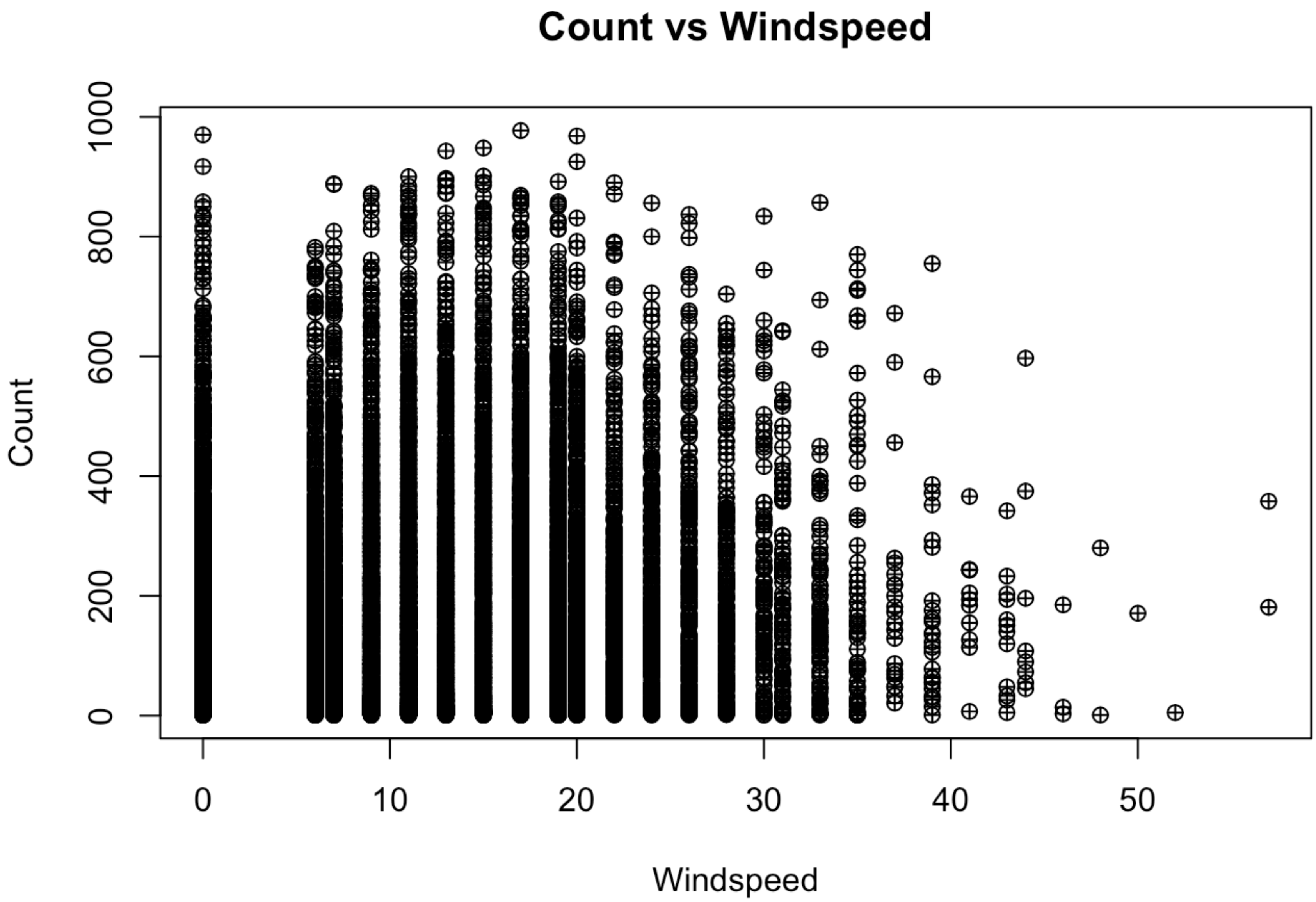
```
## Warning in predict.lm(lm(y ~ x2), newData = data.frame(x2 = xp), interval = "predi
ction", : predictions on current data refer to _future_ responses
```

```
ci_low <- CI[,2]
ci_hi <- CI[,3]
pi_low <- PI[,2]
pi_hi <- PI[,3]
plot(x2,y, ylab = "Count", xlab = "Humidity", main = "Count vs Humidity", pch = 10, y
lim = c(min(pi_low), max(pi_hi)))
abline(lm(y~x2), col = "red", lwd = 2)
lines(x = x2, y = ci_low, col = "blue", lty = 2, lwd = 2)
lines(x = x2, y = ci_hi, col = "blue", lty = 2, lwd = 2)
lines(x = x2, y = pi_low, col = "purple", lty = 2, lwd = 2)
lines(x = x2, y = pi_hi, col = "purple", lty = 2, lwd = 2)
legend("bottomright", legend = c("Fitted Values", "95% CI", "95% PI"), lwd = c(2,2,2)
, lty = c(1,2,2), col = c("red", "blue", "purple"))
```



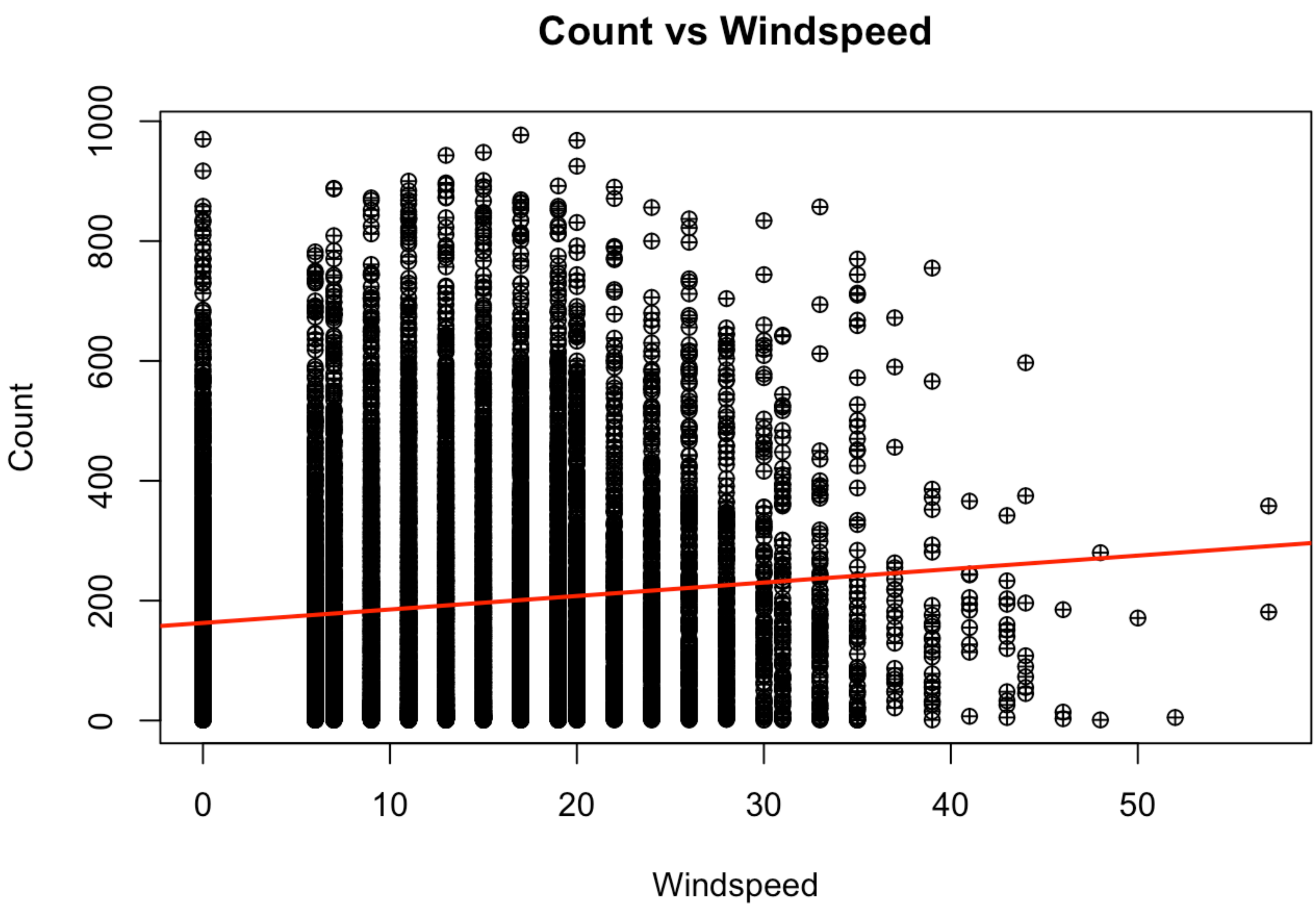
3(a) Scatter Plot for Count vs Windspeed

```
plot(x3,y, ylab = "Count", xlab = "Windspeed", main = "Count vs Windspeed", pch = 10)
```



3(b) Line of the best fit

```
plot(x3,y, ylab = "Count", xlab = "Windspeed", main = "Count vs Windspeed", pch = 10)
abline(model3, col = "red", lwd = 2)
```



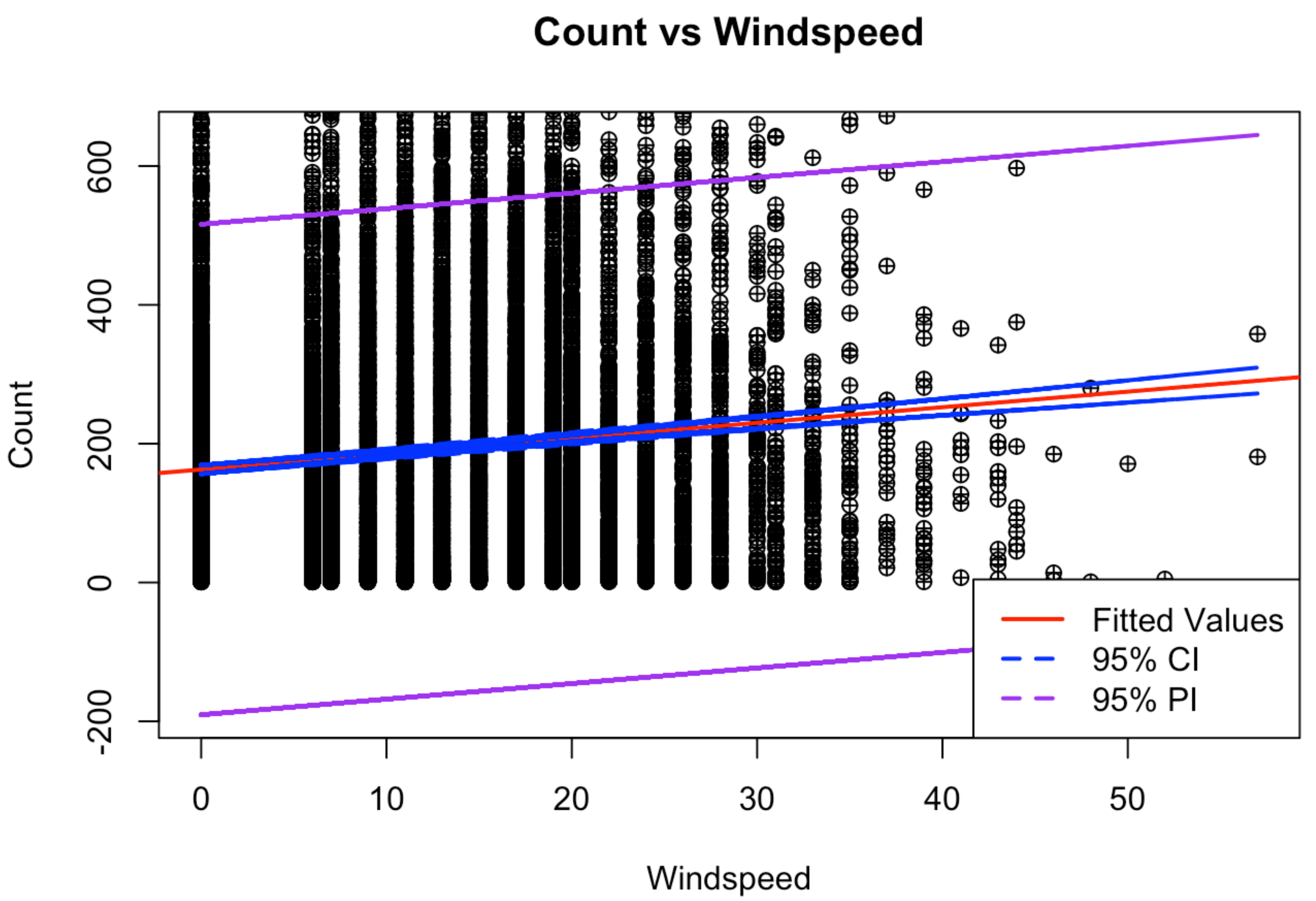
3(c) 95% Confidence and prediction interval

Count vs Windspeed

```
xp <- seq(from = min(x3), to = max(x3), length.out = 100)
CI <- predict(lm(y~x3), newData = data.frame(x3 = xp), interval = "confidence", level
= 0.95)
PI <- predict(lm(y~x3), newData = data.frame(x3 = xp), interval = "prediction", level
= 0.95)
```

```
## Warning in predict.lm(lm(y ~ x3), newData = data.frame(x3 = xp), interval = "predi
ction", : predictions on current data refer to _future_ responses
```

```
ci_low <- CI[,2]
ci_hi <- CI[,3]
pi_low <- PI[,2]
pi_hi <- PI[,3]
plot(x3,y, ylab = "Count", xlab = "Windspeed", main = "Count vs Windspeed", pch = 10,
ylim = c(min(pi_low), max(pi_hi)))
abline(lm(y~x3), col = "red", lwd = 2)
lines(x = x3, y = ci_low, col = "blue", lty = 2, lwd = 2)
lines(x = x3, y = ci_hi, col = "blue", lty = 2, lwd = 2)
lines(x = x3, y = pi_low, col = "purple", lty = 2, lwd = 2)
lines(x = x3, y = pi_hi, col = "purple", lty = 2, lwd = 2)
legend("bottomright", legend = c("Fitted Values", "95% CI", "95% PI"), lwd = c(2,2,2)
, lty = c(1,2,2), col = c("red", "blue", "purple"))
```



E. Using you're results form part (d) predict the number of bike rentals in hours for which

i. The outside temprature is 80 degrees fahrenheit
ii. The wind speed in 15 mph
iii. The relative humidity is 100%

iv. when outside temprature is 80 degrees

```
Xi1 = 80
yp_hat1 <- beta0a_hat + beta1a_hat * Xi1
yp_hat1
```

```
## [1] 250.594
```

```
predict(object = model1, newdata = data.frame(x1 = 80), interval = "prediction", leve
l = 0.95)
```

```
##       fit       lwr      upr
## 1 250.594 -75.73129 576.9192
```

ii. When wind speed is 15 mph

```
Xi2 <- 15
yp_hat <- beta0c_hat + beta1c_hat * Xi2
yp_hat
```

```
## [1] 196.5234
```

```
predict(object = model3, newdata = data.frame(x3 = 15), interval = "prediction", leve
l = 0.95)
```

```
##        fit       lwr      upr
## 1 196.5234 -156.7573 549.8041
```

iii. When relative humidity is 100%

```
Xi3 <- 100
yp_hat <- beta0b_hat + beta1b_hat * Xi3
yp_hat
```

```
## [1] 77.71875
```

```
predict(object = model2, newdata = data.frame(x2 = 100), interval = "prediction", lev
el = 0.95)
```

```
##        fit      lwr      upr
## 1 77.71875 -259.092 414.5295
```

F. Fit a linear regression model relation count to season using automated functions.

```
model_season <- lm(y ~ factor(x4), data = BikeShare)
summary(model_season)
```

```
##
## Call:
## lm(formula = y ~ factor(x4), data = BikeShare)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -233.42 -115.99  -38.99   87.58  749.01
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  116.343      3.387   34.35   <2e-16 ***
## factor(x4)2   98.908      4.769   20.74   <2e-16 ***
## factor(x4)3  118.074      4.769   24.76   <2e-16 ***
## factor(x4)4   82.645      4.769   17.33   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 175.5 on 10882 degrees of freedom
## Multiple R-squared:  0.06132,    Adjusted R-squared:  0.06106
## F-statistic: 236.9 on 3 and 10882 DF,  p-value: < 2.2e-16
```

From the model we can see that all of the season's catergorical variables are highly significant. Regression equation will look something like this : Y = 116.343 + 98.908X1 + 118.074X2 + 82.645X3, where Beta0 is 116.343 and beta1 = 98.908, beta2 = 118.075, beta3 = 82.645. When it'll be spring season there will 116.343 rentals per day/hour where as number of rental increases in fall season, declines in summer and winter. Expected value in all of the seasons will looks as follows :

1. Y_spring = Beta0 = 116.343
2. Y_summer = Beta0 + Beta2X2 = 116.343 + 98.908*X2
3. Y_fall = Beta0 + Beta3X3 = 116.343 + 118.074*X3
4. Y_winter = Beta0 + Beta4X4 = 116.343 + 82.645*X4

G. Fit a linear regression model relation count to weather using automated functions.

```
model_Weather <- lm(y ~ factor(x5), data = BikeShare)
summary(model_Weather)
```

```
##
## Call:
## lm(formula = y ~ factor(x5), data = BikeShare)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -204.24 -142.24  -44.90   90.76  772.15
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  205.237      2.117  96.936  < 2e-16 ***
## factor(x5)2  -26.281      3.982  -6.599 4.32e-11 ***
## factor(x5)3  -86.390      6.482 -13.328  < 2e-16 ***
## factor(x5)4  -41.237    179.567  -0.230    0.818
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 179.6 on 10882 degrees of freedom
## Multiple R-squared:  0.01775,    Adjusted R-squared:  0.01747
## F-statistic: 65.53 on 3 and 10882 DF,  p-value: < 2.2e-16
```

From the model we can see that most of the variables are highly significant apart from 4 which is stormy. Regression equation will look something like this : Y = 205.237 - 26.281X1 - 86.390X2 - 41.237X3, where Beta0 is 205.237 and beta1 = -26.281, beta2 = -86.390, beta3 = -41.237. When it'll be nice/sunny weather there will 205.237 rentals per day/hour where as number of rental decreases in cloudy weather, declines even more in stormy and rainy. Expected value in all of the weather's will looks as follows :

1. Y_spring = Beta0 = 205.237
2. Y_summer = Beta0 + Beta2X2 = 205.237 - 26.281*X2
3. Y_fall = Beta0 + Beta3X3 = 205.237 - 86.390*X3
4. Y_winter = Beta0 + Beta4X4 = 205.237 - 41.237*X4