TCS

timesheet <- rio::import("Assignment 2 Exhibits.xlsx", sheet = "Time", skip=5) %>%   
 as\_tibble() %>%   
 janitor::clean\_names()  
timesheet <- timesheet %>%   
 mutate(rpc\_rate = right\_party\_connect\_count / call\_count)

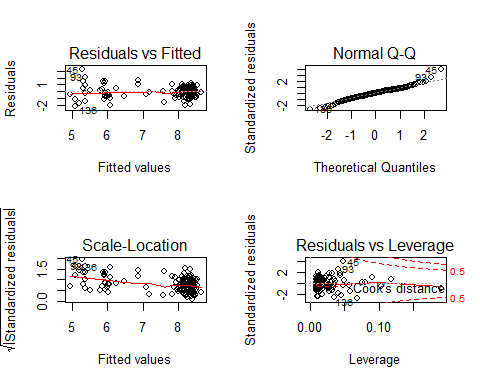
metrics <- rio::import("Q2 TCS\_Metrics\_FInal.xlsx", sheet = "TCS\_Metrics") %>%   
 as\_tibble() %>%   
 janitor::clean\_names()  
metrics <- metrics[,1:7]

df <- metrics %>%   
 left\_join(timesheet %>%   
 select(dialer\_id,work\_time\_in\_hours, rpc\_rate))  
df <- df %>% filter(total\_payment\_amount != 0 )

lm\_mod <- lm(log(total\_payment\_amount) ~ log(work\_time\_in\_hours) + rpc\_rate, data = df)  
summary(lm\_mod)

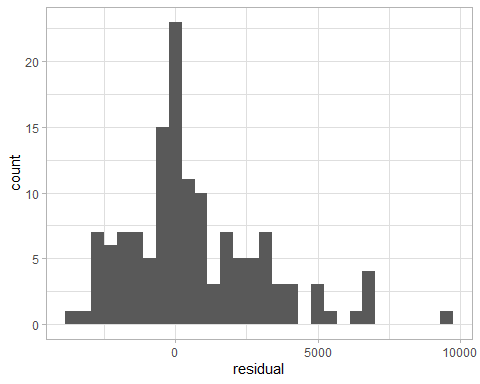
##   
## Call:  
## lm(formula = log(total\_payment\_amount) ~ log(work\_time\_in\_hours) +   
## rpc\_rate, data = df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2.1087 -0.4656 0.0607 0.4733 3.2158   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.72102 0.54393 6.841 2.6e-10 \*\*\*  
## log(work\_time\_in\_hours) 0.99660 0.07245 13.755 < 2e-16 \*\*\*  
## rpc\_rate 5.24963 4.69640 1.118 0.266   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.8302 on 133 degrees of freedom  
## Multiple R-squared: 0.6345, Adjusted R-squared: 0.629   
## F-statistic: 115.4 on 2 and 133 DF, p-value: < 2.2e-16

par(mfrow=c(2,2))  
plot(lm\_mod, ask = F)



par(mfrow=c(1,1))

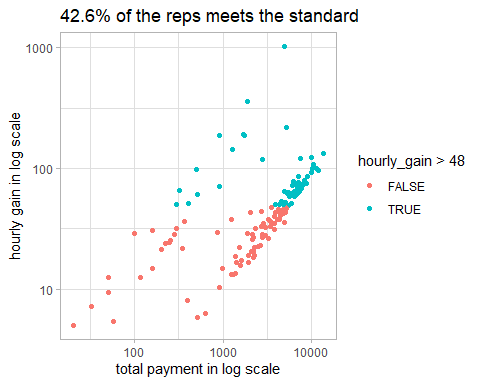
df <- df %>%   
 modelr::add\_predictions(lm\_mod) %>%   
 mutate(pred = pred %>% exp) %>%   
 mutate(residual = total\_payment\_amount - pred)  
  
df %>%   
 ggplot(aes(residual))+  
 geom\_histogram()



df <- df %>%   
 mutate(rpc\_adj\_total = total\_payment\_amount / exp(rpc\_rate \* 5.24963)) %>%   
 mutate(hourly\_gain = rpc\_adj\_total / work\_time\_in\_hours)   
  
df %>%   
 summarise(mean(hourly\_gain>60.3))

## # A tibble: 1 x 1  
## `mean(hourly\_gain > 60.3)`  
## <dbl>  
## 1 0.338

df %>%   
 ggplot(aes(total\_payment\_amount, hourly\_gain, color = hourly\_gain>48))+  
 geom\_point()+  
 scale\_x\_log10()+  
 scale\_y\_log10()+  
 labs(  
 title = "42.6% of the reps meets the standard",  
 x = "total payment in log scale",  
 y = "hourly gain in log scale"  
 )

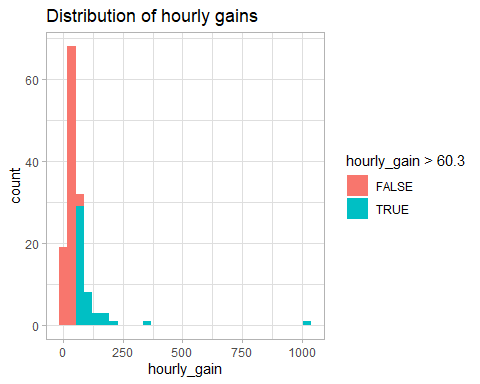


On average the performance is actually good.

df %>%   
 summarise(mean(hourly\_gain))

## # A tibble: 1 x 1  
## `mean(hourly\_gain)`  
## <dbl>  
## 1 60.3

df %>%   
 ggplot(aes(hourly\_gain, fill = hourly\_gain>60.3))+  
 geom\_histogram() +  
 labs(title = "Distribution of hourly gains")



t.test(df$hourly\_gain, mu = 60.3)

##   
## One Sample t-test  
##   
## data: df$hourly\_gain  
## t = -0.0021704, df = 135, p-value = 0.9983  
## alternative hypothesis: true mean is not equal to 60.3  
## 95 percent confidence interval:  
## 44.09682 76.46765  
## sample estimates:  
## mean of x   
## 60.28224

new\_ppl <- df %>%   
 mutate(new\_total = if\_else(hourly\_gain<60.3, 60.3\*work\_time\_in\_hours \* exp(rpc\_rate \* 5.24963), total\_payment\_amount))

new\_rev <- new\_ppl %>%   
 summarise(total = sum(new\_total)) %>%   
 pull(total)  
  
old\_rev <- df %>%   
 summarise(total = sum(total\_payment\_amount)) %>%   
 pull(total)  
  
new\_rev / old\_rev

## [1] 1.369115

new\_rev - old\_rev

## [1] 190616.1