

Ejercicio 3:

Apertura de archivos

```
getwd()
```

```
## [1] "D:/MAIN/CODING/R/WORKS/GENERA/INDIE"
```

```
GT = read.csv("TASK3.csv", sep=";")

lm_eqn <- function(df, y, x){
  m <- lm(y ~ x, df);
  eq <- substitute(italic(y) == a + b %.% italic(x)*", "~italic(r)^2~"="~r2,
                    list(a = format(unname(coef(m)[1]), digits = 2),
                          b = format(unname(coef(m)[2]), digits = 2),
                          r2 = format(summary(m)$r.squared, digits = 3)))
  as.character(as.expression(eq));
}
```

Análisis exploratorio

Premisas:

1. Los datos están extraídos de un juego tipo “saga” en el que no existen los farmeos.
2. La dificultad no es más que la media de intentos por nivel:
 - La dificultad se calculará pues, $DIF = completed / (fail + completed)$
3. El flujo de acciones dentro de una partida debe seguir el esquema:
4. Cada ‘ID Mission’ solo puede contener un missionStarted.
5. No contabilizan partidas con ayuda inicial.
- He observado que el valor del Booster se conserva desde el inicio hasta el final de la mission, lo que quiere decir, que si filtramos quitando todo Booster, estaríamos quitando las partidas con ayudas iniciales, de forma sencilla y directa
6. No contabilizan partidas tras uso de continues, solo hasta el momento de usarlo.
- Esta condición está directamente ligada a la condición 4), el razonamiento a seguir es sencillo:
 - la ID de misión es clave para nuestra limpieza y filtraje. Cada ID_mision solo puede contener un mission Started
 - Y una sola acción posterior: mCompleted, mFailed, mAbandoned(está no influirá en la dificultad), y como lo único que puede ocurrir dentro de la mission_id para que haya *más de dos filas* es el uso de continues, y éstos no se van a contar, pues sencillamente filtramos por grupo manteniendo sólo las 2 primeras filas

```
names(GT) = c("user_id", "action", "level_id", "mission_id", "B1", "B2", "B3", "leftover")

summary(GT)
```

```
##           user_id           action           level_id
## 49989f01f069: 376 missionAbandoned: 103 Min. :40.0
## e74b14927d1e: 327 missionCompleted: 6512 1st Qu.:44.0
## 7b521e79e19d: 325 missionFailed :13422 Median :49.0
## 14ca17d9503f: 302 missionStarted :20319 Mean :51.2
## e6b5582aa050: 301 useContinue : 607 3rd Qu.:58.0
## 156b7f92ff90: 296 Max. :70.0
## (Other) :39036
##           mission_id           B1           B2           B3
## 1ec6a2f64de6: 6 Min. :0.0000 Min. :0.0000 Min. :0.0000
## 20e4c403102c: 6 1st Qu.:0.0000 1st Qu.:0.0000 1st Qu.:0.0000
## 6194bc45854e: 6 Median :0.0000 Median :0.0000 Median :0.0000
## 675266cb1956: 6 Mean :0.0201 Mean :0.0326 Mean :0.0238
## 78aaead721c6: 6 3rd Qu.:0.0000 3rd Qu.:0.0000 3rd Qu.:0.0000
## 848b8c19e5a4: 6 Max. :1.0000 Max. :1.0000 Max. :1.0000
## (Other) :40927 NA's :607 NA's :607 NA's :607
##           leftover
## Min. : 0.00
## 1st Qu.: 1.00
## Median : 4.00
## Mean : 4.87
## 3rd Qu.: 7.00
## Max. :31.00
## NA's :34451
```

```
n_mission_id = as.integer(GT %>% distinct(mission_id, .keep_all = FALSE) %>% summarise(n=n(
)))
#20320
m_useContinue = as.integer(GT %>% filter(action=="useContinue") %>% distinct(mission_id, .k
eep_all = FALSE) %>% summarise(n=n()))
#596
```

Matriz que me indica la cantidad de completados o fallos tras el uso de continue. Esto nos ayudará a cotejar el resultado cuando vayamos a hacer el cálculo total

```
m_excess = GT %>% group_by(mission_id) %>%
```

```

mutate(m_cont=ifelse(action=="useContinue",1,0),
       check=sum(m_cont)) %>%
filter(check==1) %>% ##de esta forma filtramos los mission_id en los que se haya usado
group_by(mission_id) %>%
mutate(m_start=ifelse(action=="missionStarted",1,0),
       m_comp=ifelse(action=="missionCompleted",1,0),
       m_fail=ifelse(action=="missionFailed",1,0)) %>%
summarise(ST=sum(m_start),
          COM=sum(m_comp),
          FAI=sum(m_fail)) %>%
mutate(comp_excess=ifelse(COM-ST==0,1,0),
       fail_excess=ifelse(FAI-ST==1,1,0)) %>%
summarise(comp_excess=sum(comp_excess),fail_excess=sum(fail_excess))

# comp_excess fail_excess
# 431          145

```

#Ejercicio 1: Limpieza de tabla: ***

```

DT_cleaned = GT %>% replace_na(list(B1=0,B2=0,B3=0)) %>% ##Cambiamos los NA (en el futuro e
sto puede ser útil)
       filter_at(vars(c(B1,B2,B3)),all_vars(.==0)) %>% ###Cumplimos con la condicion
5, se eliminan 3232 registros
       group_by(mission_id) %>%
       mutate(start_counter=ifelse(action=="missionStarted",1,0), ###Cumplimos con la
condicion 4. Se eliminan 5 registros (es posible que provengan de una errata del proveedor?
)
       n_starts=sum(start_counter)) %>%
       filter(n_starts == 1) %>%
       select(-c(start_counter,n_starts)) %>%
       filter(row_number()==1 | row_number()==2) ##Cumplimos con la condición 6, basa
ndonos en la condición 3

```

#Ejercicio 2: Distribución de Dificultad por nivel: ***

```

Difficulty_wo_boosters = DT_cleaned %>%
group_by(level_id) %>%
mutate(fail=ifelse(action=="missionFailed",1,0),
       comp=ifelse(action=="missionCompleted",1,0)) %>%
summarise(DIFF = sum(comp)/(sum(comp)+sum(fail)))

```

Observamos la evolución de la dificultad según nivel:

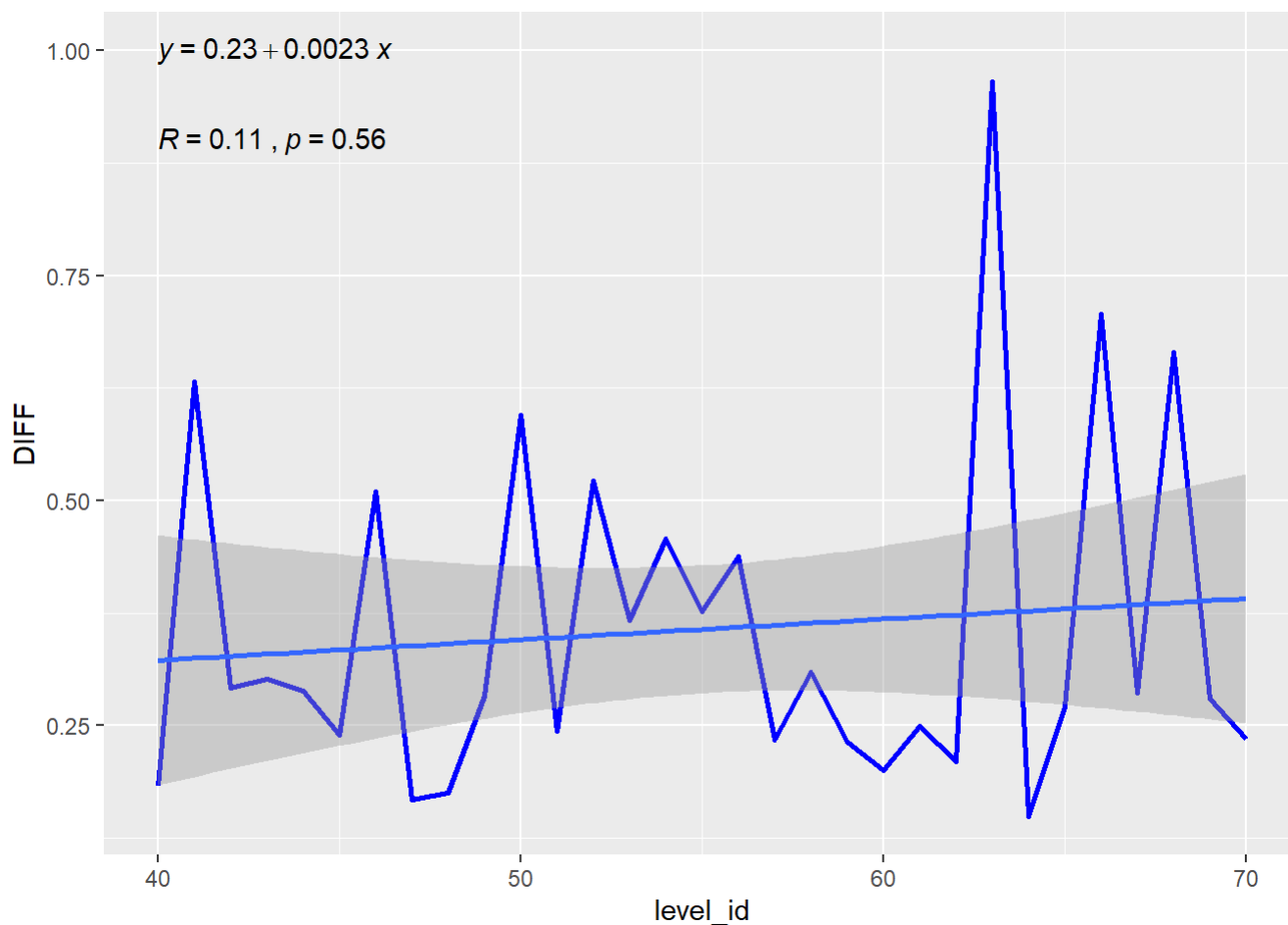
```

Difficulty_wo_boosters %>% ggplot(aes(x=level_id,y=DIFF))+
  geom_line(size=1,color="blue",
           add = "reg.line")+

  stat_cor(label.y = 0.9) +
  stat_regline_equation(label.y = 1)+
  geom_smooth(method='lm', formula= y~x)

```

```
## Warning: Ignoring unknown parameters: add
```



Puede constatarse a priori que no sigue una evolución con un sentido claro, excepto que aumenta de media por nivel.

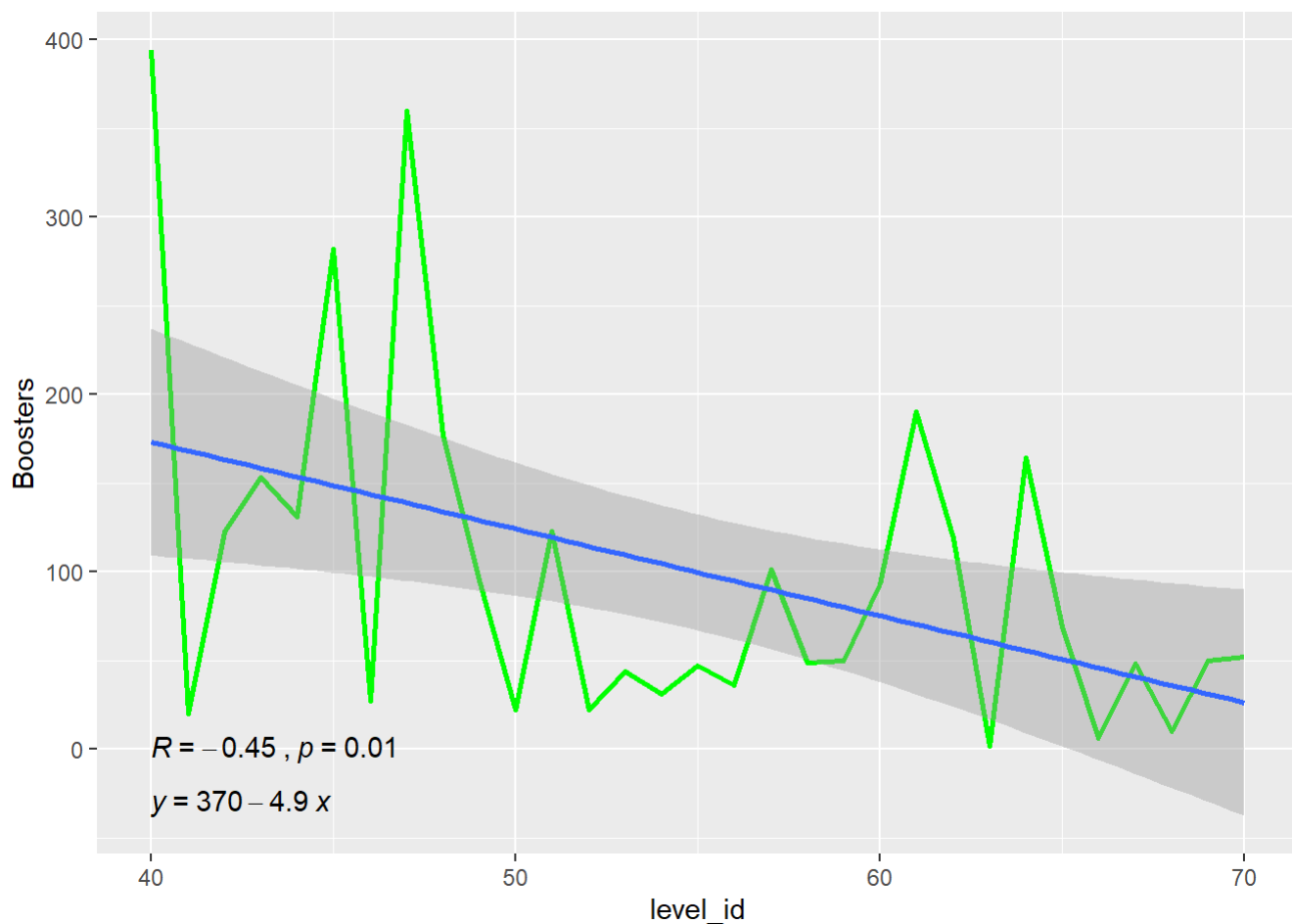
#Ejercicio 3: Uso de ayudas iniciales. Comparativas de partidas con y sin ayuda inicial ***

Visualizamos del uso de boosters:

```
Boosters = GT %>% group_by(level_id) %>%
  replace_na(list(B1=0,B2=0,B3=0)) %>%
  summarise(Boosters=sum(B1)+sum(B2)+sum(B3))
```

```
Boosters %>% ggplot(aes(x=level_id,y=Boosters))+
  geom_line(size=1,color="green",
    add = "reg.line")+
  stat_cor(label.y = 0) +
  stat_regline_equation(label.y = -30)+
  geom_smooth(method='lm', formula= y~x)
```

```
## Warning: Ignoring unknown parameters: add
```

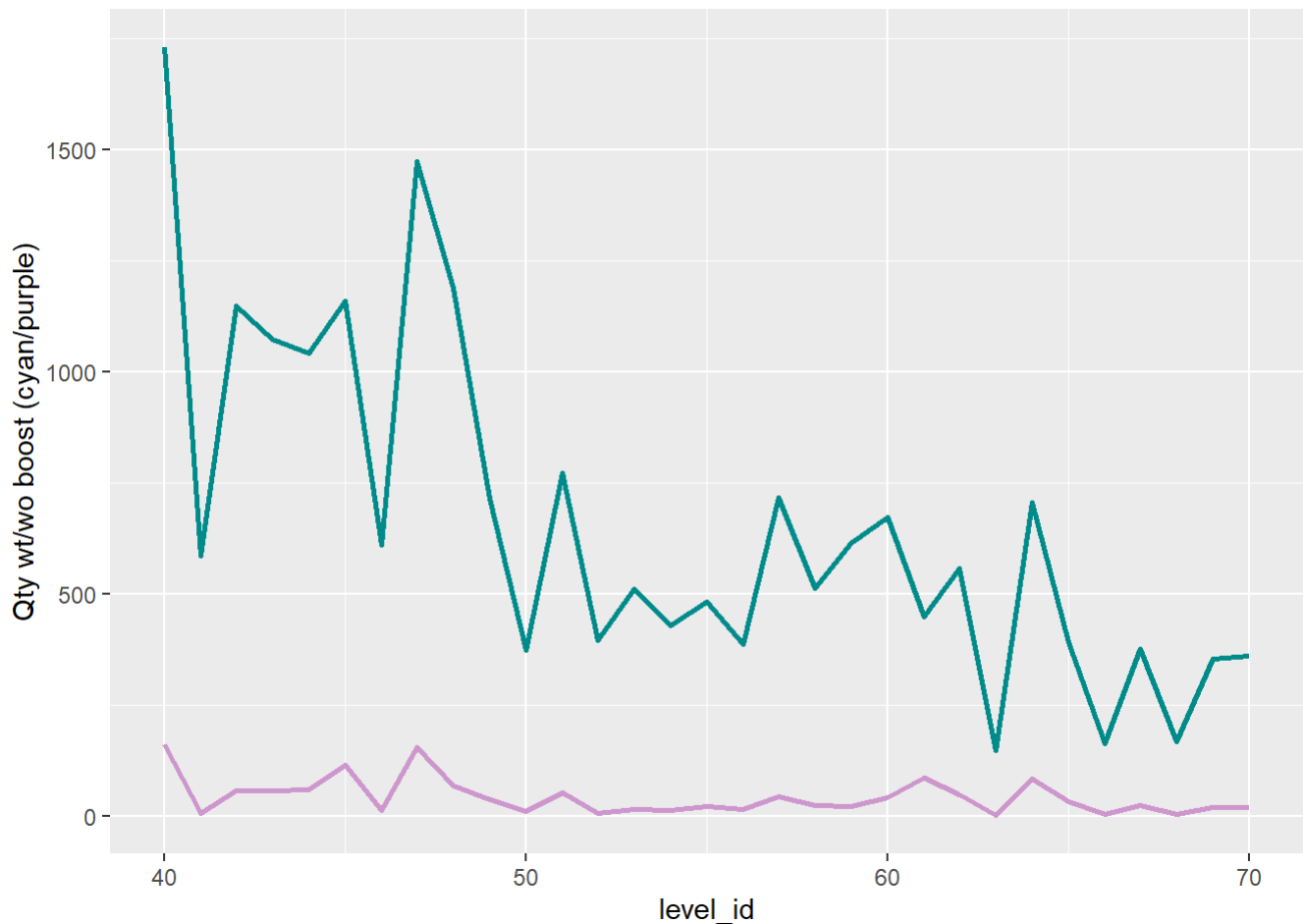


Muchas ayudas al principios, moderadas posteriormente, y en sentido descendente conforme se avanza de nivel

Visualizamos la cantidad de partidas por nivel jugadas con boosters y sin ellos:

```
Comparative_wtwo_Boosters = GT %>%
  replace_na(list(B1=0,B2=0,B3=0)) %>%
  distinct(mission_id, .keep_all = TRUE) %>%
  group_by(level_id) %>%
  mutate(wo_booster=ifelse(B1==0 | B2==0 | B3==0 ,1,0),
         wt_booster=ifelse(B1>0 | B2>0 | B3>0 ,1,0)) %>%
  summarise(wt_booster=sum(wt_booster),
            wo_booster=sum(wo_booster))
```

```
Comparative_wtwo_Boosters %>% ggplot(aes(x=level_id))+
  geom_line(aes(y=wt_booster),size=1,color="plum3")+
  geom_line(aes(y=wo_booster),size=1,color="cyan4")+
  ylab("Qty wt/wo boost (cyan/purple)")
```

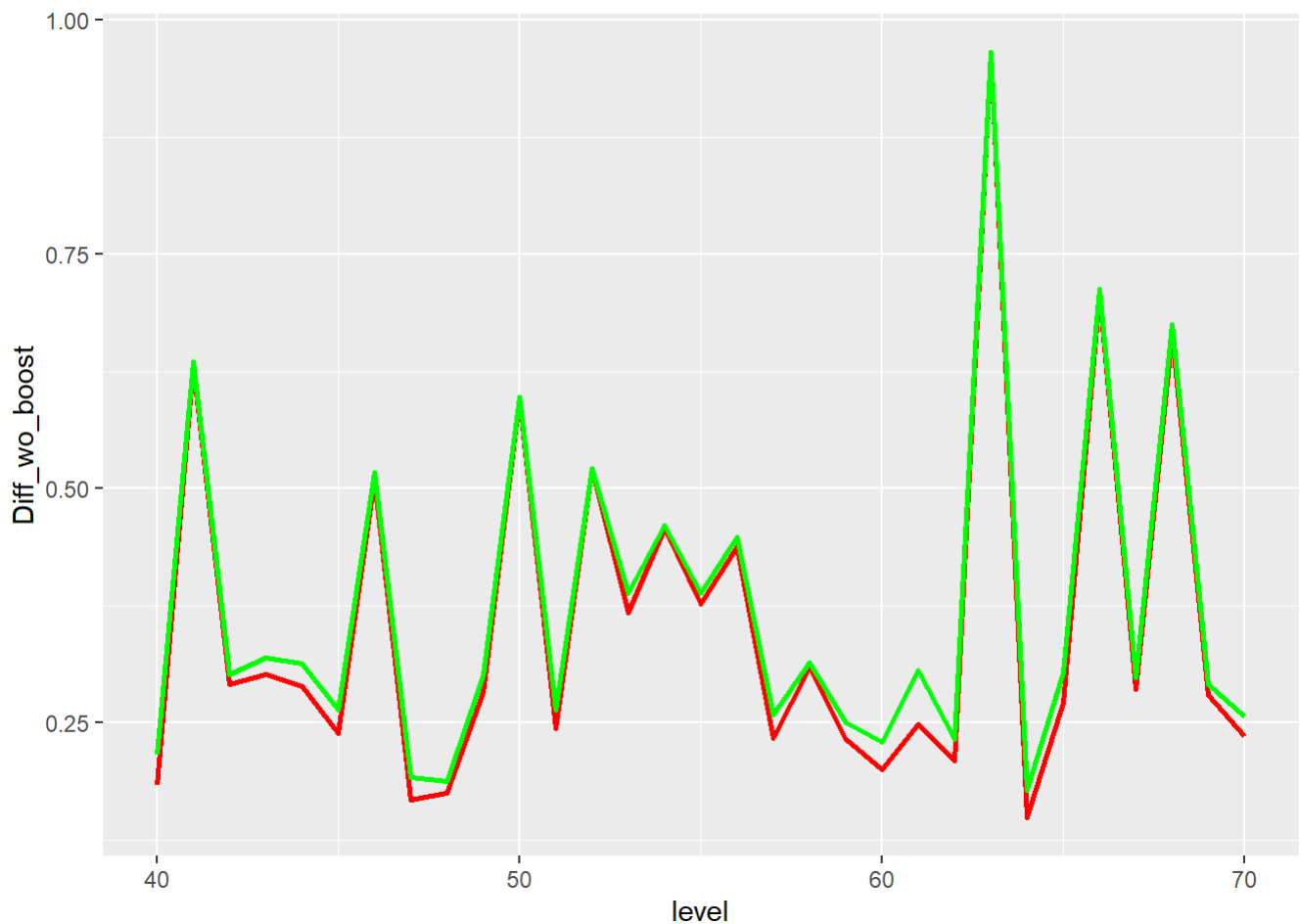


Sólo en los últimos niveles parecen acercarse el número de usos de éstos y con bastante cercanía

```
Difficulty_wt_boosters = GT %>% replace_na(list(B1=0,B2=0,B3=0)) %>%
  group_by(mission_id) %>%
  mutate(start_counter=ifelse(action=="missionStarted",1,0),
         n_starts=sum(start_counter)) %>%
  filter(n_starts == 1) %>%
  select(-c(start_counter,n_starts)) %>%
  filter(row_number()==1 | row_number()==2) %>%
  group_by(level_id) %>%
  mutate(fail=ifelse(action=="missionFailed",1,0),
         comp=ifelse(action=="missionCompleted",1,0)) %>%
  summarise(DIFF = sum(comp)/(sum(comp)+sum(fail)))

Diff_comparative=cbind(Difficulty_wo_boosters,Difficulty_wt_boosters[-1])
names(Diff_comparative)=c("level","Diff_wo_boost","Diff_wt_boost")
```

```
Diff_comparative %>% ggplot(aes(x=level))+
  geom_line(aes(y=Diff_wo_boost),size=1,color="red")+
  geom_line(aes(y=Diff_wt_boost),size=1,color="green")
```



Y lo curioso es que, la dificultad aumenta ligeramente si se cuentan la resolución de partidas con el uso de Boosters

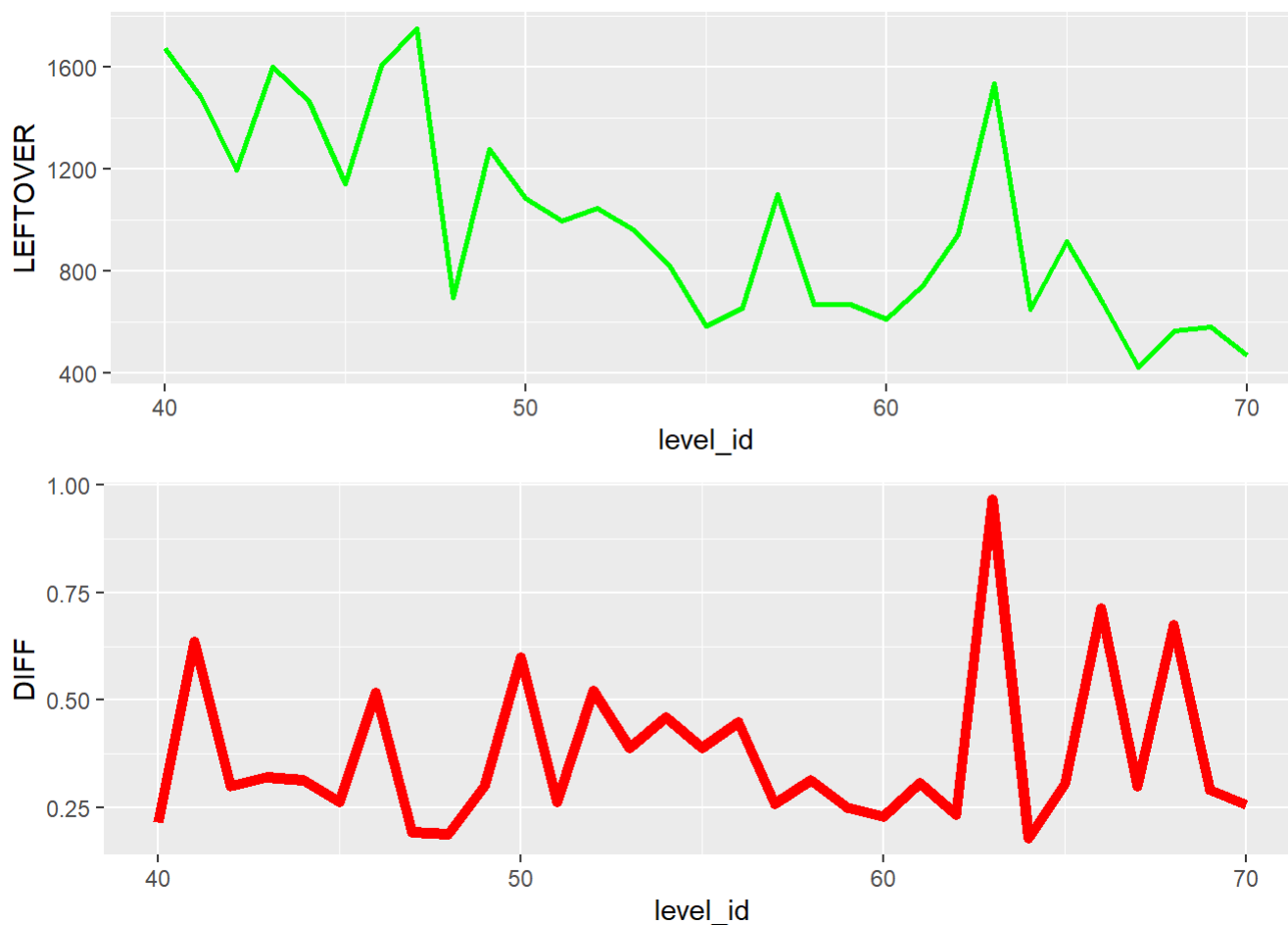
#Ejercicio 4: Comparativa Dificultad vs movimientos sobrantes: ***

Básicamente nos valdremos de la ecuación anterior pero añadiéndole una columna con el recuento de movimientos sobrantes

```
Diff_vs_leftover = GT %>% replace_na(list(B1=0,B2=0,B3=0,leftover=0)) %>%
  group_by(mission_id) %>%
  mutate(start_counter=ifelse(action=="missionStarted",1,0),
         n_starts=sum(start_counter)) %>%
  filter(n_starts == 1) %>%
  select(-c(start_counter,n_starts)) %>%
  filter(row_number()==1 | row_number()==2)%>%
  group_by(level_id) %>%
  mutate(fail=ifelse(action=="missionFailed",1,0),
         comp=ifelse(action=="missionCompleted",1,0)) %>%
  summarise(DIFF = sum(comp)/(sum(comp)+sum(fail)),LEFTOVER=sum(leftover))
```

```
E4.1 = Diff_vs_leftover %>% ggplot(aes(x=level_id))+
  geom_line(aes(y=DIFF),size=2,color="red")
E4.2 = Diff_vs_leftover %>% ggplot(aes(x=level_id))+
  geom_line(aes(y=LEFTOVER),size=1,color="green")

grid.arrange(E4.2,E4.1,nrow=2)
```

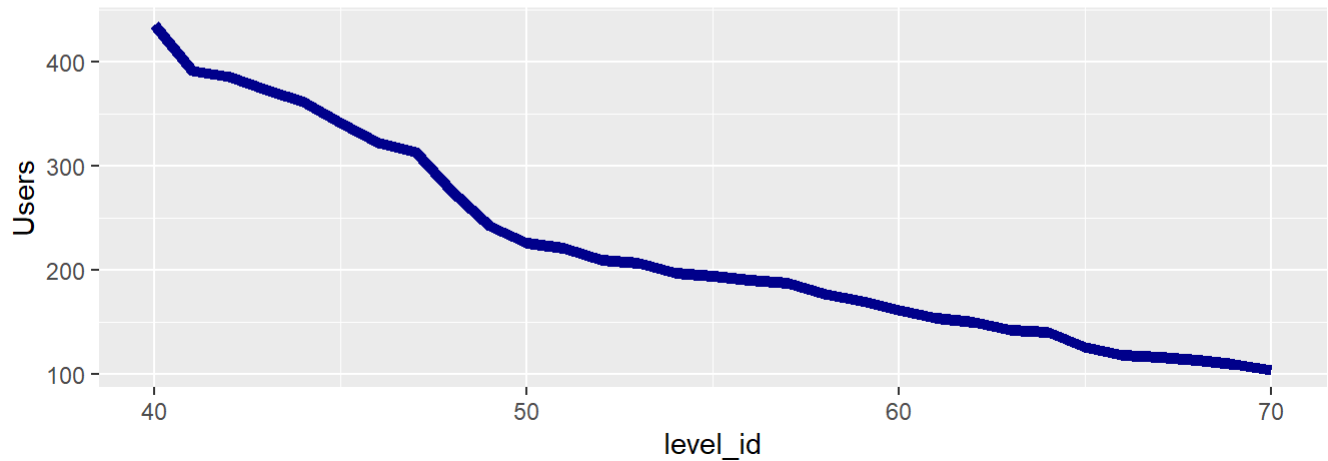
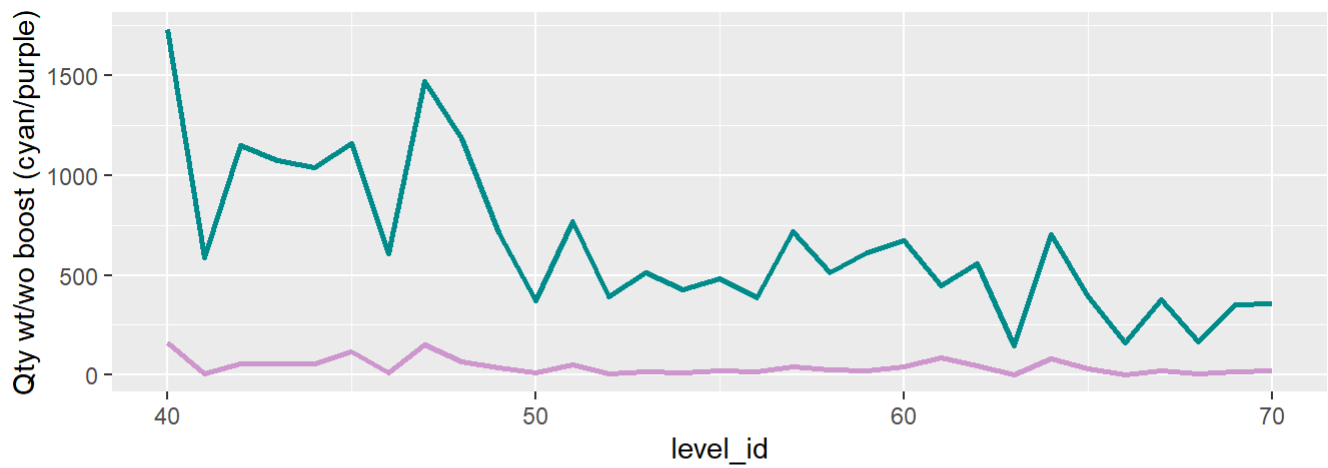


#Ejercicio 5: Conclusiones: ***

- En el gráfico correspondiente a la dificultad vs leftover podemos observar como parece existir una tendencia muy similar en su evolución conforme se aumenta de nivel. Se puede ver claramente como en los picos más significativos (en torno a los niveles 47(positivo), 48.5(negativo), a lo largo de los niveles 50-60 (negativo) y después un gran pico en torno al nivel 63-64).
- Parece que la mayor carga de movimientos sobrantes se da en los niveles de mayor dificultad o previos a éstos, pero en general sigue la misma tendencia que la dificultad, es decir, a mayor dificultad, mayor cantidad de movimientos sobrantes e ídem para la baja dificultad.
- En el caso de uso de Boosters, su uso cae en picado como vemos en la comparativa del gráfico de abajo. Pese a que la tendencia de este es descendente, no es tan pronunciada como lo es la caída de usuarios conforme se avanza. A su lado, parece una tendencia de uso casi uniforme.

```
E5.1 = Comparative_wtwo_Boosters %>% ggplot(aes(x=level_id))+
  geom_line(aes(y=wt_booster),size=1,color="plum3")+
  geom_line(aes(y=wo_booster),size=1,color="cyan4")+
  ylab("Qty wt/wo boost (cyan/purple)")
E5.2 = GT %>% group_by(level_id) %>%
  distinct(user_id) %>%
  summarise(Users=n()) %>%
  ggplot(aes(x=level_id))+
  geom_line(aes(y=Users),size=2,color="darkblue")

grid.arrange(E5.1,E5.2,nrow=2)
```

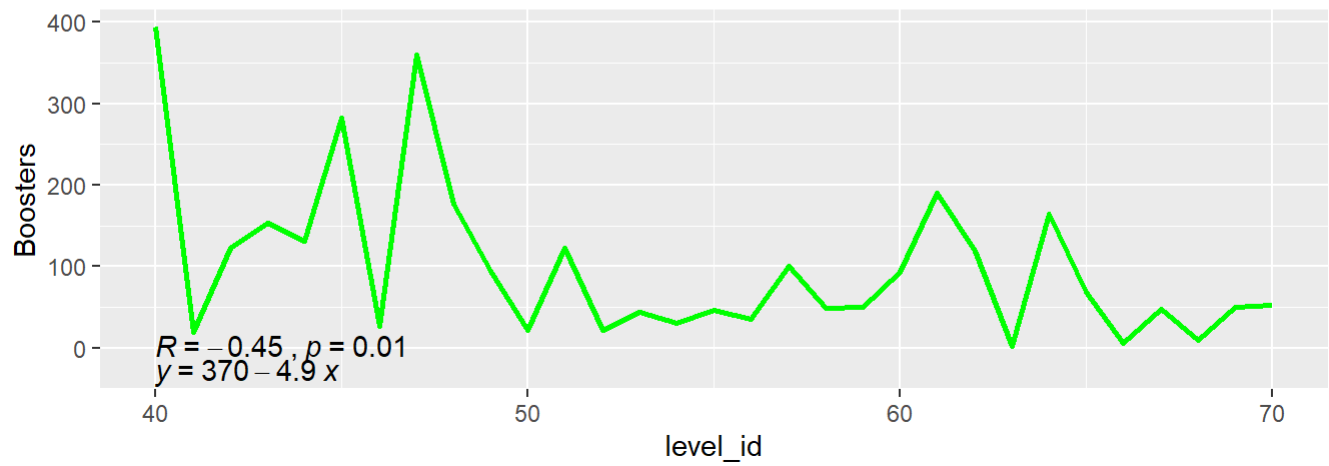
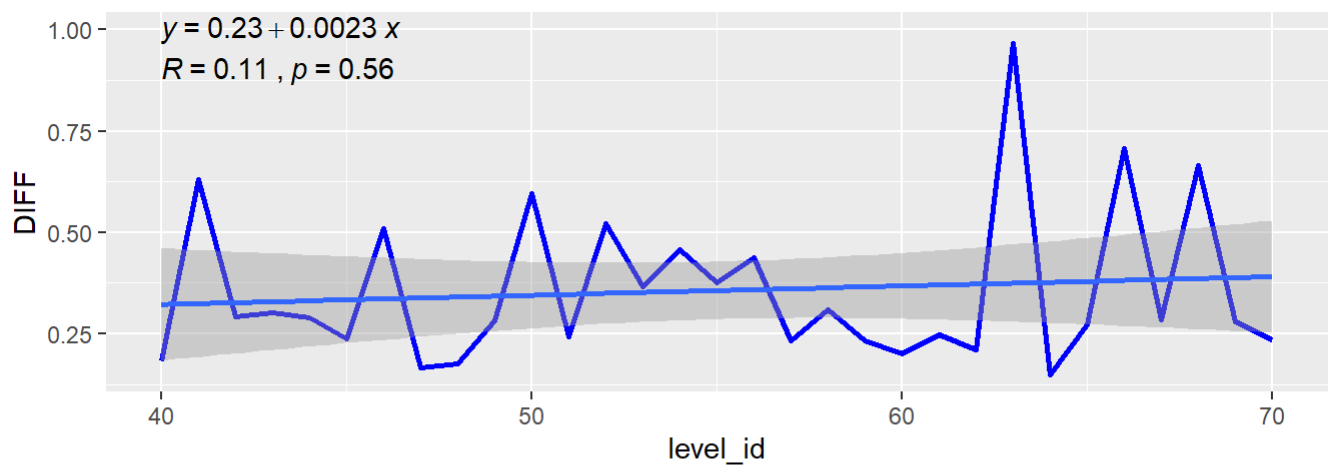



- Sin embargo ahondando un poco más se observa como la zona de niveles de mayor estabilidad evolutiva (~50-60) la caída de usuarios es algo más suave. Se va adquiriendo una mayor resiliencia conforme se sube de nivel.
- Respecto a la dificultad, además de lo observado, hacer hincapié en los niveles donde la tendencia se desregulariza: Niveles 46-50-52-63-66-68. Más abajo podríamos valorar su comportamiento respecto al uso de Boosters:

```
E5.3 = Difficulty_wo_boosters %>% ggplot(aes(x=level_id,y=DIFF))+
  geom_line(size=1,color="blue")+
  stat_cor(label.y = 0.9) +
  stat_regline_equation(label.y = 1)+
  geom_smooth(method='lm', formula= y~x)

E5.4 = Boosters %>% ggplot(aes(x=level_id,y=Boosters))+
  geom_line(size=1,color="green")+
  stat_cor(label.y = 0) +
  stat_regline_equation(label.y = -30)

grid.arrange(E5.3,E5.4,nrow=2)
```



- Se contempla, finalmente, como el uso de Boosters es *inversamente proporcional* a la dificultad del evento, lo que arroja coherencia a unos datos hasta ahora difíciles de encajar. A mayor uso de boosters > mayor éxito > mayor ratio de de missionCompleted > menor dificultad