
REPORTE DE RESULTADOS: REGRESIÓN LINEAL SIMPLE Y MULTIPLE

Objetivos

- Evaluar cómo distintas variables afectan el tiempo de reacción de los pacientes.
- Ver si existen patrones asociados al rendimiento en los juegos terapéuticos.



```
"container">
  class="row">
  class="col-md-6 col-lg-8"> <!--
    <nav id="nav" role="navigation">
      <ul>
        <li><a href="index.html">Home</a><
        <li><a href="home-events.html">Hom
        <li><a href="multi-col-menu.html">
        <li class="has-children"> <a href=
          <ul>
            <li><a href="tall-button-he
            <li><a href="image-logo.htm
            <li class="active"><a href=
          </ul>
        </li>
        <li class="has-children"> <a href=
          <ul>
            <li><a href="variable-width-
```


Pre-procesamiento de los datos

Las columnas “Administrador”, “Usuario”, “mini juego”, “color presionado”, “dificultad” y “Juego” se convirtieron a numéricas según la frecuencia.

```

Dificultad

> df5 = df.groupby(["dificultad"])[["dificultad"].count()
df5
[22]

.. dificultad
Episodio 1    2357
Episodio 2    1493
Episodio 3    1292
Episodio 4     723
Name: dificultad, dtype: int64

df["dificultad"] = df["dificultad"].replace({"Episodio 1": "1", regex=False)
df["dificultad"] = df["dificultad"].replace({"Episodio 2": "2", regex=False)
df["dificultad"] = df["dificultad"].replace({"Episodio 3": "3", regex=False)
df["dificultad"] = df["dificultad"].replace({"Episodio 4": "4", regex=False)
[23]

df5 = df.groupby(["dificultad"])[["dificultad"].count()
df5
[24]

.. dificultad
1      2357
2      1493
3      1292
4       723
Name: dificultad, dtype: int64

```

```

df1 = df.groupby(["color presionado"])[["color presionado"].count()
df1

```

```

color presionado
99      762
blue    1182
green   1298
red      29
violet  1361
yellow  1233
Name: color presionado, dtype: int64

```

```

df["color presionado"] = df["color presionado"].replace({"violet": "1", regex=False)
df["color presionado"] = df["color presionado"].replace({"green": "2", regex=False)
df["color presionado"] = df["color presionado"].replace({"yellow": "3", regex=False)
df["color presionado"] = df["color presionado"].replace({"blue": "4", regex=False)
df["color presionado"] = df["color presionado"].replace({"red": "5", regex=False)
df["color presionado"] = df["color presionado"].replace({"99": "99", regex=False)

```

```

df1 = df.groupby(["color presionado"])[["color presionado"].count()
df1

```

```

color presionado
1      1361
2      1298
3      1233
4      1182
5        29
99      762

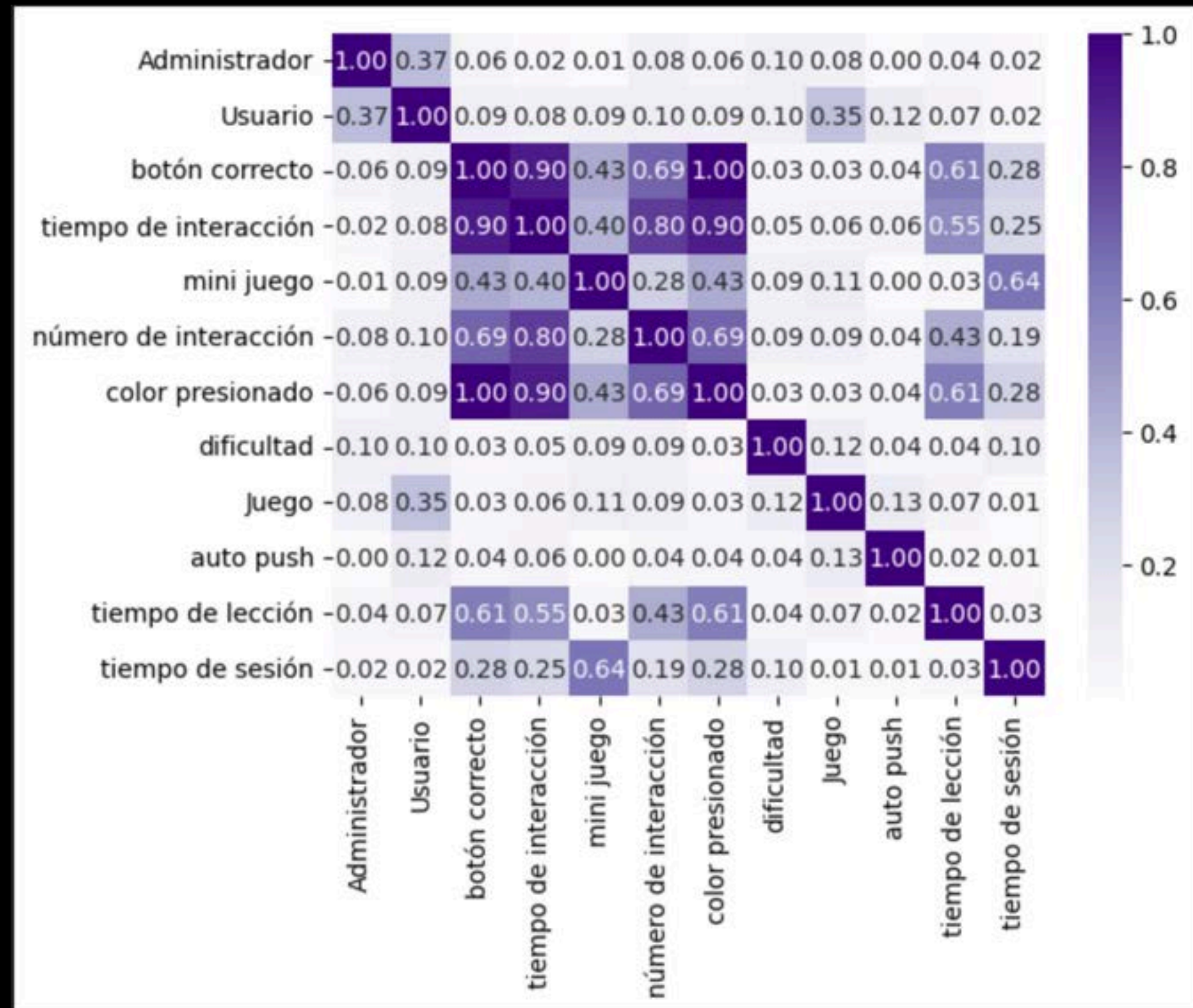
```



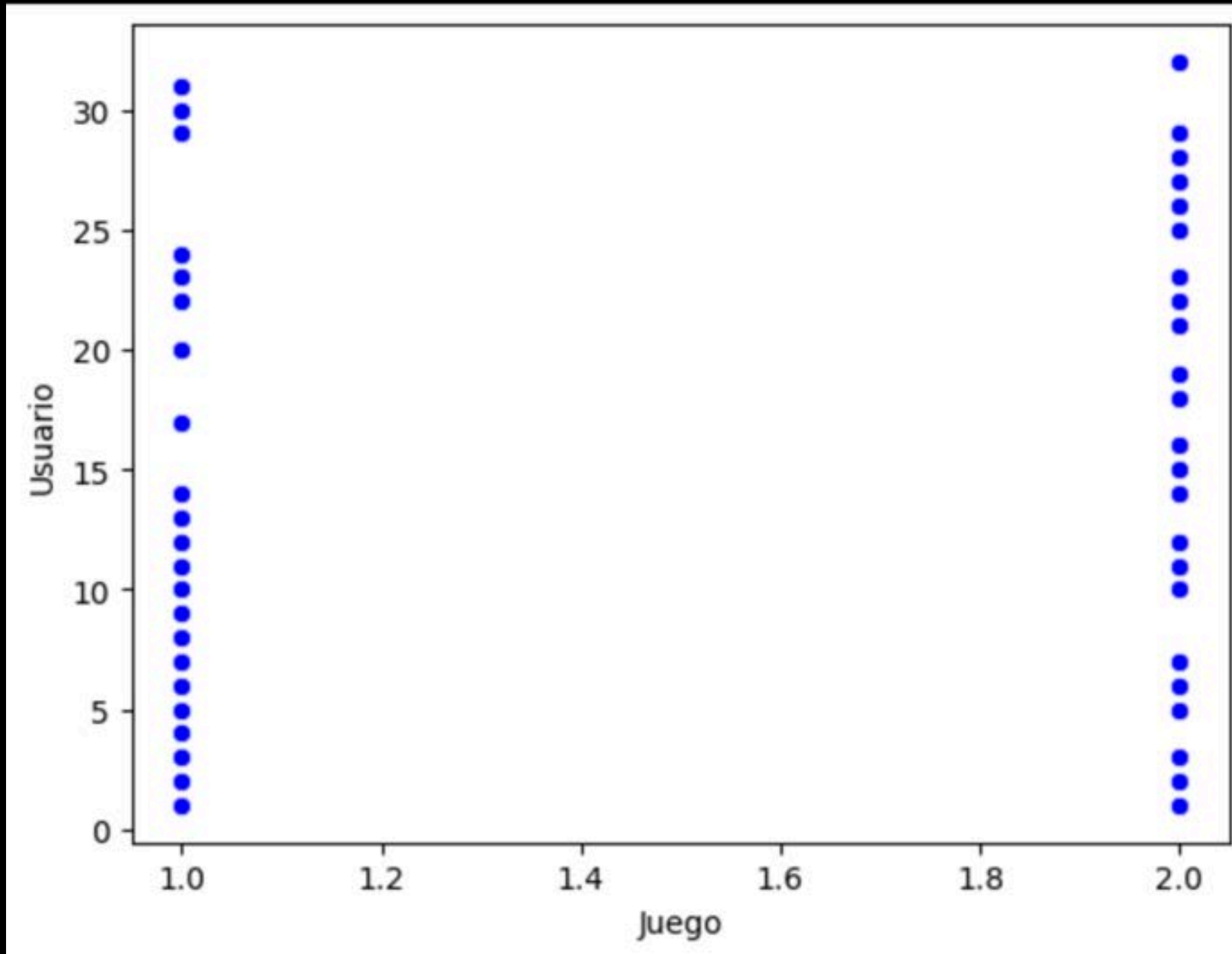
Usuario como variable objetivo

Heatmap con la variable “usuario” como objetivo

En el heatmap podemos observar que la correlación entre nuestra variable objetivo, “usuario”, y la variable “Juego” es la más alta



Regresión Simple



MODELO DE REGRESIÓN SIMPLE

MODELO DE REGRESIÓN SIMPLE

```
Vars_Indep= df[['Juego']]  
Var_Dep= df['Usuario']
```

```
from sklearn.linear_model import LinearRegression  
model= LinearRegression()
```

```
model.fit(X=Vars_Indep, y=Var_Dep)
```

```
LinearRegression ⓘ ?  
LinearRegression()
```

```
coef_DeterS=model.score(Vars_Indep,Var_Dep)  
coef_DeterS
```

```
0.12283515754211238
```

```
#Coeficiente de correlacion  
coef_CorrelS=np.sqrt(coef_DeterS)  
coef_CorrelS
```

```
np.float64(0.3504784694415798)
```

INTERPRETACIÓN:

COEFICIENTE DE DETERMINACIÓN DE 0.12:

- SE CONSIDERA UNA CORRELACIÓN DÉBIL A MODERADA POSITIVA. ES UN INDICADOR DE BAJO PODER PREDICTIVO O EXPLICATIVO.

COEFICIENTE DE CORRELACIÓN DE 0.35:

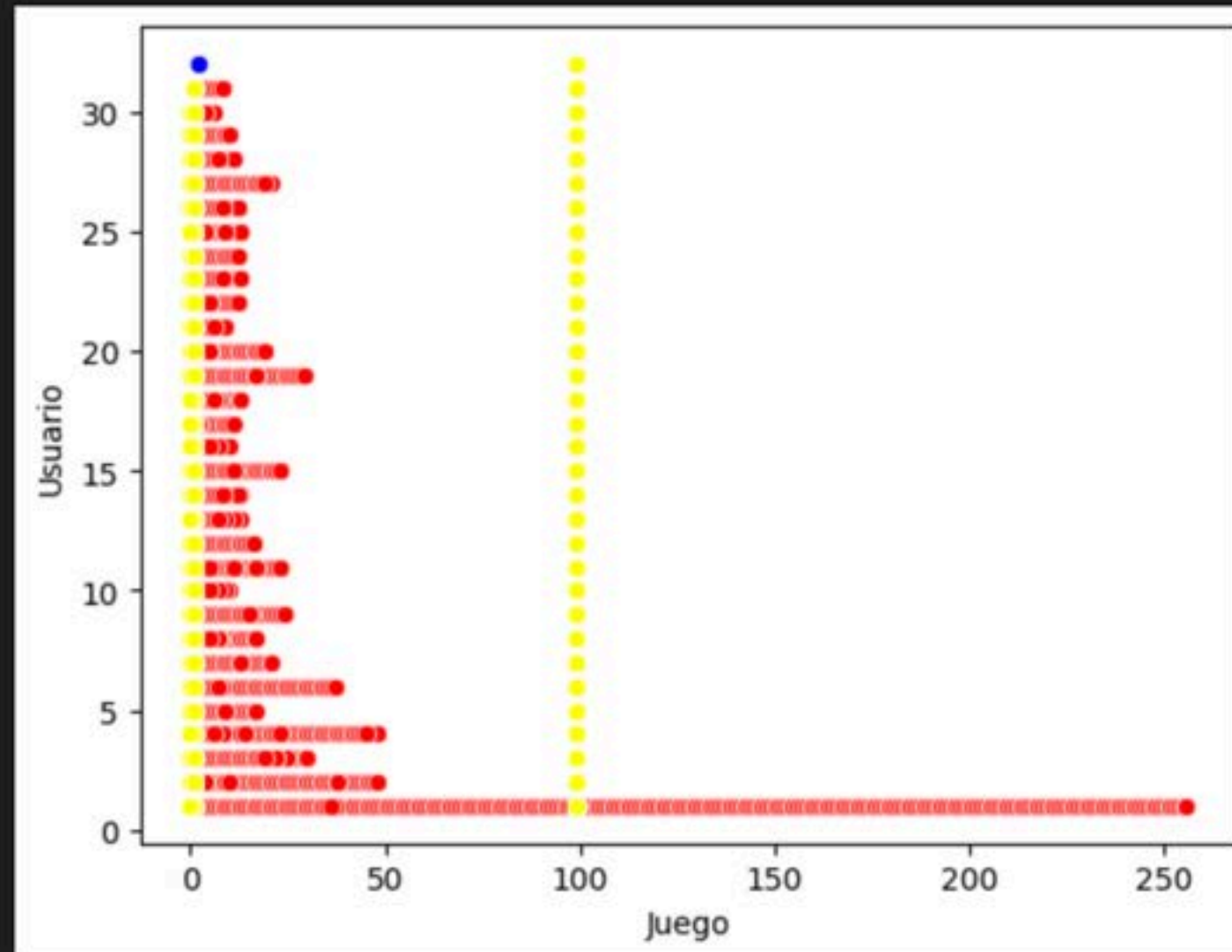
- UN R^2 DE 0.12 SIGNIFICA QUE SOLO EL 12% DE LA VARIABILIDAD EN LOS USUARIOS SE EXPLICA POR LA CANTIDAD DE JUEGOS.
- EL 88% RESTANTE SE DEBE A OTROS FACTORES NO INCLUIDOS EN EL MODELO.

MODELO DE REGRESIÓN MULTIPLE

SELECCIONAMOS LAS VARIABLES CON MAYOR CORRELACIÓN SEGÚN EL HEATMAP PARA AUMENTAR LA CORRELACION Y AL DETERMINACIÓN

```
sns.scatterplot(x='Juego', y='Usuario', color="blue", data=df)  
sns.scatterplot(x='número de interacción', y='Usuario', color="red", data=df)  
sns.scatterplot(x='botón correcto', y='Usuario', color="yellow", data=df)
```

<Axes: xlabel='Juego', ylabel='Usuario'>



MODELO DE REGRESIÓN MULTIPLE

```
Vars_Indep= df[['Juego', 'número de interacción','botón correcto']]
Var_Dep= df['Usuario']
```

```
from sklearn.linear_model import LinearRegression
model= LinearRegression()
```

```
model.fit(X=Vars_Indep, y=Var_Dep)
```

✓ 0.0s

LinearRegression ⓘ ⓘ
LinearRegression()

```
coef_Deter=model.score(Vars_Indep,Var_Dep)
coef_Deter
```

✓ 0.0s

0.15891171123791037

```
coef_Correl=np.sqrt(coef_Deter)
coef_Correl
```

✓ 0.0s

np.float64(0.3986373179193217)

INTERPRETACIÓN:

COEFICIENTE DE DETERMINACIÓN DE 0.15:

- UN R^2 DE 0.1589 SIGNIFICA QUE EL 15.89% DEL COMPORTAMIENTO DE LOS USUARIOS SE PUEDE EXPLICAR POR EL JUEGO, INTERACCIÓN Y BOTÓN CORRECTO.
- EL 84.11% RESTANTE SE DEBE A OTROS FACTORES QUE EL MODELO NO ESTÁ CAPTURANDO.

COEFICIENTE DE CORRELACIÓN DE 0.39:

- UN R DE 0.39 INDICA UNA CORRELACIÓN DÉBIL A MODERADA POSITIVA.
- ES DECIR, CUANDO AUMENTAN LOS VALORES DE JUEGO, INTERACCIÓN O BOTÓN CORRECTO, TAMBIÉN TIENDE A AUMENTAR EL NÚMERO DE USUARIOS, AUNQUE LA RELACIÓN NO ES MUY FUERTE.

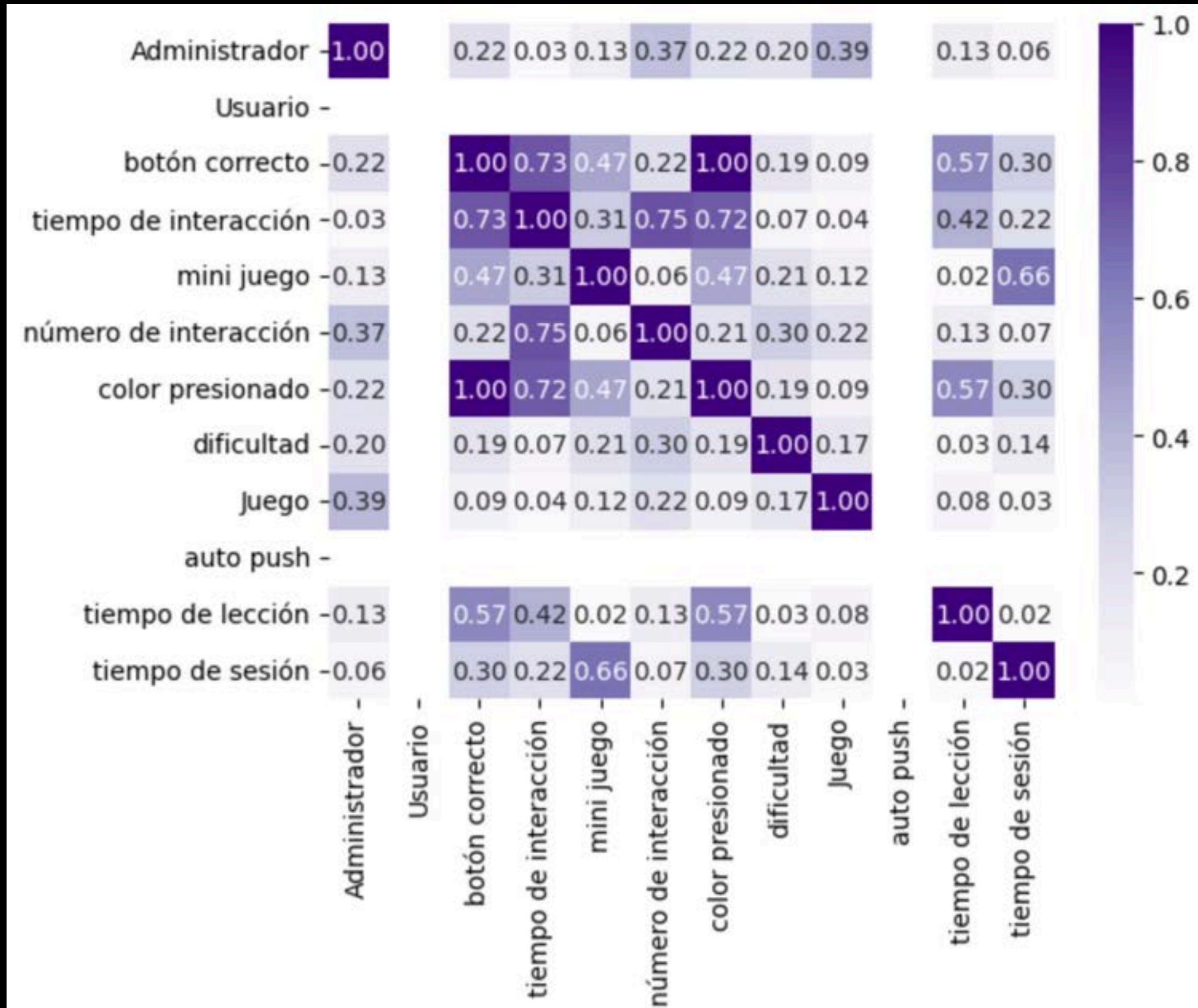
CONCLUSIONES GENERALES:

- HAY UNA MEJORA LIGERA RESPECTO AL MODELO ANTERIOR (QUE SOLO INCLUÍA JUEGO), PERO LA RELACIÓN SIGUE SIENDO DÉBIL.}
- AUNQUE LAS NUEVAS VARIABLES (INTERACCIÓN Y BOTÓN CORRECTO) AYUDAN UN POCO, LA MAYORÍA DE LA VARIACIÓN EN LOS USUARIOS DEPENDE DE OTRAS COSAS QUE AÚN NO SE ESTÁN CONSIDERANDO.

**EN EL SIGUIENTE PASO SE FILTRARON A 5 PACIENTES
ESPECIFICOS PARA ENCONTRAR INFORMACIÓN QUE
PUDIESE SER RELEVANTE**

LEONARDO

Heatmap

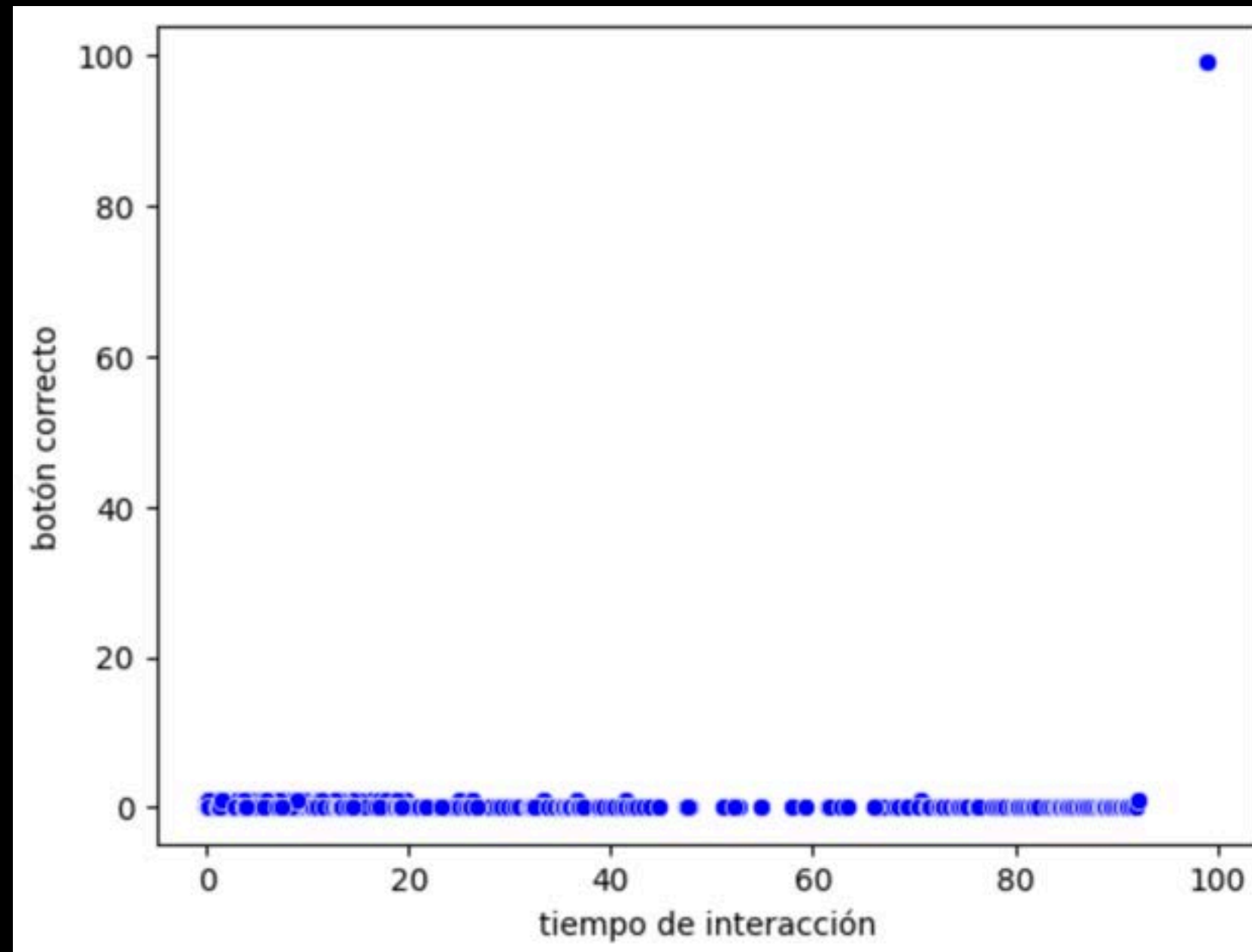


Hallazgos

LA VARIABLE TIEMPO DE INTERACCIÓN TUVO UNA CORRELACIÓN DE ALTA EN TODOS LOS PACIENTES CON NÚMERO DE INTERACCIÓN. POR LO QUE SE CREÓ UN MODELO DE REGRESIÓN SIMPLE USANDO ESAS VARIABLES.

Regresión Simple

LEONARDO



Regresión Simple

LEONARDO

INTERPRETACIÓN:

- COEFICIENTE DE DETERMINACIÓN: 0.52
- ES UN VALOR MODERADO A ALTO, LO CUAL SUGIERE QUE EL COLOR INFLUYE BASTANTE EN CÓMO REACCIONAN LOS USUARIOS.
- EL OTRO 48% DEPENDE DE OTROS FACTORES, COMO ATENCIÓN, FATIGA, EDAD, CONTEXTO VISUAL, ENTRE OTROS.

- COEFICIENTE DE CORRELACIÓN: 0.72

- ESO SIGNIFICA QUE EL COLOR QUE SE PRESIONA SÍ TIENE UN IMPACTO IMPORTANTE EN EL TIEMPO QUE TARDA UNA PERSONA EN REACCIONAR.
- A MEDIDA QUE CAMBIA EL COLOR, TAMBIÉN CAMBIA EL TIEMPO DE REACCIÓN DE MANERA BASTANTE CONSISTENTE.

```
Vars_Indep= df_filtrado1[['color presionado']]
Var_Dep= df_filtrado1['tiempo de interacción']
```

✓ 0.0s

```
from sklearn.linear_model import LinearRegression
model= LinearRegression()
```

✓ 0.0s

```
model.fit(X=Vars_Indep, y=Var_Dep)
```

✓ 0.0s

LinearRegression ⓘ ⓘ
LinearRegression()

```
coef_Deter_Leonardo=model.score(Vars_Indep,Var_Dep)
coef_Deter_Leonardo
```

✓ 0.0s

0.522752304516322

```
coef_Correl_Leonardo=np.sqrt(coef_Deter_Leonardo)
coef_Correl_Leonardo
```

✓ 0.0s

np.float64(0.7230161163600173)

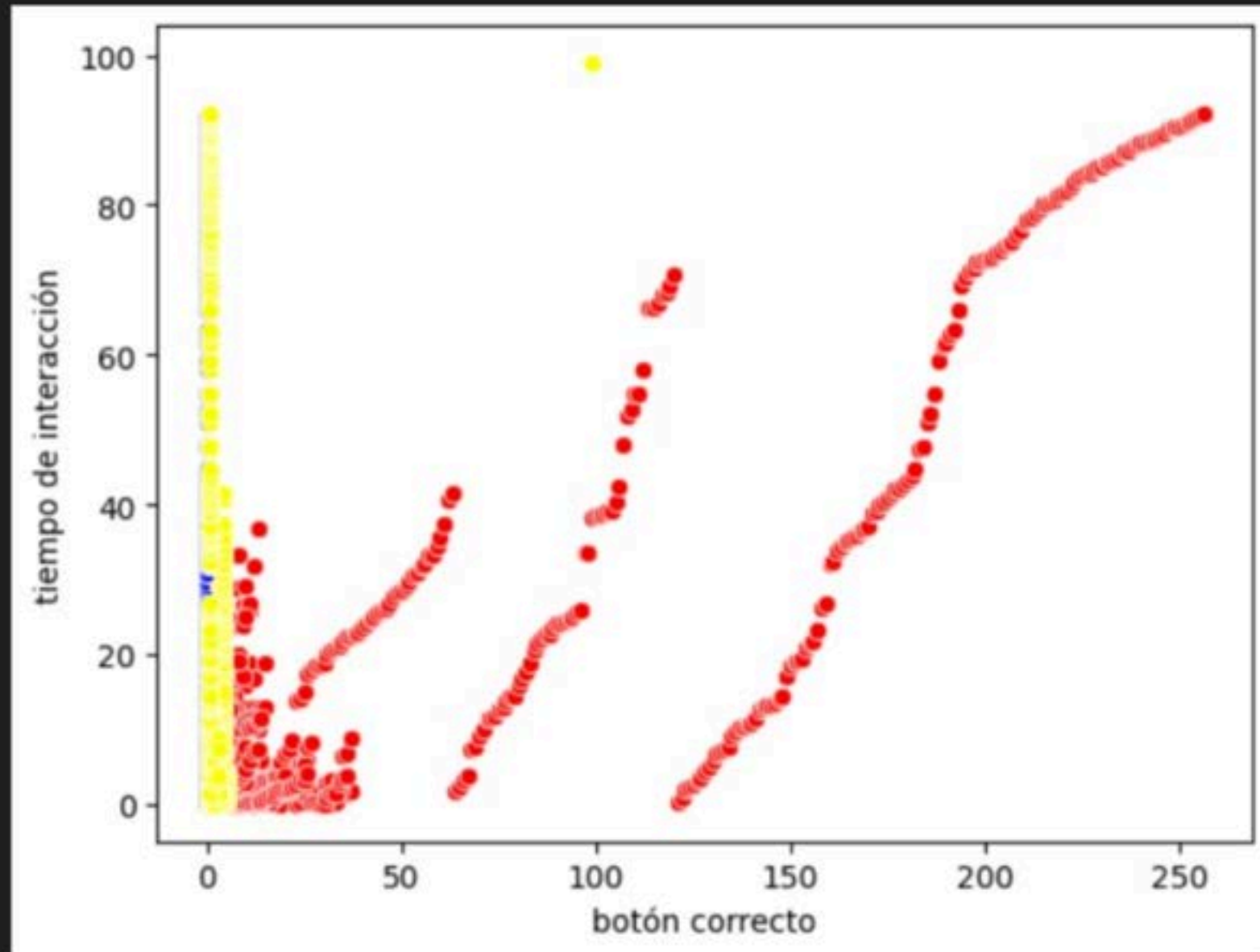
Regresión Multiple

LEONARDO

```
sns.scatterplot(x='botón correcto', y='tiempo de interacción', color="blue", data=df_filtrado1)  
sns.scatterplot(x='número de interacción', y='tiempo de interacción', color="red", data=df_filtrado1)  
sns.scatterplot(x='color presionado', y='tiempo de interacción', color="yellow", data=df_filtrado1)
```

✓ 0.1s

<Axes: xlabel='botón correcto', ylabel='tiempo de interacción'>



Regresión Multiple

LEONARDO

```
Vars_Indep= df_filtrado1[['botón correcto', 'número de interacción','color presionado']]
Var_Dep= df_filtrado1['tiempo de interacción']
```

✓ 0.0s

```
from sklearn.linear_model import LinearRegression
model= LinearRegression()
```

✓ 0.0s

```
model.fit(X=Vars_Indep, y=Var_Dep)
```

✓ 0.0s

LinearRegression ⓘ ⓘ
LinearRegression()

```
coef_Deter_Leonardo=model.score(Vars_Indep,Var_Dep)
coef_Deter_Leonardo
```

✓ 0.0s

0.899249276638887

```
coef_Correl_Leonardo=np.sqrt(coef_Deter_Leonardo)
coef_Correl_Leonardo
```

✓ 0.0s

np.float64(0.9482875495538718)

INTERPRETACIÓN:

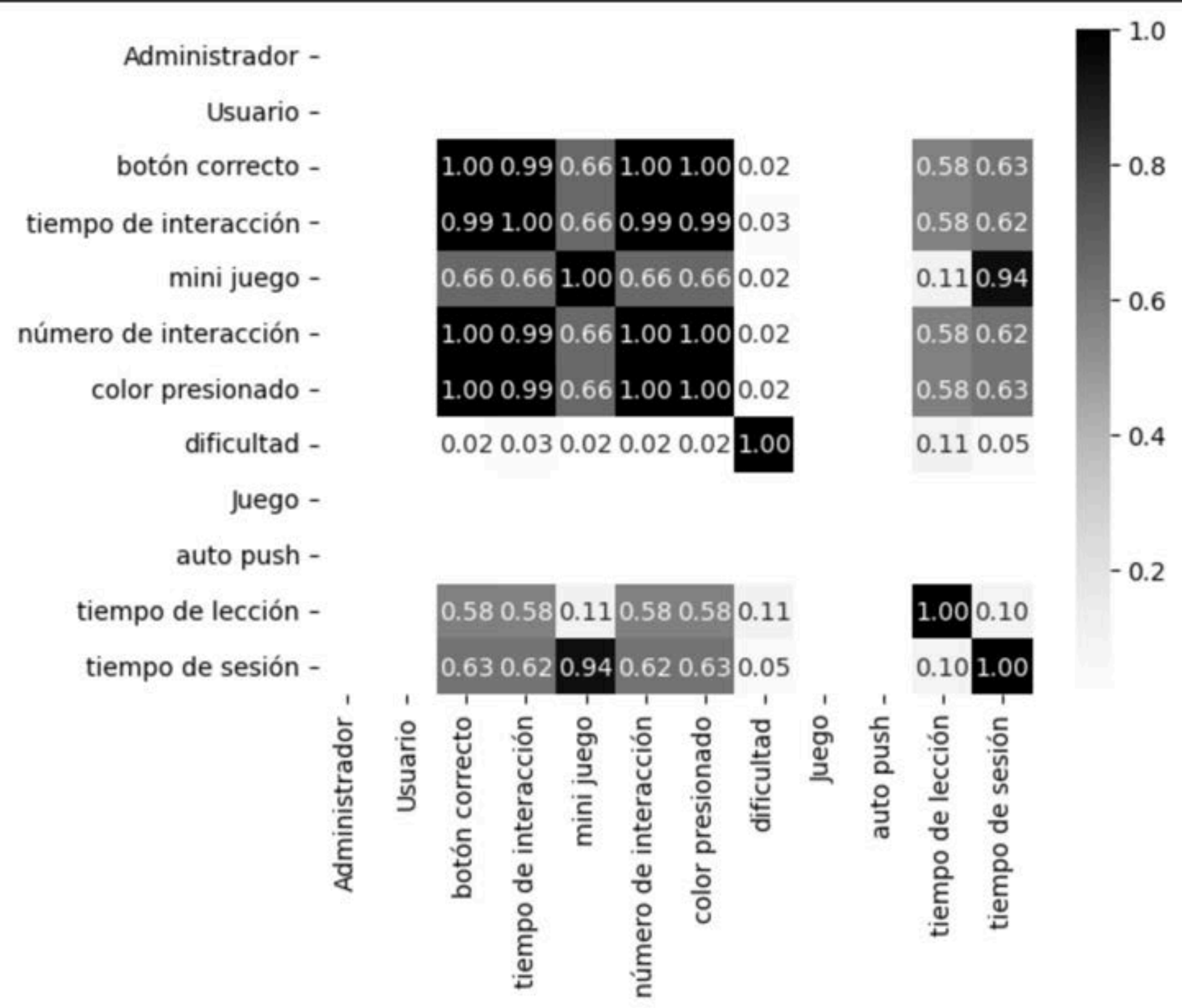
- **COEFICIENTE DE DETERMINACIÓN: 0.89**
- **ESTE VALOR INDICA UNA RELACIÓN MUY FUERTE Y POSITIVA ENTRE LAS VARIABLES INDEPENDIENTES (EN CONJUNTO) Y EL TIEMPO DE INTERACCIÓN.**
- **COEFICIENTE DE CORRELACIÓN: 0.94**
- **HAY UNA RELACIÓN MUY CLARA Y FUERTE ENTRE LAS VARIABLES (BOTÓN CORRECTO, NÚMERO DE INTERACCIÓN Y COLOR PRESIONADO) Y EL TIEMPO DE INTERACCIÓN.**

MA DEL ROSARIO

Regresión Simple

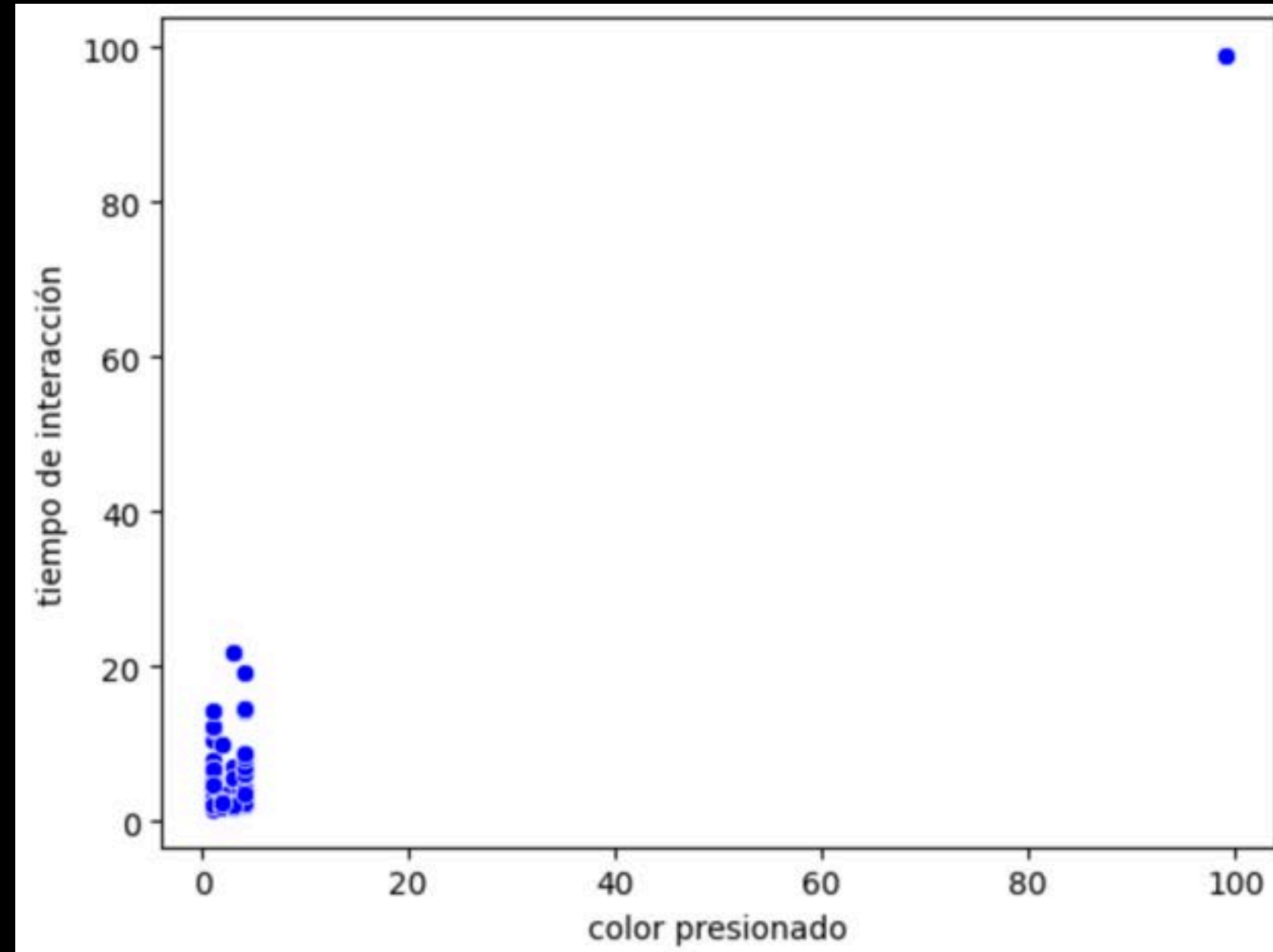
MA DEL ROSARIO

HEAT MAP



MA DEL ROSARIO SE CREÓ UN MODELO CON LAS MISMAS VARIABLES USADAS CON EL PACIENTE ANTERIOR

TIEMPO DE REACCIÓN COMO DEPENDIENTE



Regresión Simple

MA DEL ROSARIO

```
Vars_Indep= df_filtrado2[['color presionado']]
Var_Dep= df_filtrado2['tiempo de interacción']
✓ 0.0s

from sklearn.linear_model import LinearRegression
model= LinearRegression()
✓ 0.0s

model.fit(X=Vars_Indep, y=Var_Dep)
✓ 0.0s

LinearRegression
LinearRegression()

coef_Deter_Ma_Del_Rosario=model.score(Vars_Indep,Var_Dep)
coef_Deter_Ma_Del_Rosario
✓ 0.0s

0.986480456906192

coef_Correl_Ma_Del_Rosario=np.sqrt(coef_Deter_Ma_Del_Rosario)
coef_Correl_Ma_Del_Rosario
✓ 0.0s

np.float64(0.9932172254377146)
```

INTERPRETACIÓN:

COEFICIENTE DE DETERMINACIÓN (R^2): 0.98

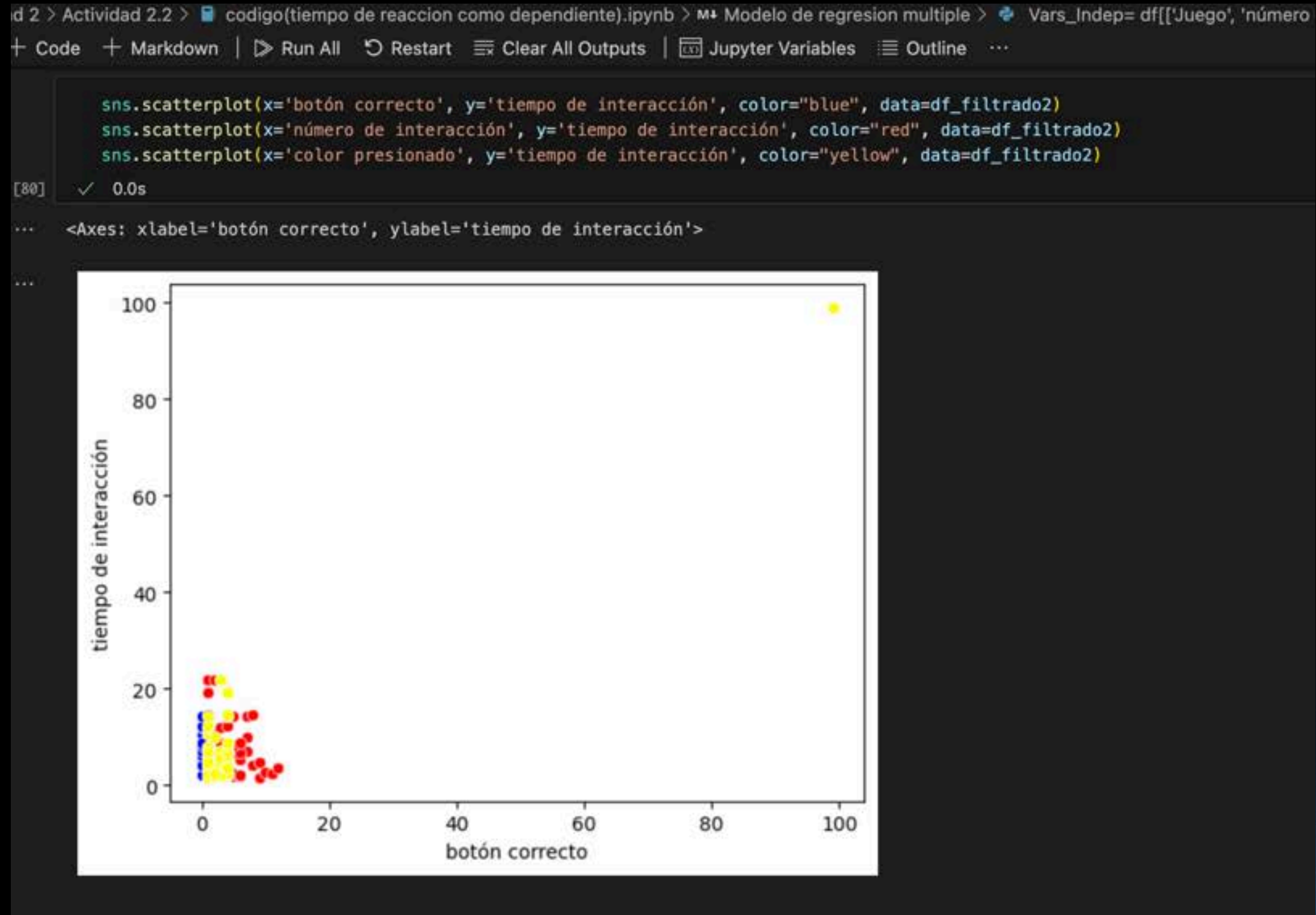
EL 98.03% DEL TIEMPO DE INTERACCIÓN DE ROSARIO PUEDE EXPLICARSE ÚNICAMENTE POR EL NÚMERO DE INTERACCIONES QUE HA REALIZADO. ESTO INDICA UNA RELACIÓN MUY FUERTE Y CONSISTENTE ENTRE AMBAS VARIABLES.

COEFICIENTE DE CORRELACIÓN (R): 0.99

EXISTE UNA CORRELACIÓN POSITIVA EXTREMADAMENTE FUERTE. ES DECIR, A MEDIDA QUE AUMENTA EL NÚMERO DE INTERACCIONES, TAMBIÉN LO HACE EL TIEMPO DE INTERACCIÓN, CASI DE FORMA PROPORCIONAL.

Regresión múltiple

MA DEL ROSARIO



Regresión múltiple

MA DEL ROSARIO

INTERPRETACIÓN:

- **COEFICIENTE DE DETERMINACIÓN (R^2): 0.9867**

EL MODELO MÚLTIPLE EXPLICA EL 98.67% DEL TIEMPO DE INTERACCIÓN, LO QUE REPRESENTA UNA MEJORA LEVE RESPECTO AL MODELO SIMPLE.

- **CORRELACIÓN MÚLTIPLE (R): 0.9933**

LA COMBINACIÓN DE LAS TRES VARIABLES INDEPENDIENTES TIENE UNA RELACIÓN MUY FUERTE CON EL TIEMPO DE INTERACCIÓN. ESTO CONFIRMA QUE ESTOS FACTORES EXPLICAN CASI POR COMPLETO EL COMPORTAMIENTO TEMPORAL DE ROSARIO EN LA ACTIVIDAD.

```
Vars_Indep= df_filtrado2[['botón correcto', 'número de interacción','color presionado']]
Var_Dep= df_filtrado2['tiempo de interacción']
```

✓ 0.0s

```
from sklearn.linear_model import LinearRegression
model= LinearRegression()
```

✓ 0.0s

```
model.fit(X=Vars_Indep, y=Var_Dep)
```

✓ 0.0s

LinearRegression ⓘ ?

LinearRegression()

```
coef_Deter_Ma_Del_Rosario=model.score(Vars_Indep,Var_Dep)
coef_Deter_Ma_Del_Rosario
```

✓ 0.0s

0.9869989145568099

```
coef_Correl_Ma_Del_Rosario=np.sqrt(coef_Deter_Ma_Del_Rosario)
coef_Correl_Ma_Del_Rosario
```

✓ 0.0s

np.float64(0.9934781902773758)

	Administrador	Usuario	botón correcto	tiempo de interacción	mini juego	número de interacción	color presionado	dificultad	juego	auto push	tiempo de lección	tiempo de sesión
Administrador	1.00	0.22	0.03	0.13	0.37	0.22	0.20	0.39	0.13	0.06		
Usuario	-											
botón correcto	-0.22	1.00	0.73	0.47	0.22	1.00	0.19	0.09	0.57	0.30		
tiempo de interacción	-0.03	0.73	1.00	0.31	0.75	0.72	0.07	0.04	0.42	0.22		
mini juego	-0.13	0.47	0.31	1.00	0.06	0.47	0.21	0.12	0.02	0.66		
número de interacción	-0.37	0.22	0.75	0.06	1.00	0.21	0.30	0.22	0.13	0.07		
color presionado	-0.22	1.00	0.72	0.47	0.21	1.00	0.19	0.09	0.57	0.30		
dificultad	-0.20	0.19	0.07	0.21	0.30	0.19	1.00	0.17	0.03	0.14		
juego	-0.39	0.09	0.04	0.12	0.22	0.09	0.17	1.00	0.08	0.03		
auto push	-											
tiempo de lección	-0.13	0.57	0.42	0.02	0.13	0.57	0.03	0.08	1.00	0.02		
tiempo de sesión	-0.06	0.30	0.22	0.66	0.07	0.30	0.14	0.03	0.02	1.00		

[illegible]

	Administrador	Usuario	botón correcto	tiempo de interacción	mini juego	número de interacción	color presionado	dificultad	juego	auto push	tiempo de lección	tiempo de sesión
Administrador	1.00	0.03	0.07	0.02	0.02	0.03	0.54	0.24	0.03	0.04	0.00	
Usuario		1.00	0.85	0.54	0.98	1.00	0.01	0.02	0.04	0.63	0.29	
botón correcto	-0.03		1.00	0.85	0.54	0.98	1.00	0.01	0.02	0.04	0.63	0.29
tiempo de interacción	-0.07		0.85	1.00	0.47	0.85	0.85	0.15	0.02	0.10	0.53	0.25
mini juego	-0.02		0.54	0.47	1.00	0.53	0.54	0.02	0.03	0.00	0.03	0.54
número de interacción	-0.02		0.98	0.85	0.53	1.00	0.98	0.03	0.02	0.04	0.61	0.29
color presionado	-0.03		1.00	0.85	0.54	0.98	1.00	0.01	0.01	0.04	0.63	0.29
dificultad	-0.54		0.01	0.15	0.02	0.03	0.01	1.00	0.34	0.03	0.07	0.02
juego	-0.24		0.02	0.02	0.03	0.02	0.01	0.34	1.00	0.18	0.08	0.02
auto push	-0.03		0.04	0.10	0.00	0.04	0.04	0.03	0.18	1.00	0.03	0.01
tiempo de lección	-0.04		0.63	0.53	0.03	0.61	0.63	0.07	0.08	0.03	1.00	0.02
tiempo de sesión	-0.00		0.29	0.25	0.54	0.29	0.29	0.02	0.02	0.01	0.02	1.00

[illegible]

Heatmap showing the correlation matrix for variables related to the game. The color scale ranges from 0.0 (light yellow) to 1.0 (dark blue).

	Administrador	Usuario	botón correcto	tiempo de interacción	mini juego	número de interacción	color presionado	dificultad	juego	auto push	tiempo de lección	tiempo de sesión
Administrador	1.00	0.97	0.34	1.00	1.00	0.01	0.64	0.24	0.62	0.24	0.02	0.10
Usuario	0.97	1.00	0.33	0.97	0.97	0.00	0.62	0.24	0.64	0.24	0.00	0.00
botón correcto	0.34	0.33	1.00	0.33	0.34	0.10	0.02	0.76	0.02	0.76	0.02	0.02
tiempo de interacción	1.00	0.97	0.33	1.00	1.00	0.00	0.64	0.24	0.64	0.24	0.00	0.00
mini juego	1.00	0.97	0.34	1.00	1.00	0.02	0.64	0.24	0.64	0.24	0.02	0.02
número de interacción	0.01	0.00	0.10	0.00	0.02	1.00	0.05	0.13	0.05	0.13	1.00	0.00
color presionado	0.64	0.62	0.02	0.64	0.64	0.05	1.00	0.03	1.00	0.03	0.00	0.00
dificultad	0.24	0.24	0.76	0.24	0.24	0.13	0.03	1.00	0.03	1.00	0.00	0.00
juego	0.62	0.24	0.02	0.64	0.64	0.05	1.00	0.03	1.00	0.03	0.00	0.00
auto push	0.02	0.76	0.02	0.64	0.64	0.02	0.05	0.13	0.03	1.00	0.00	0.00
tiempo de lección	0.10	0.00	0.02	0.00	0.02	1.00	0.05	0.13	0.03	0.03	1.00	0.00
tiempo de sesión	0.00	0.00	0.02	0.00	0.02	0.00	0.05	0.13	0.03	0.03	0.00	1.00

Hallazgos

LA VARIABLE TIEMPO DE INTERACCIÓN TUVO UNA CORRELACIÓN DE ALTA EN TODOS LOS PACIENTES CON COLOR PRESIONADO POR LO QUE SE CREÓ UN MODELO DE REGRESIÓN SIMEPLE USANDO ESAS VARIABLES.

MODELO DE REGRESIÓN SIMPLE

```
Vars_Indep= df_filtrado1[['color presionado']]  
Var_Dep= df_filtrado1['tiempo de interacción']
```

✓ 0.0s

```
from sklearn.linear_model import LinearRegression  
model= LinearRegression()
```

✓ 0.0s

```
model.fit(X=Vars_Indep, y=Var_Dep)
```

✓ 0.0s

▼ LinearRegression ⓘ ?

LinearRegression()

RESULTADOS DE REGRESIÓN SIMPLE

```
coef_Deter_Leonardo=model.score(Vars_Indep,Var_Dep)
coef_Deter_Leonardo
```

✓ 0.0s

0.522752304516322

```
coef_Correl_Leonardo=np.sqrt(coef_Deter_Leonardo)
coef_Correl_Leonardo
```

✓ 0.0s

np.float64(0.7230161163600173)

```
coef_Deter_Ma_Del_Rosario=model.score(Vars_Indep,Var_Dep)
coef_Deter_Ma_Del_Rosario
```

✓ 0.0s

0.986480456906192

```
coef_Correl_Ma_Del_Rosario=np.sqrt(coef_Deter_Ma_Del_Rosario)
coef_Correl_Ma_Del_Rosario
```

✓ 0.0s

np.float64(0.9932172254377146)

```
coef_Deter_Nicolas=model.score(Vars_Indep,Var_Dep)
coef_Deter_Nicolas
```

✓ 0.0s

0.718949654259092

```
coef_Correl_Nicolas=np.sqrt(coef_Deter_Nicolas)
coef_Correl_Nicolas
```

✓ 0.0s

np.float64(0.8479089893727345)

```
coef_Deter_Rene=model.score(Vars_Indep,Var_Dep)
coef_Deter_Rene
```

✓ 0.0s

0.7603262434611102

```
coef_Correl_Rene=np.sqrt(coef_Deter_Rene)
coef_Correl_Rene
```

✓ 0.0s

np.float64(0.8719668820896297)

```
#Coeficiente de determinacion
coef_Deter_Sergio=model.score(X=Vars_Indep, y=Var_Dep)
coef_Deter_Sergio
```

✓ 0.0s

0.9357466041432032

```
#Corroboramos cual es el coeficiente de Correlación de nuestro modelo
coef_Correl_Sergio=np.sqrt(coef_Deter_Sergio)
coef_Correl_Sergio
```

✓ 0.0s

np.float64(0.9673399630653141)

MODELO DE REGRESIÓN MULTIPLE

```
Vars_Indep= df_filtrado1[['botón correcto', 'número de interacción','color presionado']]  
Var_Dep= df_filtrado1['tiempo de interacción']
```

✓ 0.0s

```
from sklearn.linear_model import LinearRegression  
model= LinearRegression()
```

✓ 0.0s

```
model.fit(X=Vars_Indep, y=Var_Dep)
```

✓ 0.0s

▼ LinearRegression ⓘ ?

LinearRegression()

RESULTADOS DE REGRESIÓN MULTIPLE

<pre>coef_Deter_Leonardo=model.score(Vars_Indep,Var_Dep) coef_Deter_Leonardo</pre>
0.899249276638887
<pre>coef_Correl_Leonardo=np.sqrt(coef_Deter_Leonardo) coef_Correl_Leonardo</pre>
np.float64(0.9482875495538718)

<pre>coef_Deter_Ma_Del_Rosario=model.score(Vars_Indep,Var_Dep) coef_Deter_Ma_Del_Rosario</pre>
0.9869989145568099
<div><div>+ Code</div><div>+ Markdown</div></div>
<pre>coef_Correl_Ma_Del_Rosario=np.sqrt(coef_Deter_Ma_Del_Rosario) coef_Correl_Ma_Del_Rosario</pre>
np.float64(0.9934781902773758)

<pre>coef_Deter_Nicolas=model.score(Vars_Indep,Var_Dep) coef_Deter_Nicolas</pre>
0.7261693654875385
<pre>coef_Correl_Nicolas=np.sqrt(coef_Deter_Nicolas) coef_Correl_Nicolas</pre>
np.float64(0.8521557166900534)

<pre>coef_Deter_Rene=model.score(Vars_Indep,Var_Dep) coef_Deter_Rene</pre>
0.8347855215948432
<pre>coef_Correl_Rene=np.sqrt(coef_Deter_Rene) coef_Correl_Rene</pre>
np.float64(0.9136659792259112)

<pre>coef_Deter_Sergio = model.score(Vars_Indep,Var_Dep) coef_Deter_Sergio</pre>
✓ 0.0s
0.9422953208959974
<pre>#Corroboramos cual es el coeficiente de Correlación de nuestro modelo coef_Correl_Sergio=np.sqrt(coef_Deter_Sergio) coef_Correl_Sergio</pre>
✓ 0.0s
np.float64(0.9707189711219192)

EJEMPLO DE USO DEL MODELO



LEONARDO

Predicciones	tiempo de interacción
3.321580	10.558070
3.739423	1.249828
2.300622	1.866516
1.904927	10.099820
2.207986	17.817100
...	...
12.113270	3.650664
13.229866	3.916507
99.058414	99.000000
99.058414	99.000000
99.058414	99.000000

MA DEL ROSARIO

Predicciones	tiempo de interacción
8.802128	19.303690
6.766576	6.267806
6.454829	5.766922
7.191890	7.450372
6.880142	7.564992
...	...
4.847196	2.782771
4.535448	2.316111
5.372903	3.616160
98.987395	99.000000
98.987395	99.000000

NICOLAS

Predicciones	tiempo de interacción
8.608516	5.399169
9.862741	1.283400
10.763358	2.700226
9.871684	3.050262
10.233806	4.750256
...	...
99.014899	99.000000
99.014899	99.000000
10.754415	0.983153
99.014899	99.000000
99.014899	99.000000

EJEMPLO DE USO DEL MODELO

RENE

Predicciones	tiempo de interacción
29.291442	29.307090
34.206405	27.907880
29.107530	19.051010
26.870221	16.306860
24.632365	9.367180
29.547327	9.383972
28.740254	10.959300
27.933180	11.343060
27.126106	12.142600
26.319032	12.859410
25.511959	19.192930
16.121831	19.326270
15.314757	19.943110
24.521520	11.981560
22.283664	8.197347
18.615572	11.847620
17.808498	13.031260

SERGIO

Predicciones	tiempo de interacción
4.156505	2.946720
4.852990	2.066744
5.807019	2.600328
6.503505	3.249712
98.981492	99.000000
...	...
5.807019	3.250430
6.503505	3.315942
7.371686	2.516372
98.981492	99.000000
98.981492	99.000000

CONCLUSIÓN

DESPUÉS DE REALIZAR EL ANÁLISIS Y GENERAR LAS GRÁFICAS CORRESPONDIENTES UTILIZANDO DISTINTAS VARIABLES COMO OBJETIVO, SE OBSERVARON CIERTAS INCONSISTENCIAS EN LOS RESULTADOS. ESTAS INCONGRUENCIAS SUGIEREN QUE EL MODELO ACTUAL NO TIENE LA SUFICIENTE INFORMACIÓN PARA IDENTIFICAR PATRONES CONFIABLES O RELACIONES IMPORTANTES ENTRE LAS VARIABLES. UNA DE LAS PRINCIPALES LIMITACIONES DETECTADAS FUE LA CANTIDAD REDUCIDA DE DATOS DISPONIBLES, LO CUAL PERJUDICA DIRECTAMENTE EL MODELO EN CUESTIÓN, LA PRECISIÓN DE LOS CÁLCULOS Y LA CALIDAD DE LAS VISUALIZACIONES. UN CONJUNTO DE DATOS PEQUEÑO GENERA RESULTADOS SESGADOS.

POR LO TANTO, CONCLUIMOS QUE ES IMPORTANTE TENER UNA BASE DE DATOS MÁS AMPLIA Y EQUILIBRADA PARA MEJORAR EL MODELO, OBTENER ANÁLISIS MÁS SÓLIDOS Y GENERAR VISUALIZACIONES QUE REFLEJEN EL COMPORTAMIENTO REAL DE LOS USUARIOS ESTUDIADOS. CON MÁS DATOS, NO SOLO SE MEJORARÍA LA CAPACIDAD PREDICTIVA DEL MODELO, SINO QUE TAMBIÉN SE PODRÍAN IDENTIFICAR CON MAYOR CLARIDAD LOS PATRONES DE INTERACCIÓN O LA RESPUESTA DE LOS PARTICIPANTES ANALIZADOS.

GRACIAS
GRACIAS