Universidad del Valle de Guatemala
Facultad de ingeniería

Laboratorio #3
Deep Learning

Fredy Velásquez 201011
Angel Higueros 20460

Guatemala 18 de agosto del 2023

Ejercicio 1

1. Haga un análisis exploratorio de los datos para entenderlos mejor, documente todos los análisis.

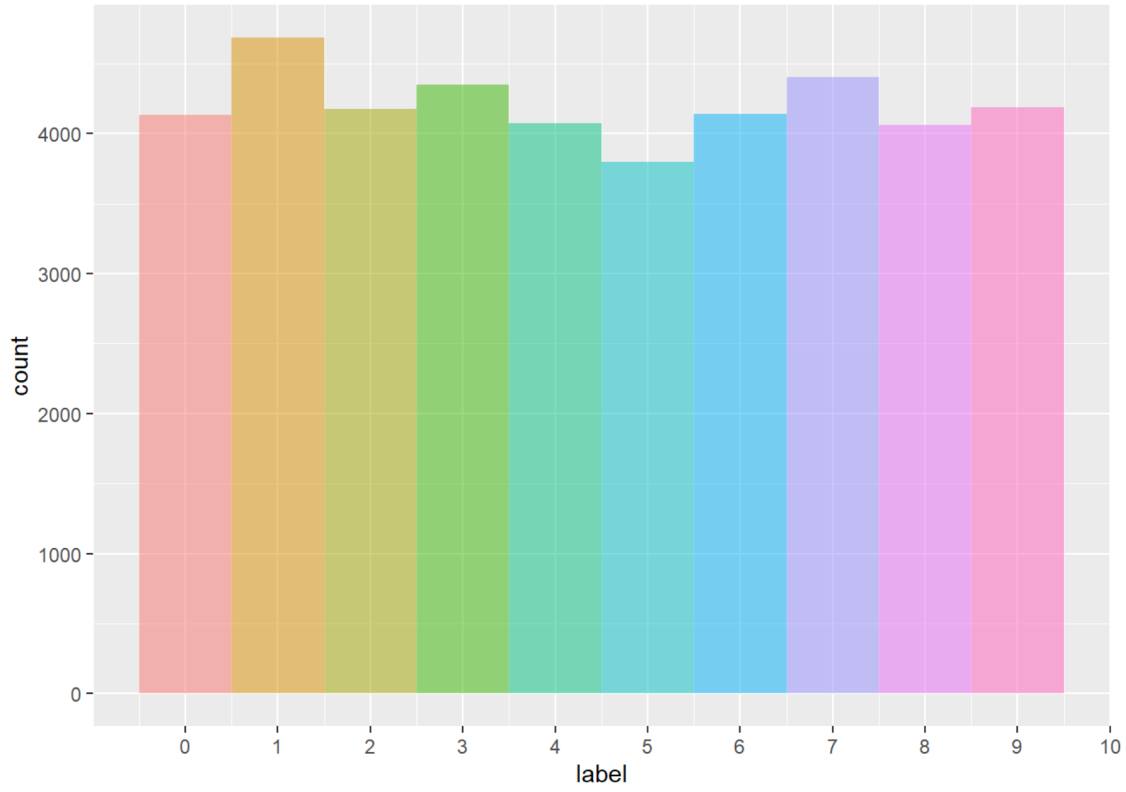
https://github.com/fredyvelasquezgt/Lab-3-DS

Para poder entrar en contexto con los datos primero veremos la cantidad de labels o dígitos proporcionados, con la intención de determinar qué dígito contiene más píxeles

```
table(db$label)
```

```
##
##    0    1    2    3    4    5    6    7    8    9
## 4132 4684 4177 4351 4072 3795 4137 4401 4063 4188
```

```
ggplot(data = db) + geom_histogram(aes(x=label,fill=factor(label)),bins=10, position = "stack",a
lpha = 0.5)+theme(legend.position="none")+ scale_x_continuous(breaks = seq(0, 100, 1.))
```



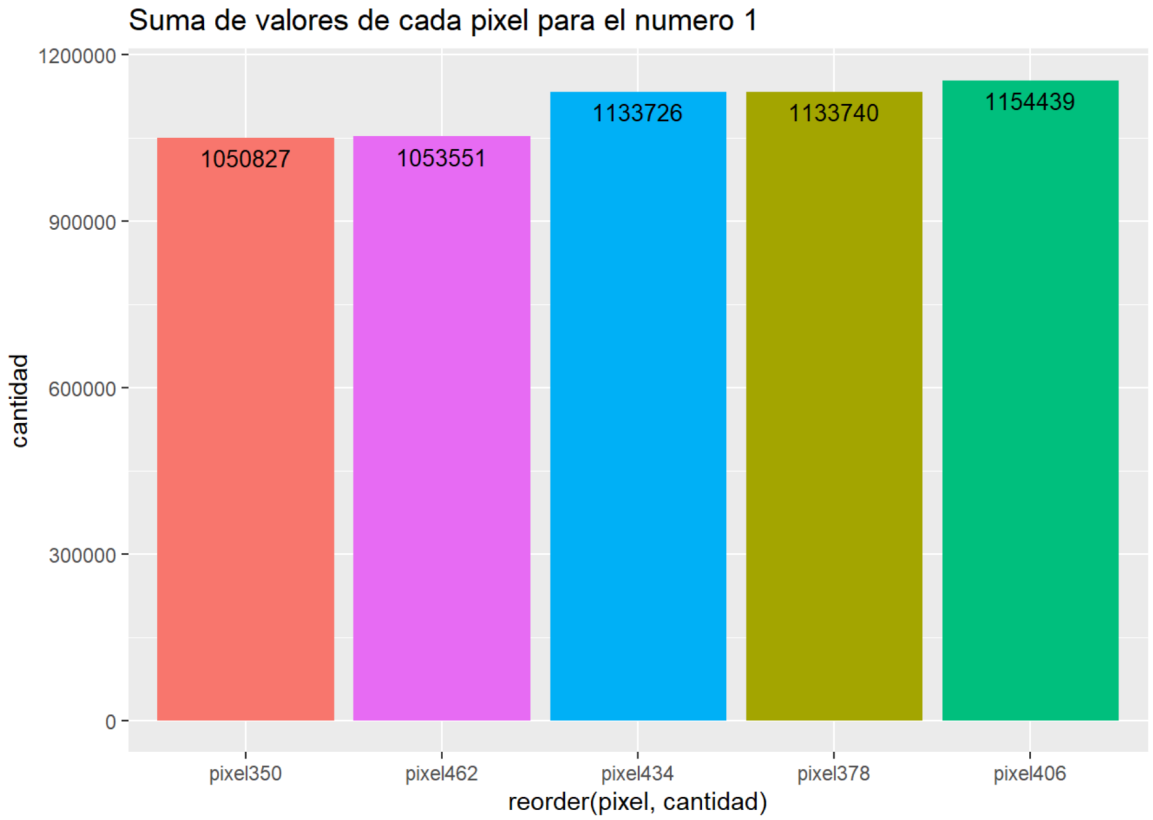
Tras observar gráficamente vemos que el número 1 es el que más filas tiene guardado datos, dando a entender que puede ser el numero mas dificil a predecir por ello es necesario tener más filas con datos de este,y el siguiente número es el 7, esto con la idea que estos números tienden a confundirse por ello fue necesario más iteraciones de prueba para posterior almacenar su resultado en este dataset.

```
number1<- subset(db, label == 1)
#Save all one's in new dataframe
dt<-colSums(number1[,-1])


dtF<-data.frame (pixel  = c(colnames(number1[,-1])),
                 cantidad = c(dt)
                 )

dtF2 <- dtF[order(-dtF$cantidad),]
dtF2<-head(dtF2,n=5)


ggplot(data=dtF2, aes(x=reorder(pixel,cantidad),y=cantidad, fill=pixel)) +
  geom_bar(stat="identity", position=position_dodge())+
  geom_text(aes(label=cantidad), vjust=1.6, color="black",
            position = position_dodge(0.9), size=3.5)+
  labs(title="Suma de valores de cada pixel para el numero 1 ")+
  theme(legend.position="none")
```



Tal como se observa en la grafica anterior, se puede observar que el pixel numero 406 del numero 1 presenta mas cambios de color, o por otro lado es un pixel que en su mayoria esta pintado o tiene un valor del color. Análisis del pixel 406.

```
number1Group<-subset(db,db$label==1)
summary(as.factor(number1Group$pixel406))
```

```
##     253     254     255     252     251     250     249       0     240     243
##    1680    1129     758     475      74      43      19      16      16      16
##     248     246     241     247     244     233     235     230     242     196
##      14      13      12      12      11      10      10       9       9       8
##     209     216     221     227     234     236     239     191     207     231
##       8       8       8       8       8       8       8       7       7       7
##     195     203     238     151     223     224     225     232     128     154
##       6       6       6       5       5       5       5       5       4       4
##     155     159     177     188     198     200     204     205     206     214
##       4       4       4       4       4       4       4       4       4       4
##     215     218     222     237     138     148     157     170     172     184
##       4       4       4       4       3       3       3       3       3       3
##     192     197     210     213     217     220     226     228     245      32
##       3       3       3       3       3       3       3       3       3       2
##      34      51      56      57      64      74     102     115     116     117
##       2       2       2       2       2       2       2       2       2       2
##     119     124     126     134     143     144     149     150     156     165
##       2       2       2       2       2       2       2       2       2       2
##     171     182     185     186     189     190     194     202     211 (Other)
##       2       2       2       2       2       2       2       2       2      61
```

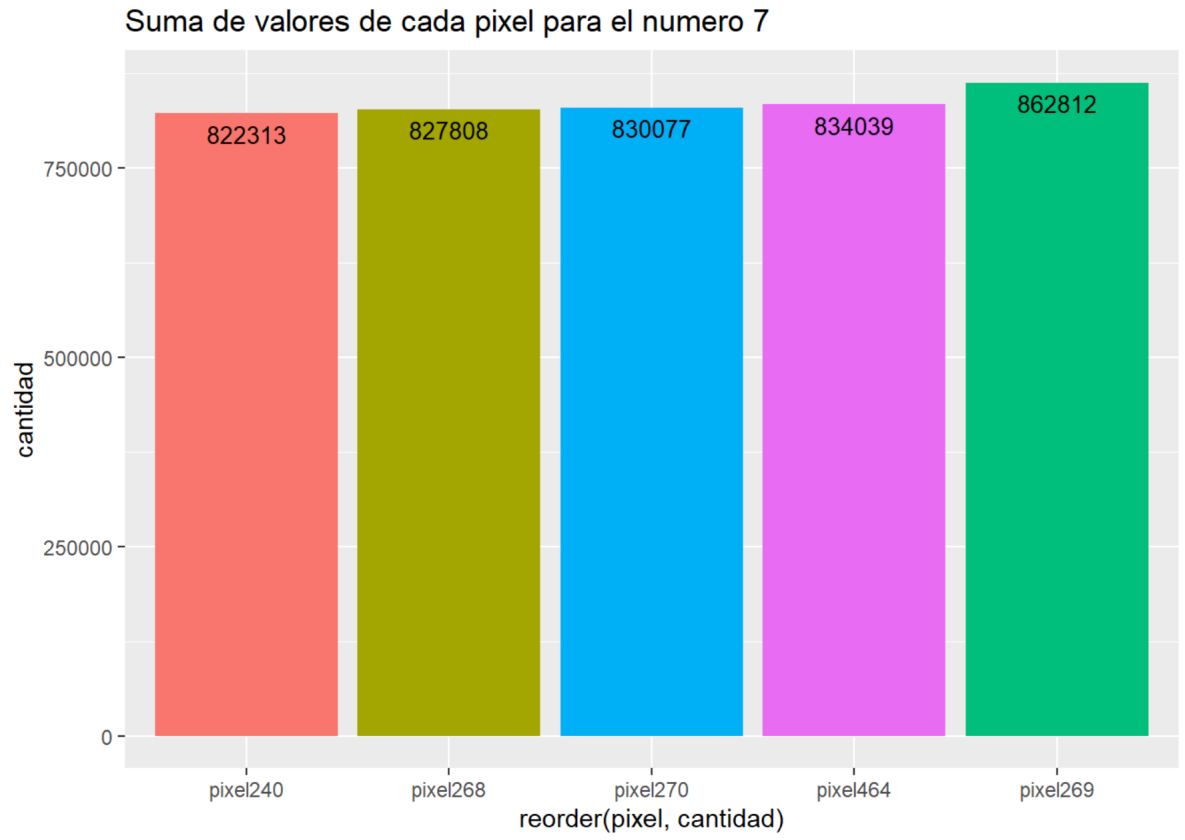
Como se logra observar anteriormente el pixel 406 tiene mas cambios de color, demostrando que efectivamente este pixel si cambia mucho el color de su relleno.

```
number7<- subset(db, label == 7)
#Save all one's in new dataframe
dt7<-colSums(number7[,-1])


dtF7<-data.frame (pixel  = c(colnames(number7[,-1])),
                  cantidad = c(dt7)
                  )

dtF27 <- dtF7[order(-dtF7$cantidad),]
dtF27<-head(dtF27,n=5)


ggplot(data=dtF27, aes(x=reorder(pixel,cantidad),y=cantidad, fill=pixel)) +
  geom_bar(stat="identity", position=position_dodge())+
  geom_text(aes(label=cantidad), vjust=1.6, color="black",
            position = position_dodge(0.9), size=3.5)+
  labs(title="Suma de valores de cada pixel para el numero 7 ")+
  theme(legend.position="none")
```



Tras observar los pixeles del número 7 se observó que en este caso el pixel con más color o más datos es el pixel 269, indicando que a pesar que el número 1 y 7 se parecen los pixeles con más color son distintos.

```
number7Group<-subset(db,db$label==7)
summary(as.factor(number7Group$pixel269))
```

```
##     253     252     254       0     255     251     250     234     217     235
##    1065     584     508     326     147      44      40      31      24      24
##     243     249     128     191     236     241     244     200     233     247
##      23      22      20      19      19      19      19      18      18      18
##     215     240     195     230     232     245     183     226     228     246
##      17      17      14      14      14      14      13      13      13      13
##     248      37      71      96     114     139     170     184     214     231
##      13      12      12      12      12      12      12      12      12      12
##     238     116     140     169     222     223      64      86     106     127
##      12      11      11      11      11      11      10      10      10      10
##     154     168     176     187     198     202     210     220     221     229
##      10      10      10      10      10      10      10      10      10      10
##     242      10     113     118     126     131     149     156     188     213
##      10       9       9       9       9       9       9       9       9       9
##     237      57      84     102     133     163     181     199     203     205
##       9       8       8       8       8       8       8       8       8       8
##     208     227     239       2      23      56      82     138     158     159
##       8       8       8       7       7       7       7       7       7       7
##     160     174     180     192     216     225       9      17      18 (Other)
##       7       7       7       7       7       7       6       6       6     638
```

Como se logra observar anteriormente el pixel 269 tiene mas cambios de color, demostrando que efectivamente este pixel si cambia mucho el color de su relleno. De hecho, se observa que este píxel de este número tiene más cambios que el pixel 406 del número 1.

2. Haga un modelo de redes neuronales simple, determine la efectividad del modelo.

	precision	recall	f1-score	support
0	0.90	0.96	0.93	2896
1	0.98	0.95	0.96	3314
2	0.94	0.87	0.90	2925
3	0.94	0.89	0.91	2982
4	0.93	0.95	0.94	2857
5	0.90	0.84	0.87	2663
6	0.95	0.96	0.95	2921
7	0.90	0.95	0.92	3075
8	0.81	0.89	0.85	2866
9	0.91	0.88	0.89	2901
accuracy			0.92	29400
macro avg	0.92	0.91	0.91	29400
weighted avg	0.92	0.92	0.92	29400

Se tuvo una precisión media de 0.92 indicando que fue alta y sin overfitting.

3. Haga un modelo de Deep learning, determine la efectividad del modelo.

```
Epoch 1/20
500/500 [==============================] - 7s 13ms/step - loss: 1.1246 - accuracy: 0.6811
Epoch 2/20
500/500 [==============================] - 6s 12ms/step - loss: 0.1765 - accuracy: 0.9439
Epoch 10/20
500/500 [==============================] - 6s 12ms/step - loss: 0.1802 - accuracy: 0.9446
Epoch 11/20
500/500 [==============================] - 6s 12ms/step - loss: 0.1721 - accuracy: 0.9466
Epoch 12/20
500/500 [==============================] - 7s 13ms/step - loss: 0.1672 - accuracy: 0.9459
Epoch 13/20
500/500 [==============================] - 7s 13ms/step - loss: 0.1466 - accuracy: 0.9566
Epoch 14/20
500/500 [==============================] - 6s 13ms/step - loss: 0.1481 - accuracy: 0.9545
Epoch 15/20
500/500 [==============================] - 6s 13ms/step - loss: 0.1428 - accuracy: 0.9567
Epoch 16/20
500/500 [==============================] - 6s 12ms/step - loss: 0.1470 - accuracy: 0.9554
Epoch 17/20
500/500 [==============================] - 6s 12ms/step - loss: 0.1396 - accuracy: 0.9566
Epoch 18/20
500/500 [==============================] - 6s 12ms/step - loss: 0.1385 - accuracy: 0.9600
Epoch 19/20
500/500 [==============================] - 6s 12ms/step - loss: 0.1370 - accuracy: 0.9601
Epoch 20/20
500/500 [==============================] - 6s 12ms/step - loss: 0.1234 - accuracy: 0.9635
```

Este modelo tuvo mejores resultados dado que la red neuronal fue entrenada, siendo la precisión de 0.96 siendo mejor que el modelo de la red neuronal simple.

4. Haga un modelo con cualquier otro algoritmo que el grupo seleccione, determine la efectividad del modelo. Puede basarse en los modelos que han sido probados con el data set que pueden encontrar en el siguiente link: http://yann.lecun.com/exdb/mnist/

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1    2    3    4    5    6    7    8    9
         0 1005    0    0    0    0 3491    0    0    0    0
         1    0 1151    0    0    0    0    0    0    0    0
         2    0    0 1075    0    0    0    0    0    0    0
         3    0    0    0 1066    0    0    0    0    0    0
         4    0    0    0    0 1027    0    0    0    0    0
         5    0    0    0    0    0  992    0    0    0    0
         6    0    0    0    0    0    0 1044    0    0    0
         7    0    0    0    0    0    0    0 1102    0    0
         8    0    0    0    0    0    0    0    0 1015    0
         9    0    0    0    0    0    0    0    0    0 1032

Overall Statistics

               Accuracy : 0.7506
                 95% CI : (0.7434, 0.7578)
    No Information Rate : 0.3202
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.7254

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: 0 Class: 1 Class: 2 Class: 3 Class: 4
Sensitivity           1.00000  1.00000  1.00000  1.00000  1.00000
Specificity           0.73136  1.00000  1.00000  1.00000  1.00000
Pos Pred Value        0.22353  1.00000  1.00000  1.00000  1.00000
Neg Pred Value        1.00000  1.00000  1.00000  1.00000  1.00000
Prevalence            0.07179  0.08221  0.07679  0.07614  0.07336
Detection Rate        0.07179  0.08221  0.07679  0.07614  0.07336
Detection Prevalence  0.32114  0.08221  0.07679  0.07614  0.07336
Balanced Accuracy     0.86568  1.00000  1.00000  1.00000  1.00000
                     Class: 5 Class: 6 Class: 7 Class: 8 Class: 9
Sensitivity           0.22128  1.00000  1.00000   1.0000  1.00000
Specificity           1.00000  1.00000  1.00000   1.0000  1.00000
Pos Pred Value        1.00000  1.00000  1.00000   1.0000  1.00000
Neg Pred Value        0.73163  1.00000  1.00000   1.0000  1.00000
Prevalence            0.32021  0.07457  0.07871   0.0725  0.07371
Detection Rate        0.07086  0.07457  0.07871   0.0725  0.07371
Detection Prevalence  0.07086  0.07457  0.07871   0.0725  0.07371
Balanced Accuracy     0.61064  1.00000  1.00000   1.0000  1.00000
```

La precisión del modelo es de 0.75, siendo mejor a comparación que los otros modelos.

5. Pruebe el mejor modelo ingresando imágenes de dígitos hechos a mano por los integrantes del grupo. Discuta el desempeño de su modelo y los resultados.

```
[[0.11880005151033401, 1.2444871572370175e-05, 0.29038935899734497, 0.012404449284076 69, 0.005807082634419203
 0.010891187936067581, 0.005521070212125778, 6.723721162416041e-05, 0.5507513284683228, 0.005355861037969589]]

8
El numero a predicho es:  8
```

La evidencia demuestra que el programa creado con deep learning fue el más efectivo. Se predijo que el número era el 8 con una probabilidad de 0.5 mayor respecto a los demás.

6. Haga un informe donde incluya el análisis exploratorio, la descripción de los modelos, la efectividad de cada uno y la comparación entre ellos.

Modelo	Precisión
RN Simple	0.92
Deep learning	0.96
KNN	0.75

Como se observa el modelo deep learning demostró que tiene mejor precisión que otros modelos, sobre todo por su entrenamiento.

Ejercicio 2

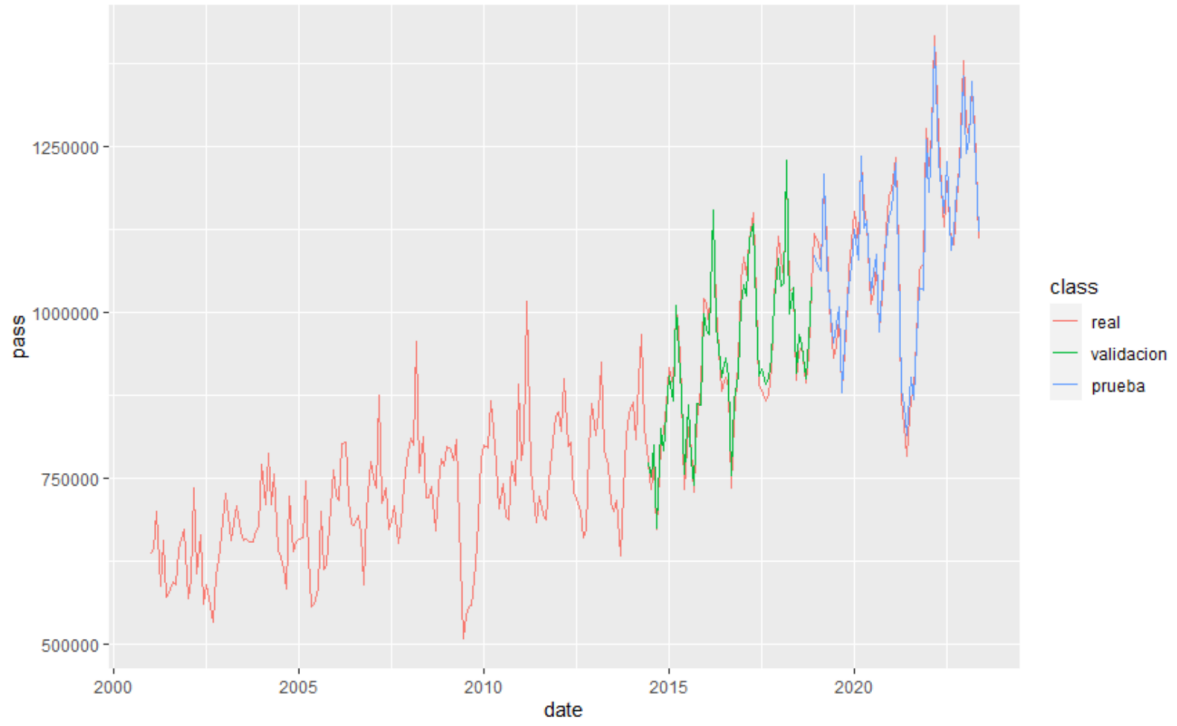
1. Utilice los conjuntos de entrenamiento y prueba de una de las series que utilizó en el Laboratorio 2.

https://github.com/fredyvelasquezgt/Lab-3-DS/blob/main/Ejercicio2.R

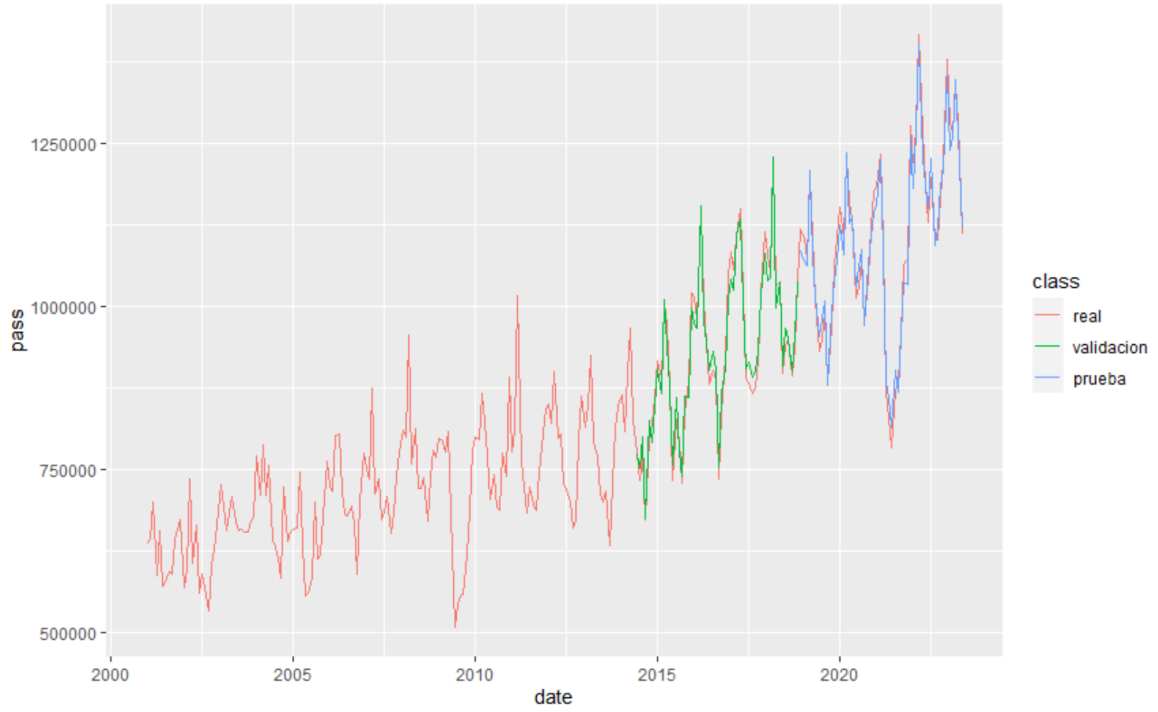
2. Haga al menos 2 modelos con configuraciones diferentes usando LSTM. Úselos para predecir.

El modelo que predijo mejor fue el modelo 2, aunque en la gráfica no se demuestra mucha diferencia entre cada uno de los modelos, el modelo 2 resultó mejor para predecir debido a que toma más detalles en cuenta, es más complejo.

Modelo 1



Modelo 2



3. ¿Cuál predijo mejor?¿Son mejores que los modelos creados en el laboratorio pasado?¿Cómo lo determinaron?

Finalmente, llegamos a la conclusión de que estos modelos son más efectivos en predicción en comparación con los modelos ARIMA. Esto se debe a que los modelos basados en LSTM son capaces de identificar patrones de mayor complejidad. En esta instancia, estamos examinando el patrón de consumo de gasolina diésel, y según los análisis de las representaciones gráficas previas, podemos afirmar que este patrón exhibe una naturaleza compleja debido a los diversos eventos que han tenido lugar en los últimos años.

Además, a diferencia de los modelos ARIMA, los modelos LSTM no dependen de la estacionariedad de la serie temporal para su desempeño. Esta conclusión también ha sido respaldada por la comparación de las representaciones gráficas. Las visualizaciones de los modelos ARIMA mostraron menos nivel de detalle y sus predicciones resultaron más limitadas. Por el contrario, los modelos LSTM ofrecieron una representación más natural y realista del comportamiento de la serie temporal. Estos modelos también tomaron en consideración los valores extremos que se presentaron a lo largo de los años en el transcurso de sus predicciones.

Modelos LSTM:

https://github.com/fredyvelasquezgt/Lab-3-DS/blob/main/Informe_Ejercicio_2.Rmd