

Universidad Del Valle de Guatemala
Data Science
Sección 10
Ciclo 2 2022



Laboratorio 3

Reconocimiento de caracteres manuscritos

Marco Ramírez #19588
Estuardo Hernández #19202

Contenido

Ejercicio 1	3
Análisis exploratorio	3
Haga un modelo de redes neuronales simple, determine la efectividad del modelo	3
Haga un modelo de Deep learning, determine la efectividad del modelo	4
Haga un modelo con cualquier otro algoritmo que el grupo seleccione, determine la efectividad del modelo (KNN)	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.	6
Resumen	6
Ejercicio 2	7
Utilice los conjuntos de entrenamiento y prueba de una de las series que utilizó en el Laboratorio 2. Haga al menos 2 modelos con configuraciones diferentes usando LSTM. Úselos para predecir.	7
¿Cuál predijo mejor? ¿Son mejores que los modelos arima del laboratorio pasado? ¿Cómo lo determinaron? Agregue los resultados de este ejercicio al informe del ejercicio 1.	7

Ejercicio 1

1. Análisis exploratorio

https://github.com/MarcoRamirezGT/Lab3_DataScience/blob/main/Analisis%20Exploratorio%20Ejercicio1.pdf

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

El código de este ejercicio se encuentra en el siguiente [link](#)

	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

Como se puede observar la precisión de este modelo tuvo una precisión media de 0.92 indicando que fue alta y sin indicios de overfitting. Por ende fue un modelo.

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

El código de este ejercicio se encuentra en el siguiente [link](#):

```
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
```

Como se observa , este modelo tuvo mejores resultados, debido a que la red neuronal fue entrenada, y se utilizó deep learning. Por ende se obtuvo una precisión de 0.96, siendo mejor que el modelo de red neuronal simple.

4. Haga un modelo con cualquier otro algoritmo que el grupo seleccione, determine la efectividad del modelo (KNN)

El código de este ejercicio se encuentra en el siguiente [link](#):

```

Confusion Matrix (Query Label vs Predicted)
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

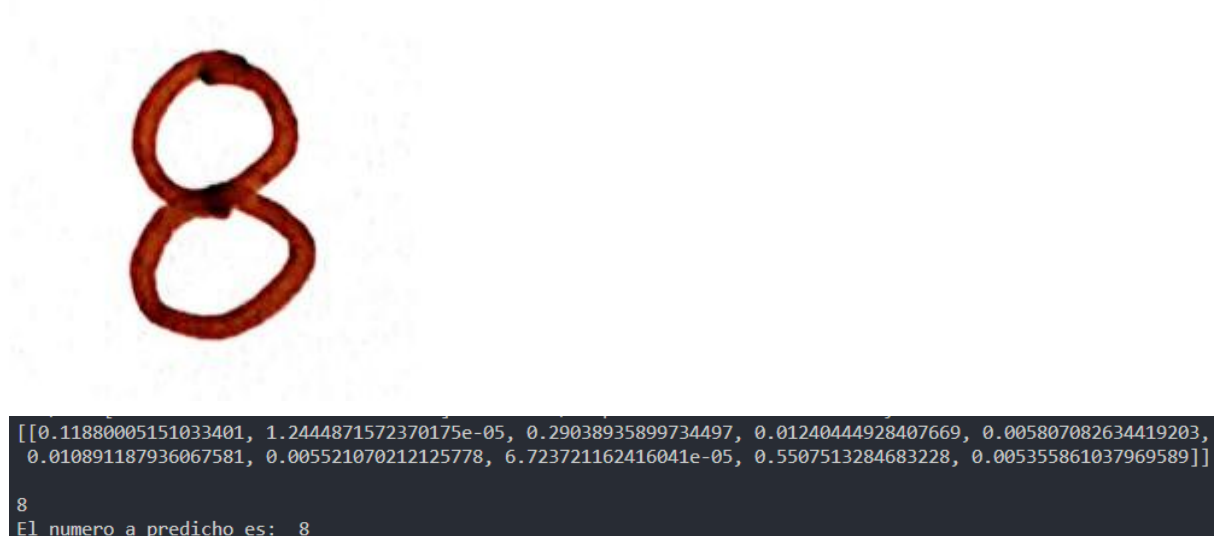
```

Tal como se observa, la precisión del modelo es de 0.75, siendo este el porcentaje menor a comparación de los otros modelos. Teniendo como mejor modelo el de deep learning.

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.

Prueba del modelo en el siguiente [link](#):

Tras poner a prueba la siguiente imagen



El programa creado con deep learning, fue capaz de acertar el número que tenía la imagen, demostrando que el modelo creado tiene una alta precisión. Dando a entender que el deep learning tiene un gran potencial y sobre todo no puede ayudar en muchos temas. En este caso predijo el que era el número 8 con una probabilidad de 0.5, mayor al resto de números.

6. Resumen

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

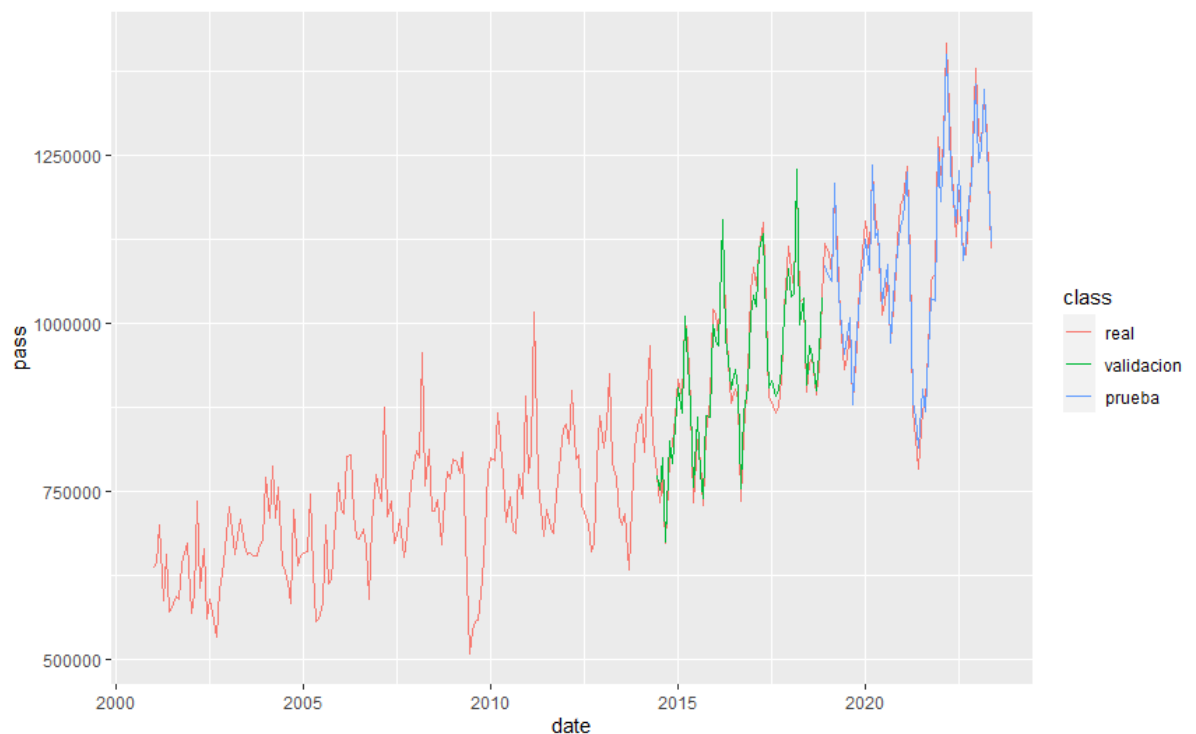
7. Utilice los conjuntos de entrenamiento y prueba de una de las series que utilizó en el Laboratorio 2. Haga al menos 2 modelos con configuraciones diferentes usando LSTM. Úselos para predecir.

El código de este ejercicio se encuentra en el siguiente [link](#).

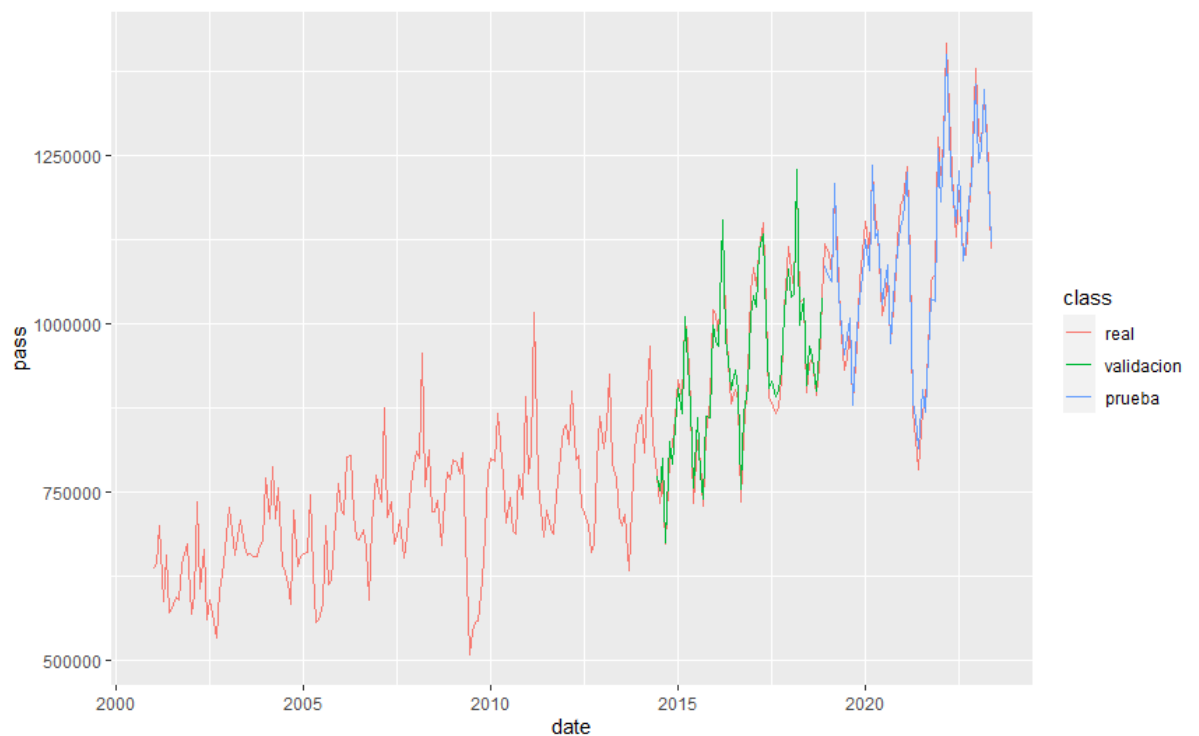
8. ¿Cuál predijo mejor? ¿Son mejores que los modelos arima del laboratorio pasado? ¿Cómo lo determinaron? Agregue los resultados de este ejercicio al informe del ejercicio 1.

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



Finalmente, concluimos que estos modelos son mejores para predecir que los modelos ARIMA porque los modelos basados en LSTM detectan patrones complejos, y en este caso se está analizando el consumo de la gasolina diesel y por las gráficas anteriores decimos que tiene un comportamiento complejo por todos los acontecimientos ocurridos durante los últimos años. Además, a diferencia de ARIMA, no se basa en la estacionariedad de la serie. También lo determinamos con la comparación de las gráficas, las gráficas de los modelos ARIMA tenían menos detalle y lo que predecían era menos, con los modelos LSTM se ve más natural o real el comportamiento de la serie y que toma en cuenta los puntos extremos ocurridos durante el paso de los años.

Más detalles de los modelos LSTM [aquí](#).