

Informe Proyecto 2 entrega 7

Juan Luis Solórzano (carnet: 201598)

Micaela Yataz (carnet: 18960)

2025-01-20

https://github.com/JusSolo/Mineria_Proyecto2.git

git: https://github.com/JusSolo/Mineria_Proyecto2.git

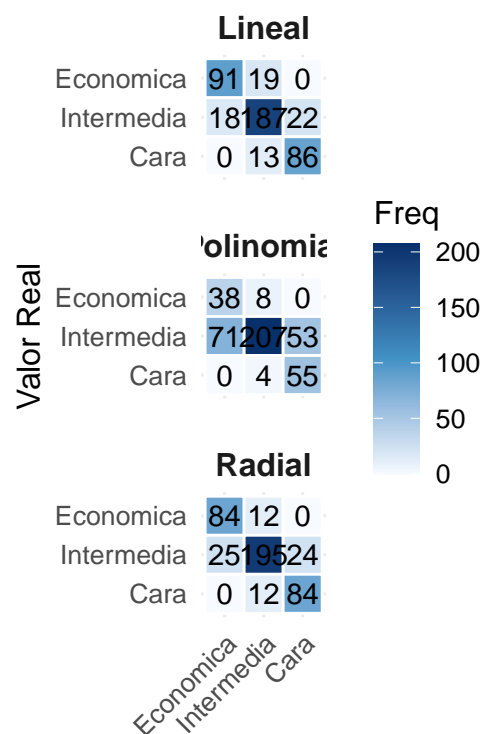
Introducción:

A lo largo del semestre hemos usado diferentes modelos de aprendizaje supervisado tanto en su versión de clasificación como de regresión. Para este último informe del proyecto 2 se pretende probar máquina de vectores de soporte con diferentes topologías para clasificar los precios de las casas en categorías y otras redes para predecir el precio de las mismas. Por otro lado se desea comparar todos los modelos anteriores.

Modelo de Clasificación con redes Neuronales

Comparacion de primeros 3 modelos con distintos kernels

Matrices de Confusión por Modelo



Observando las matrices de confusión se puede concluir que los 3 modelos se equivocan más clasificando la categoría de las casas intermedias. El modelo radial y lineal parecen ser los mejores, Siendo el radial más equilibrado y el lineal mejor en la clasificación entre las casas económicas e intermedias.

Table 1: Comparación de métricas macro-promediadas para los 3 modelos SVM usando datos de prueba

	Accuracy	Kappa	Sensitivity	Precision	F1
Linear	0.835	0.733	0.828	0.840	0.834
Radial	0.833	0.725	0.813	0.850	0.828
Polynomial	0.688	0.442	0.601	0.795	0.634

Table 2: Comparación de métricas macro-promediadas para los 3 modelos SVM usando datos de entrenamiento

	Accuracy	Kappa	Sensitivity	Precision	F1
Linear	0.890	0.822	0.882	0.895	0.888
Radial	0.929	0.885	0.922	0.935	0.928
Polynomial	0.800	0.651	0.734	0.900	0.776

Comparado ambas tablas el único modelo que parece sobre ajustado es el polynomial, pues es que tiene mayores diferencias en las métricas de desempeño con los datos de prueba y entrenamiento.

Modelo ajustado

Entre los 3 modelos anteriores se decidió ajustar el modelo radial, pues es más flexible que el lineal y fue un poco pero que el lineal. Posiblemente al ajustarlo su desempeño mejore.

Mejores hiperparámetros:

```
##      sigma      C
## 21 0.001 100
```

Table 3: Métricas macro-promediadas del modelo SVM Radial tuneado (entrenamiento vs prueba)

	Accuracy	Kappa	Sensitivity	Precision	F1
Tuned Radial (Train)	0.902	0.842	0.892	0.911	0.901
Tuned Radial (Test)	0.823	0.713	0.813	0.831	0.821

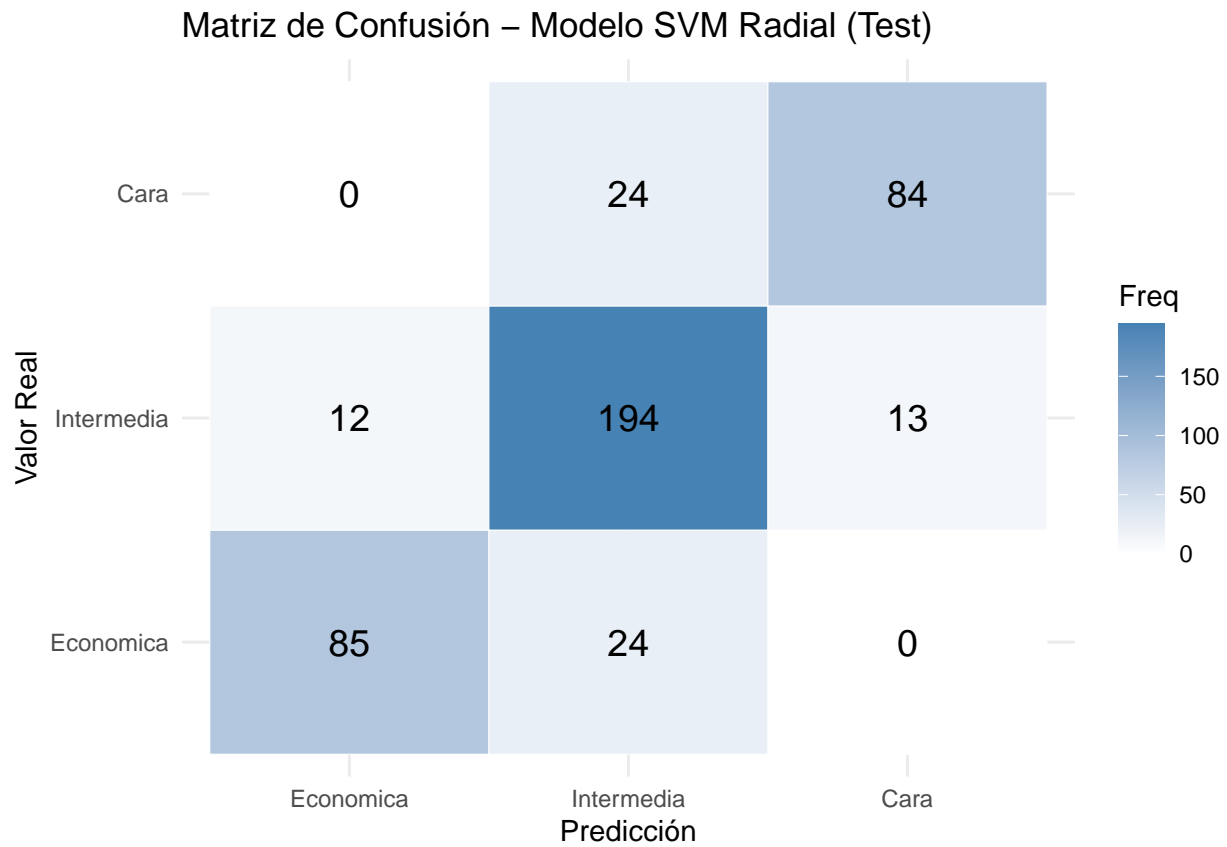
El modelo tuneado tiene claramente sobre ajuste y es peor, por lo que vamos a volver a ajustarlo pero separando los datos de entrenamiento en 2 entrenamiento y validación para evitar sobre ajuste.

Mejores hiperparámetros:

```
##      sigma      C
## 3 0.001 100
```

Table 4: Métricas del modelo SVM Radial tuneado (con train)

	Accuracy	Kappa	Sensitivity	Precision	F1
Validación (subtrain)	0.896	0.830	0.883	0.907	0.894
Prueba (test final)	0.833	0.726	0.814	0.848	0.829

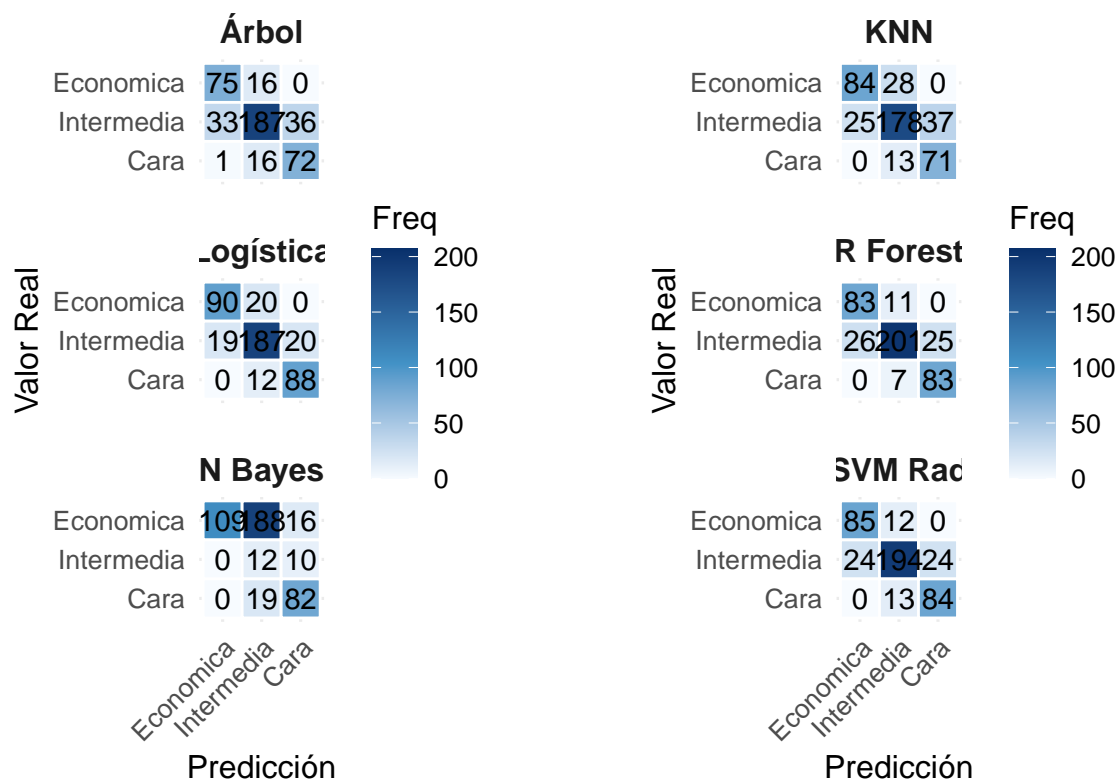


Viendo las métricas y las matrices de confusión y las métricas el modelo tuneado radial parece ser un poco mejor que el lineal, pero por muy poco.

Comparacion de todos los modelos de clasificacion usados a lo largo del proyecto

Matrices de confusión

Matrices de Confusión por Modelo Matrices de Confusión por Modelo



Observado las matrices de confucion se puede notar que naive bayes el el único modelo que no clasifica para nada bien. Los demás parecen mas o menos igual de buenos clasificando, observar las matrices de confucion es un poco engorroso.

Tablas comparando Espacio en memoria usada en ejecucion y tiempo de ejecucion

```
##
## Attaching package: 'pryr'
##
## The following objects are masked from 'package:purrr':
##
##   compose, partial
##
## The following object is masked from 'package:dplyr':
##
##   where
##
## Warning in microbenchmark(predict(modelo, newdata = datos), times = 10, : less
## accurate nanosecond times to avoid potential integer overflows
```

Table 5: Comparación de complejidad espacial y temporal de modelos ordenados por tiempo de ejecución

Modelo	Tiempo (ms)	Memoria (MB)	Precisión
Árbol de Decisión	0.596	0.08	0.766
Regresión Logística	2.065	1.65	0.837
SVM Radial	9.639	0.48	0.833
KNN	15.118	0.78	0.768
Random Forest	15.998	6.58	0.842
Naive Bayes	55.846	0.02	0.466

Table 6: Comparación de complejidad espacial y temporal de modelos ordenados por espacio en memoria

Modelo	Tiempo (ms)	Memoria (MB)	Precisión
Naive Bayes	55.846	0.02	0.466
Árbol de Decisión	0.596	0.08	0.766
SVM Radial	9.639	0.48	0.833
KNN	15.118	0.78	0.768
Regresión Logística	2.065	1.65	0.837
Random Forest	15.998	6.58	0.842

Table 7: Comparación de modelos ordenados por precisión

Modelo	Tiempo (ms)	Memoria (MB)	Precisión
Random Forest	15.998	6.58	0.842
Regresión Logística	2.065	1.65	0.837
SVM Radial	9.639	0.48	0.833
KNN	15.118	0.78	0.768
Árbol de Decisión	0.596	0.08	0.766
Naive Bayes	55.846	0.02	0.466

Arriba se puede observar la misma tabla pero ordenada según cada una de sus columnas. Como el SVM es el único modelo que aparece en el top 3 en las 3 tablas podemos considerar que en términos generales es el mejor modelo. Si se toma en cuenta el tiempo de ejecución y la precisión el mejor modelo es la regresión logística, pero es el segundo que ocupa más memoria. Para estos datos de la manera que fueron tratados el peor modelo es Naive Bayes, esto puede deberse a que algunas de las variables no son independientes (ir a ver análisis exploratorio informe 1). En conclusión exceptuando Naive Bayes todos los modelos funcionan. Si se necesitara un modelo muy rápido y que no ocupe mucha memoria (en la ejecución) el algoritmo a elegir sería El Árbol de decisión. Si se buscara la precisión a toda costa sería el Random Forest el elegido. Pero si se desea un equilibrio entre los 3 el modelo a elegir sería la regresión logística (si no importa mucho el espacio en memoria) y un SVM radial (si no importa que el tiempo de ejecución sea más lento). En conclusión no hay un modelo que sea el mejor en todo. El que se elija usar debería ser el que se pueda ejecutar en el tiempo y espacio que se requiere con la precisión más alta. Para tomar esa decisión podría ser útil usar alguna función de costo que pondere las métricas y elegir el modelo que lo minimice y cumpla con las restricciones.

Modelo de Regresion con redes Neuronales

Comparacion de todos los modelos de regresion usados a lo largo del proyecto