Aprenentatge computacional

Pràctica 1b

Profesores:
Pau Rodríguez
Jordi Gonzàlez

Alumnos:
Alejandro Garcia Carballo
1423957
Juan Plúa Gutiérrez
1358255

Índice

		Página
1.	Introducción	2
2.	Estudio preliminar	3
3.	Análisis e interpretación de los datos	4
4.	Dificultades	14
5.	Conclusiones	15
6.	Anexo	16

1. Introducción

Esta informe es la continuación de la primera práctica y en él se detalla todo el proceso de desarrollo que se ha seguido. Esta segunda parte se presentan varios problemas que resolveremos y comprenderemos aplicando modelos de clasificación numérica a partir de una discriminación de los datos y categorizarlos en clases.

En primer lugar, se aplicarán los modelos de *Regresión Logística* y de *Máquina de Vectores de Soporte* para ir comprobando cómo varía la precisión de cada uno de ellos y dónde se probarán diferentes parámetros para los kernels para poder encontrar la configuración del clasificador que de los mejores resultados.

En segundo lugar, una vez comprobados las diferentes maneras de aprendizaje de un clasificador se procederá a trabajar la presentación de los resultados del clasificador con otros de medidas de rendimiento mucho más completas. Para esto aplicaremos y cambiaremos los parámetros de aprendizaje del clasificador para encontrar una mejor configuración.

Por último, se visualizará el clasificador en un espacio de muestras donde podremos observar las diferentes categorías, la frontera de decisión representada por el clasificador, la distancia que hay entre cada punto y esta frontera, y, la probabilidad que hay en cada muestra que pertenezca a cada clase.

2. Estudio preliminar

En este apartado haremos un estudio del conjunto de datos que se nos ha proporcionado y cual es su significado.

La base de datos con la que trabajaremos contiene una serie de características que se les ha asignado a un conjunto de 1353 sitios web los cuales han sido sometidos a un estudio para saber si son páginas web fraudulentas. En este estudio se detectaron un total de 548 webs legítimos, 702 fraudulentos y 103 sospechosos. Cuando a un sitio web se le considera sospechoso significa que este puede ser fraudulento o legítimo, esto significa que el sitio web tiene algunas características legítimas y otras de fraudulentas.

Los datos los hemos separado en entradas, que serán las características de los sitios web, y la salida, que corresponderá al resultado del estudio, que determina si el sitio es legítimo, sospechoso o fraudulento.

A continuación mostramos una tabla con todos los atributos que contiene nuestra base de datos:

ATRIBUTOS		
ID	**Nombre**	**Descripción**
V1	id	Identificador
V2	SFH	Controlador de formularios anormales del servidor
V3	popUpWindow	Ventana emergente
V4	SSLfinal_State	Estado final de la capa de seguridad de los sockets
V5	Request_URL	URL solicitada
V6	URL_of_Anchor	URL de enlace
V7	web_traffic	Tráfico de la web
V8	URL_Length	Longitud de la URL
V9	page_of_domian	Página de dominio
V10	having_IP_Address	Posesión de una dirección IP
V11	Result	Resultado

Tabla 1: Tabla precisión logistic regression

3. Análisis e interpretación de los datos

Todos los datos utilizados, tanto los de entrada como los de salida, son categóricos. Tenemos un total de 1300 puntos de 9 dimensiones, donde cada dimensión puede tener los valores de 0, 1, -1. Por lo tanto, el espacio muestral de cualquier par de características serà el siguiente:

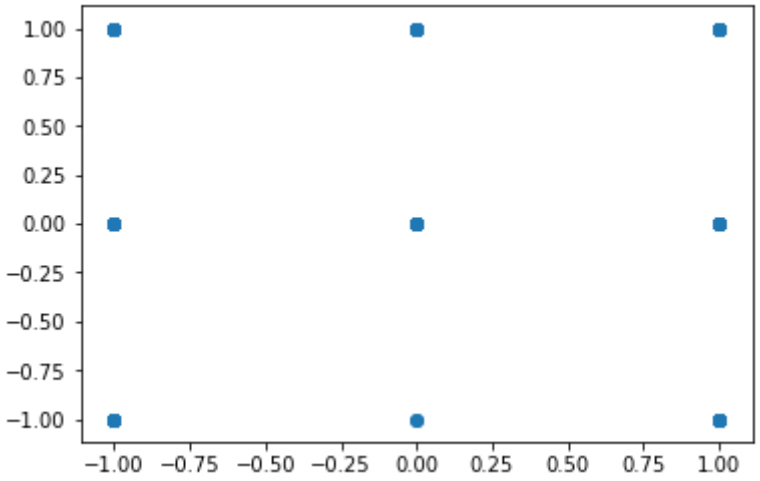

Ilustración 1: Espacio muestral

Quin és el millor model de classificació, en termes de precisió, i en el cas dels svm amb quin kernel i quins paràmetres del svm?

Para mostrar los datos obtenidos con distintos clasificadores y distintos métodos de entrenamiento utilizaremos tablas.

En la siguiente tabla mostraremos los resultados obtenidos en precisión obtenidos con un clasificador de logistic regression. En ella indicamos también los distintos métodos de selección de los datos de entrenamiento y de validación:

Método selección	Precisión (%)	C	fit_intercept	penalty	tol
Todas las muestras	83%	2.0	True	l2	0.000001
Básica					
50%(T) - 50%(V)	82%	1.5	True	l2	0.001
80%(T) - 20%(V)	78%	1.5	True	l2	0.001
70%(T) - 30%(V)	81%	1.5	True	l2	0.001
K-Fold					
K = 2	85%	2.0	True	l2	0.001
K = 3	86%	2.0	True	l2	0.001
K = 4	86%	2.0	True	l2	0.001
K = 5	86%	2.0	True	l2	0.001
K = 6	86%	2.0	True	l2	0.001
LOOCV	82%	2.0	True	l2	0.001

Tabla 2: Tabla precisión logistic regression

Como observamos en la Tabla 1, el método que nos está dando una precisión más alta utilizando regresión logística como clasificador es el K-Fold con K = 6 o K = 5.

A continuación haremos un proceso similar pero utilizando un clasificador SVM y teniendo en cuenta los distintos kernels que podemos utilizar:

Método selección	Precisión (%)	Kernel
Básica		
50%(T) - 50%(V)	85%	Linear
80%(T) - 20%(V)	84%	Linear
70%(T) - 30%(V)	85%	Linear
50%(T) - 50%(V)	59%	Poly
80%(T) - 20%(V)	61%	Poly
70%(T) - 30%(V)	57%	Poly
50%(T) - 50%(V)	58%	Rbf
80%(T) - 20%(V)	58%	Rbf
70%(T) - 30%(V)	57%	Rbf
K-Fold		
K = 2	84%	Linear
K = 3	86%	Linear
K = 4	86%	Linear
K = 5	86%	Linear
K = 6	85%	Linear
K = 2	59%	Poly
K = 3	60%	Poly
K = 4	60%	Poly
K = 5	60%	Poly
K = 6	60%	Poly
K = 2	59%	Rbf
K = 3	59%	Rbf
K = 4	59%	Rbf

K = 5	59%	Rbf
K = 6	59%	Rbf
LOOCV	82%	Linear

Tabla 3: Tabla precisión SVM

En la *tabla 3* podemos observar que el método de selección que nos está dando un valor más alto es el K-fold con K = 4 o K = 5, con un resultado muy cercano al que conseguimos haciendo una partición de 70% entrenamiento y 30% validación.

Analizando estas dos tablas, tenemos que tanto la regresión logística como el SVM nos están dando un valor de precisión igual, y que dentro del clasificador SVM, el kernel que está dando un mejor rendimiento es el linear.

Com influeix l'avaluació dels experiments en la interpretació dels resultats? Quin creieu que és el més fiable?

Creemos que, de los métodos de selección utilizados, el más fiable es el Leave one Out, ya que es la más precisa teniendo en cuenta que realizamos el proceso de clasificación para cada una de las muestras, tantas veces como muestras tengamos, haciendo un resultado medio de cada una de las iteraciones.

Quina informació complementària dóna cada una de les mesures de rendiment implementades?

Para poder tener diferentes medidas de rendimiento que puedan ser más completas, calcularemos el F1 score y haremos varias gráficas de Precision-Recall y ROC.

El F1 nos proporcionará una medida basada en precision, que es el número de correctos positivos que hemos obtenido en el resultado dividido entre el total de valores positivos que hemos obtenido del clasificador, y el recall, que es el número de resultados positivos correctos dividido por el número de todas las muestras relevantes (todas las muestras que deberían haberse identificado como positivas).

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

La curva ROC nos ayudará a saber con qué seguridad el clasificador decide clasificar si una muestra es de clase A o B.

En la *Tabla 4* mostramos cuáles han sido los resultados obtenidos de calcular el F1 score utilizando como clasificador la regresión logística, y probando con 3 tipos distintos de promedios que permite utilizar la función f1 score de sklearn (macro, micro y weighted):

Método selección	Average macro	Average micro	Average weighted
50%T - 50%V	0.56	0.81	0.78
70%T - 30%V	0.56	0.81	0.78
80%T - 20%V	0.57	0.84	0.81
Todos los datos	0.58	0.83	0.80
K-fold			
K = 2	0.56	0.81	0.78
K = 3	0.54	0.79	0.75
K = 4	0.55	0.79	0.76
K = 5	0.54	0.78	0.74
K = 6	0.56	0.82	0.78

Tabla 4: Tabla F1 score con regresión logística

Como podemos observar, el valor que nos está dando un resultado más alto (F1 es mejor cuanto más cercano a 1 sea), es el método de dividir los datos en un 80% de entrenamiento y un 20% de validación.

A continuación mostraremos las gráficas de precision-recall y ROC que hemos obtenido con distintos clasificadores y distintas configuraciones:

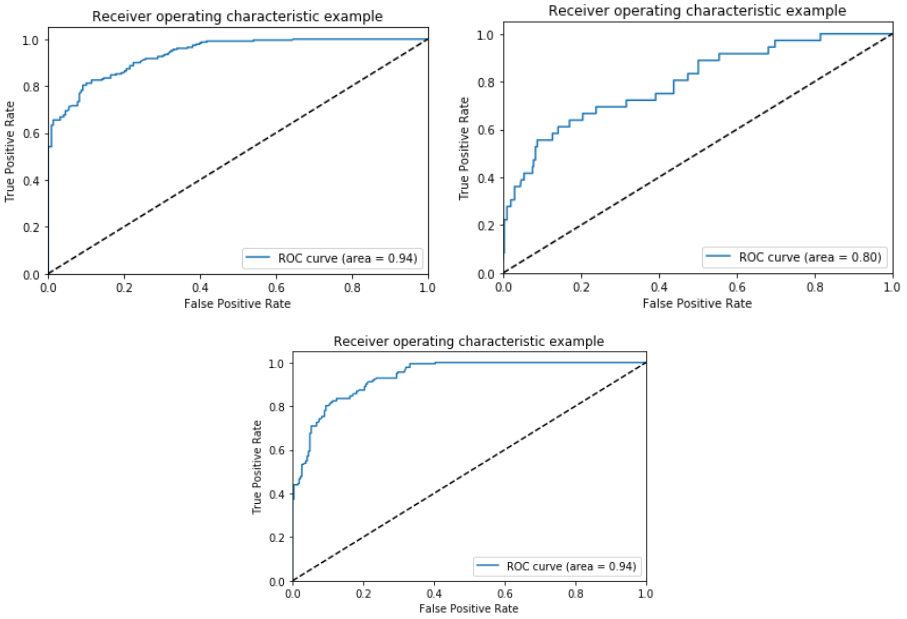

Ilustración 1: Gráficas ROC individuales para clase -1, 0 y 1

Estas tres gráficas son las curvas ROC que hemos obtenido utilizando un clasificador SVM con kernel linear. Como podemos observar, teniendo en cuenta que el eje Y son los True Positive, tenemos unos valores más correctos a la hora de predecir valores -1 y 1 que en 0. Esto seguramente es debido a que a la hora de entrenar nuestro modelo, lo hacemos de forma binaria, teniendo en cuenta que los valores sospechosos para nosotros serán phishing, ya que hemos pensado que es mejor en este caso tener valores falsos positivos que falsos negativos, ya que la información privada de los usuarios está en juego.

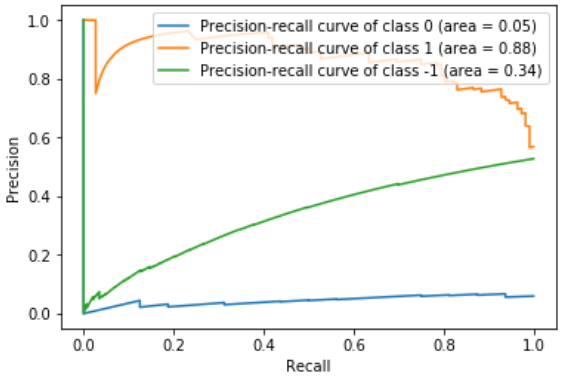

Ilustración 2: Precision Recall Logístic Regression L1

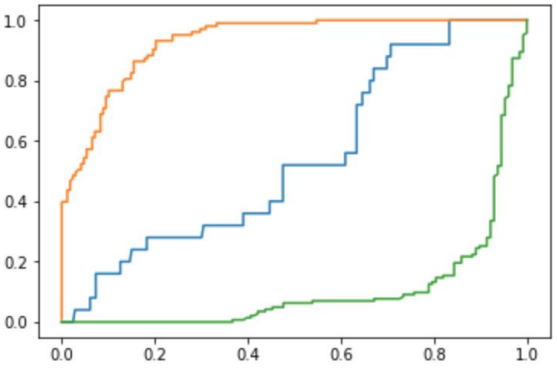

Ilustración 3: ROC Logístic Regression L1

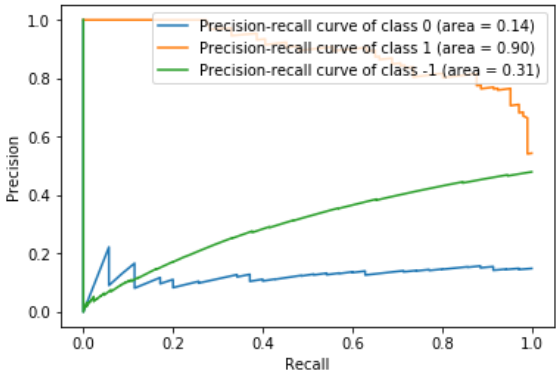

Ilustración 4: Precision Recall Logístic Regression L2

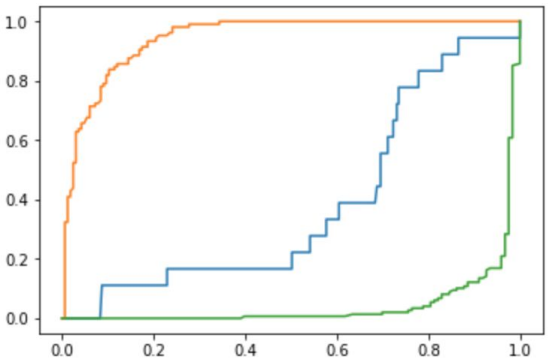

Ilustración 5: ROC L2

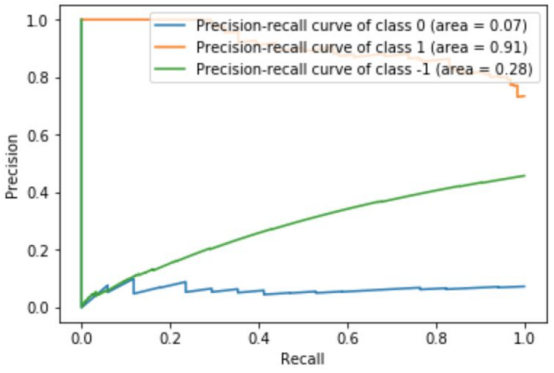

Ilustración 6: Precision Recall SVM Linear

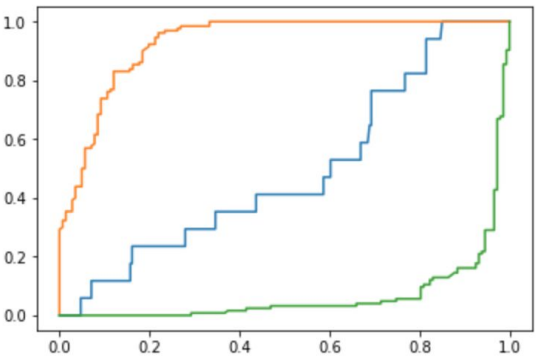

Ilustración 7: ROC SVM Linear

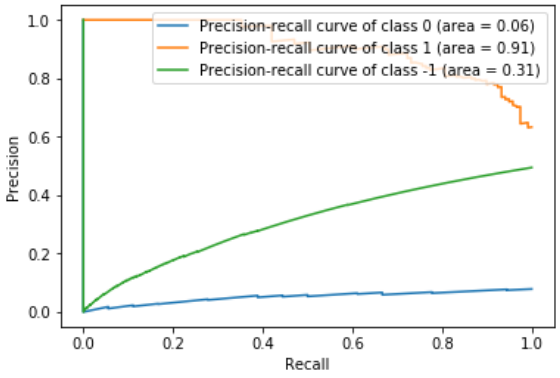

Ilustración 8: Precision Recall SVM RBF

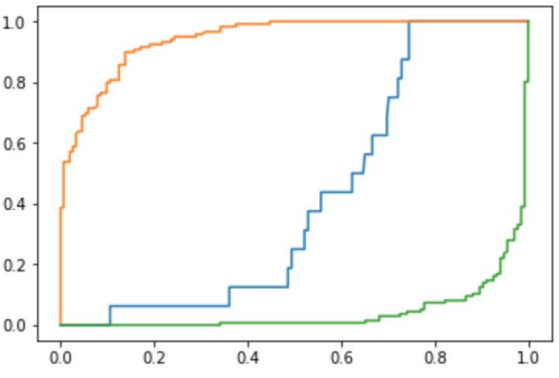

Ilustración 9: ROC SVM RBF

Quina és la mesura més complerta i quin significat té en el vostre problema?

La medida de precisión sería la adecuada para un modelo que le da mucha importancia a los falsos positivos, mientras que el recall es una medida que funcionará mejor en modelos que quieran dar importancia a los falsos negativos. De esta manera, el valor de F1 nos hace un balance entre la medida de precisión y la medida de recall. Por lo tanto, creemos que para nuestro problema, la medida de F1 score será la más completa.

Com de diferent és la configuració final del classificador tenint en compte aquestes noves mesures respecte al millor model que heu trobat utilitzant únicament la precisió?

Teniendo en cuenta estos datos, hemos obtenido como mejor método de clasificación con regresión logística el método de utilizar un 80% de los datos para entrenamiento y un 20% para validación, mientras que en un inicio, el método de K-Fold nos daba mejores resultados.

A continuación visualizamos los resultados de nuestro clasificador teniendo en cuenta que la salida será -1 o 1, sin tener en cuenta los valores 0. En un inicio, intentamos realizar un clasificador que detectara entre -1, 1 o 0 para saber cuáles eran sospechosos, pero la distribución de las muestras nos hizo dudar sobre si esto era correcto. Finalmente decidimos clasificar entre -1 o 1.

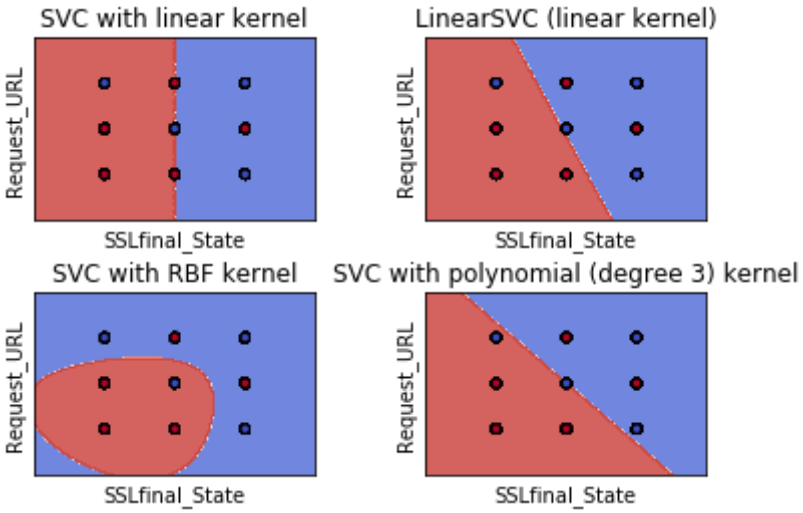

Ilustración 10: Visualización SVM con características Request URL y SSLfinal_State

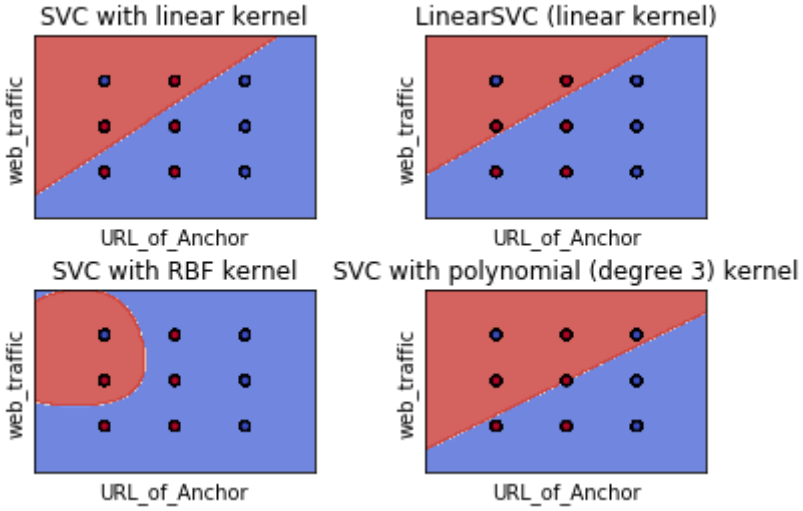

Ilustración 11: Visualización SVM con características web_traffic y URL_of_Anchor

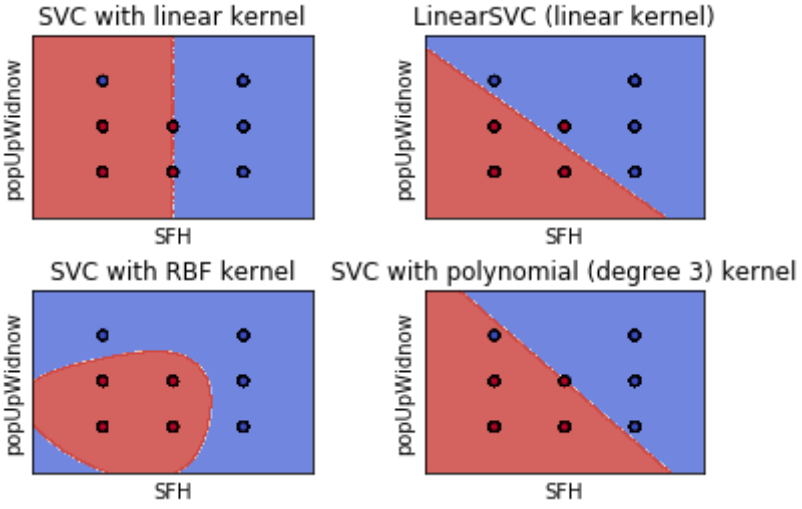

Ilustración 12: Visualización SVM con características popUpWindow y SFH

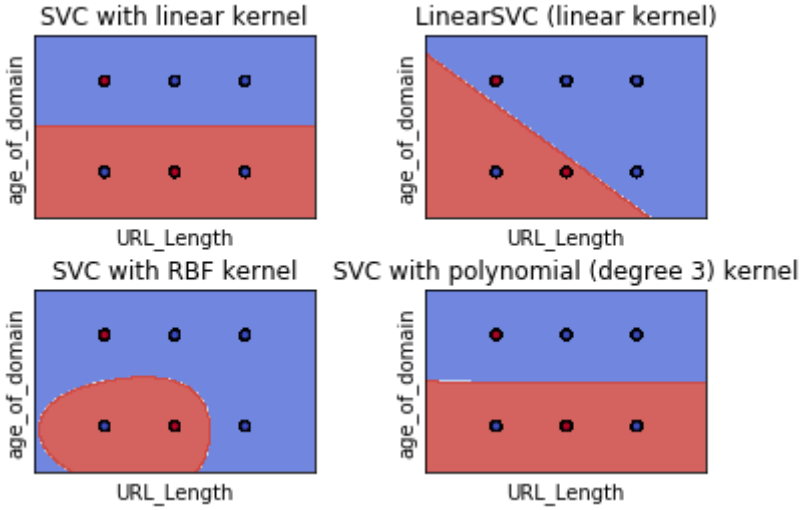

Ilustración 13: Visualización SVM con características age_of_domain y URL_Lenght

Hemos visualizado varios pares de características y en cada uno de esos pares hemos utilizado distintas configuraciones para el clasificador SVM. Hemos probado con distintos kernels para ver cómo influye la utilización de estos en el resultado de la clasificación.

4. Dificultades

Durante el proceso de realización de este proyecto hemos tenido bastantes dificultades a la hora de plantear cómo clasificar los datos. En un inicio usábamos para el conjunto de entrenamiento de salida 3 posibles valores (phishing, legítimo o sospechoso), pero esto nos daba dificultades y a la hora de visualizar los datos nos dimos cuenta que quizá era debido a la distribución de nuestras muestras. Finalmente, decidimos pedir ayuda en el foro y gracias a esta ayuda decidimos clasificar entre 1 o -1, sin tener en cuenta las muestras de 0 en el conjunto de entrenamiento de salida.

5. Conclusiones

En esta práctica hemos logrado hacer completar en gran parte todos los apartados aunque los resultados obtenidos en un principio no eran del todo precisos, pero después de cambiar el método de clasificación hemos la podido trabajar con otros para comprarlos y escoger la configuración del clasificación que nos ofrecía los mejores resultados.

Después de la primera parte, la experiencia de programar el código en los programas de lenguaje python en Jupyter Notebook ha resultado más familiar, sencillo y cómodo. Pero si hemos notado complicaciones a la hora de poder aplicar todos los conceptos dados en teoría en la práctica debido a su complejidad.

Como mejor modelo un clasificador teniendo en cuenta el F1 score de regresión logística que utiliza el método de selección de dividir los datos en un 80% de entrenamiento y un 20% en datos de validación con un F1 score de 0.84 utilizando un average macro. En cambio, si tomamos en cuenta la precisión nuestro mejor clasificador sería Logistic Regression conK-fold o bien SVM con K-fold.

Por último, consideramos que lo aprendido a lo largo de esta primera y segunda parte de la práctica ha sido muy positivo para la aplicación en problemas tanto de regresión como de clasificación con datos reales.

6. Anexo

En este apartado mostraremos los resultados que obtuvimos utilizando las mismas medidas que seguimos en cada clasificador presentadas en el tercer punto de este informe pero con la diferencia de que en los siguientes casos consideramos los datos sospechosos como una categoría más a tener en cuenta en nuestros resultados.

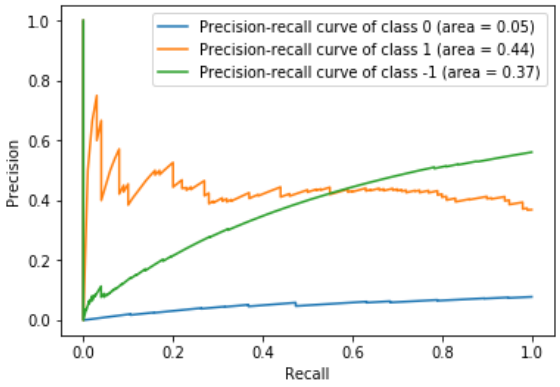

Ilustración 14: Precision Recall Logístic Regression L1

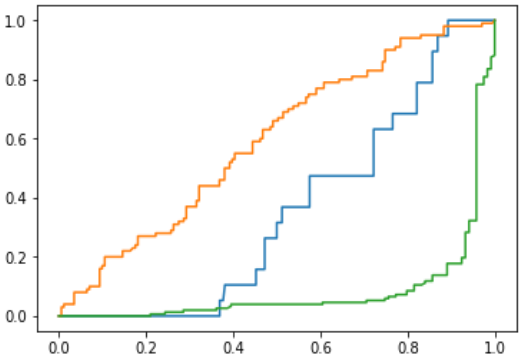

Ilustración 15: ROC Logístic Regression L1

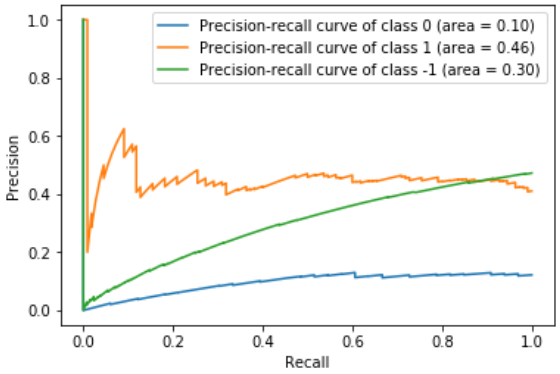

Ilustración 16: Precision Recall Logístic Regression L2

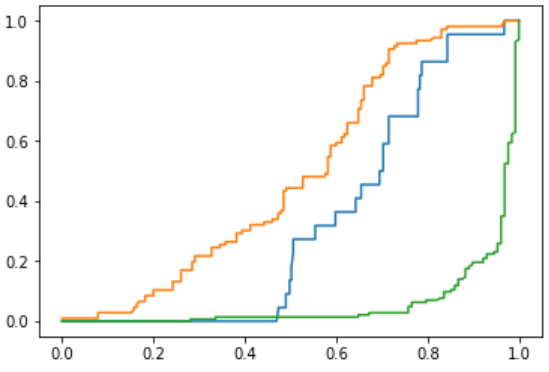

Ilustración 17: ROC L2

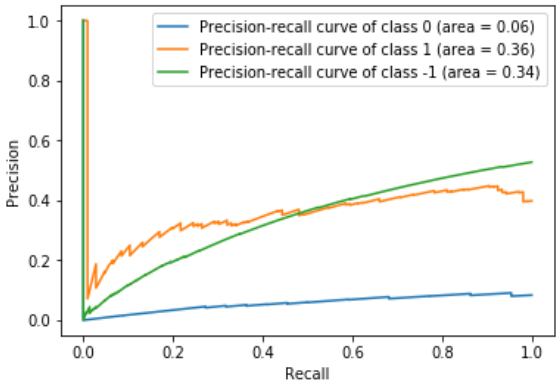

Ilustración 18: Precision Recall SVM Linear

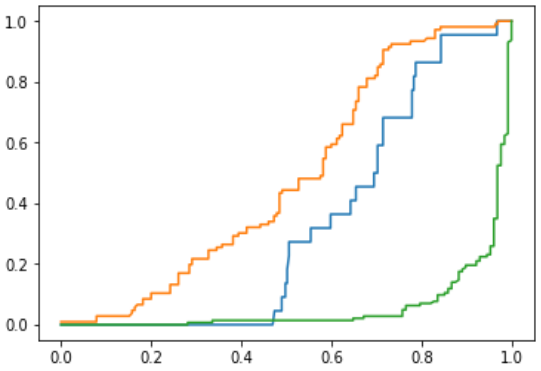

Ilustración 19: ROC SVM Linear

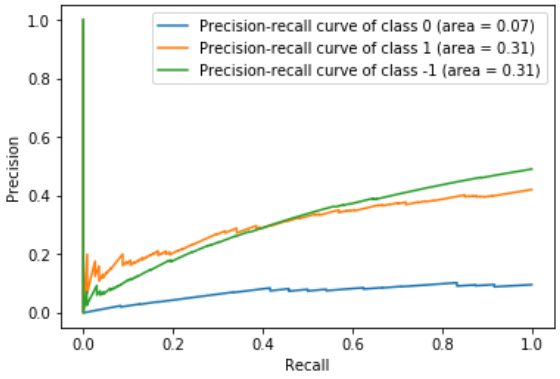

Ilustración 20: Precision Recall SVM RBF

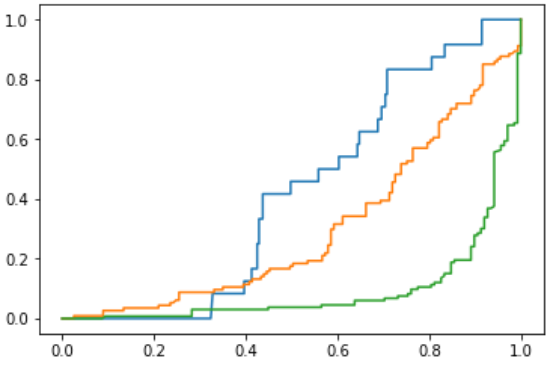

Ilustración 21: ROC SVM RBF

Como hemos podido observar los resultados obtenidos difieren bastante de los obtenidos previamente en la clasificación binaria. Y es que los datos en test no se ajustan en gran medida a los entrenados. Por esta razón, y siguiendo el consejo de nuestros profesores, decidimos hacer una clasificación binaria y considerar los resultados de entrenamiento sospechosos como si estos datos correspondieran a los de sitios web fraudulentos.