Competencia

Mejía Quintero, Camila - cmejia3@eafit.edu.co
Velasquez Gaviria, Diana Catalina - dvelasq8@eafit.edu.co
Cuscagua López, Juan Mauricio - jcuscagu@eafit.edu.co

Programa: métodos estadísticos avanzados en ciencias de los datos
Docente: Andres Ramirez Hassan

12 de octubre de 2019

Resumen

Aquí resumen del trabajo

1. Análisis Descriptivo

1.1. Variables respuesta

Para llevar a cabo el análisis sobre las bases objeto de estudio se realizó primero que todo un entendimiento general de los datos. Para esto se realizó una visualización inicial y un resumen general de las variables de la base.

Se tienen un total de 150 registros y 32 variables. En la base de datos binaria, la variable de interés (variable respuesta) es "yL" la cual toma valores de 0 y 1 y está conformada así:

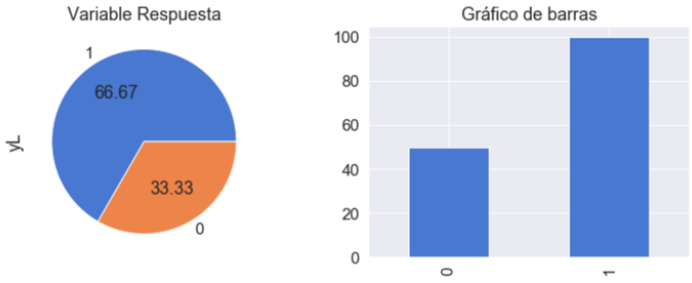


Figura 1: y binary.

Como se puede ver en la figura 1, El 33.33 % de los datos pertenecen al nivel 0, lo cual corresponde a 50 observaciones y el 66.67 % permanecen al nivel 1 y equivalente a 100 observaciones.

Para la base de datos continua la variable respuesta presenta el comportamiento observado en la figura 2

Figura 2: y continuos.

En este caso se observa que la variable respuesta presenta valores entre -6 y 6, presentando su mayor concentración entre 0 y 2 y con una distribución muy simétrica de los datos.

Y por último para la base de datos con respuesta de conteo, se presenta el comportamiento mostrado en la figura 3

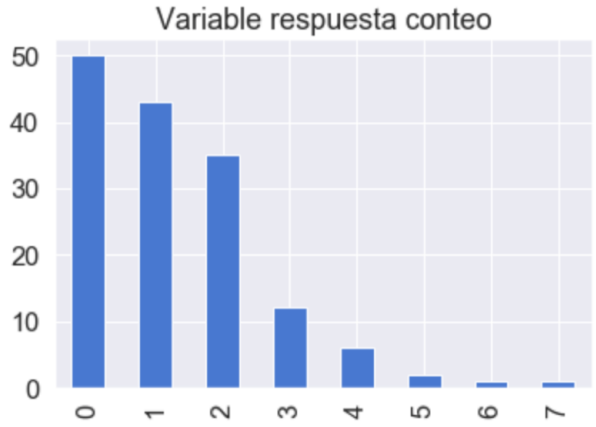


Figura 3: y continuos.

En este caso la variable respuesta toma valores de 0 a 7, siendo el 0, el 1 y el 2 los que mayor frecuencia presentan.

1.2. Variables independientes

Luego de entender el comportamiento de la variable respuesta para cada una de las bases de datos, se realizó un análisis de las variables independientes. Lo primero que se observó es que en las tres bases el valor de las x es el mismo.

En la figura 4 se muestra la distribución de cada una de las variables regresoras.

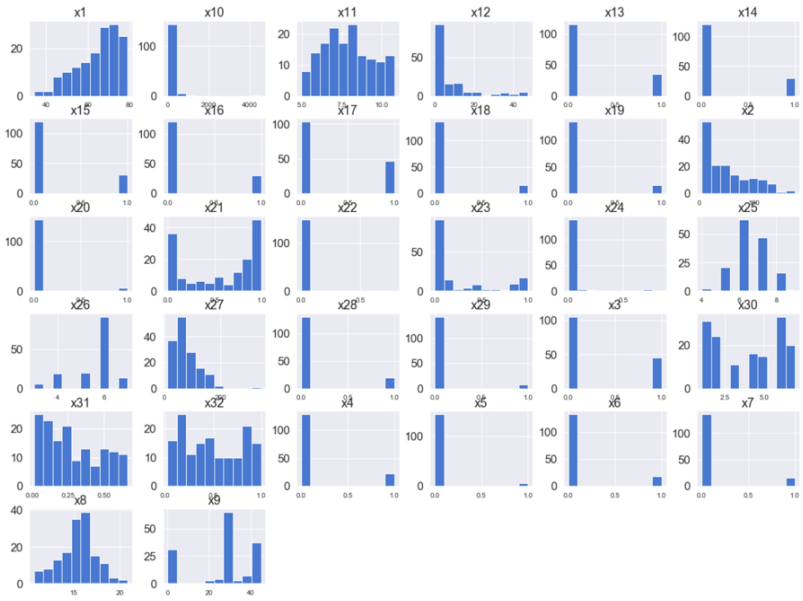


Figura 4: Histogramas.

Puede observarse que se tienen varias variables dicótomas, y otras variables como x7, x9, x26, x30 con muy pocos valores, pareciendo variables discretas.

A continuación se realizó un análisis de correlación de las variables continuas para detectar posibles indicios de multicolinealidad y se encontraron correlaciones muy fuertes tanto positivas como negativas entre algunas variables, por ejemplo x1 y x2 presentan alta asociación lineal negativa, y x31 y x32 fuerte relación positiva.

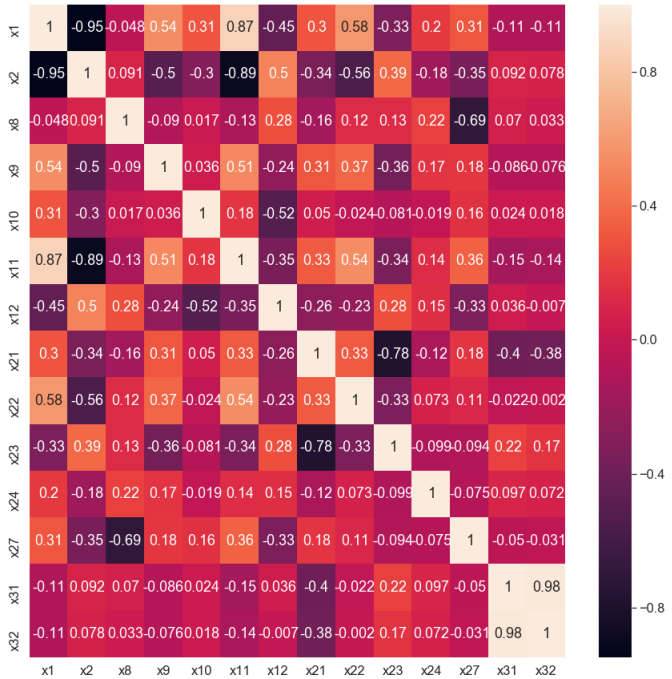


Figura 5: Matriz de Correlación.

Y finalmente se calcularon el determinante = 1.158e-13 y el número de condición = 155663226.943 de la matriz de covarianzas.

Con el uso de otros gráficos como el scatter plot pudo observarse también otro tipo de relación entre las variables, y finalmente lo que nos corrobora todo lo visto anteriormente es el determinante de la matriz de covarianza que esta muy cercano a cero indicando que se tienen problemas de multicolinealidad.

2. Problema Binario

2.1. Selección de variables

Para realizar el proceso de selección de variables se utilizaron diversas técnicas, tales como: filter methods (varianza constante, cuasi-constante, características duplicadas, correlación, mutual information, fisher score, univariate AUC score), wrapper methods (step fordward, step backward, subset) y embedding methods (lasso regularisation, trees, random forest importance, gradient. Boosted machines, se usó una técnica de bayes (spike and slab) y algoritmos evolutivos, específicamente una técnica de differential evolution con un naive bayes como función de optimización).

Con los primeros métodos se encontró que no se contaba con características duplicadas, ni con variables con varianza constante o casi contante, por lo tanto con estos métodos básicos no se eliminó ninguna característica. Adicional a todos estos métodos, se utilizó un criterio propio con base en lo observado en el análisis descriptivo para la preselección de algunas características.

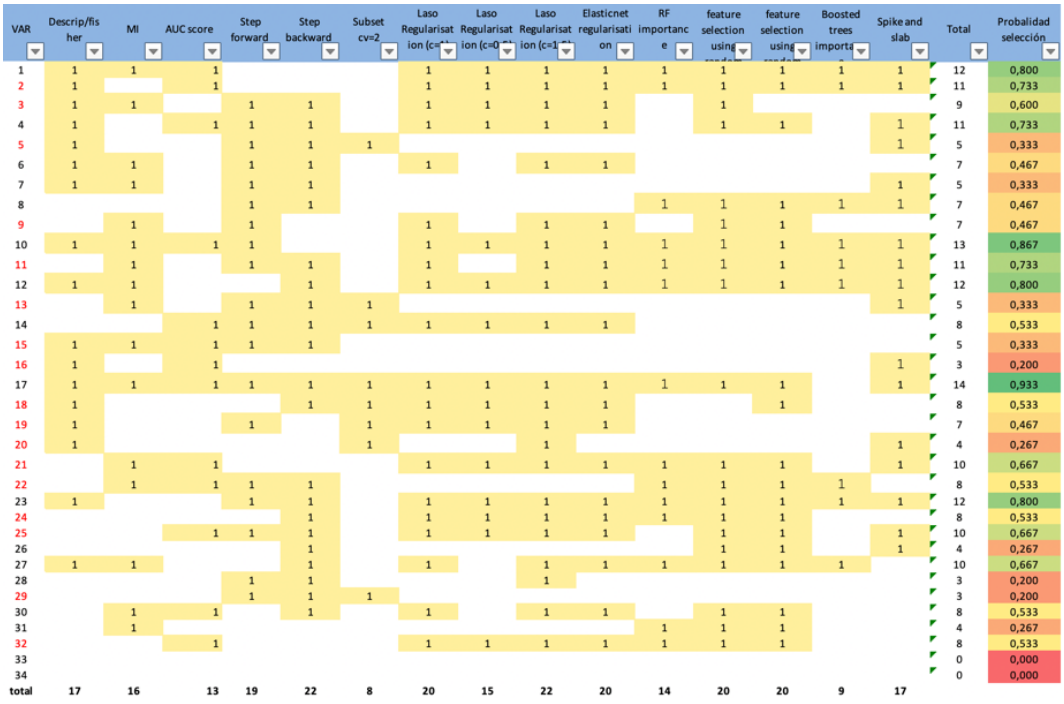
Al utilizar los wrapper methods, se realizó validación cruzada con kfold = 5 para tomar las variables donde se maximizaba el resultado del AUC en el dataset de validación. De la misma manera se realizaron varias particiones aleatorias de train y test para hacer más robusto el proceso de selección teniendo en cuenta la variabilidad presentada en los resultados y el pequeño tamaño de la muestra

Todo el proceso de selección de variables se corrió sobre el dataset de entrenamiento con el fin de evitar el sobreajuste. Estos fueron los métodos utilizados:

- Descriptivo/fisher para variables dicótomas
- Mutual Information
- AUC univariate score
- Step forward: Se iteró para diferente número de características y se tomó el de mayor AUC. Además se realizó validación cruzada con k=5
- Step backward: Se iteró para diferente número de características y se tomó el de mayor AUC. Además se realizó validación cruzada con k=5
- Subset cv=2: se realizó para un total de hasta 12 combinaciones de características con validación cruzada con k = 2. (Esto por el costo computacional que presenta).
- Laso Regularisation (c=1): se aplicó la regularización lasso con varios criterios de regularización. (c = 0.5, c = 1 y c=1.5) mientras más alto el valor del hiperparámetro de regularización, menos estricta es la penalidad. Se tomaron las características de los tres modelos.
- Elasticnet regularisation
- Random Forest importance
- Recursive feature selection using random forests importance RFE: se remueve una característica en cada iteración. La menos importante.

- Recursive feature selection using random forests importance RFECV: Es la misma técnica anterior pero con validación cruzada. Se tomó k=5.
- Gradient Boosted trees importance
- Spike and slab

Para los métodos donde se variaron los hiperparámetros de regularización se seleccionaron la cantidad de variables donde se lograba el mayor resultado de AUC. Finalmente luego de aplicar todos los métodos con sus múltiples corridas se creó una tabla con el listado de variables de la base original y la cantidad de métodos usados de la siguiente manera:



VAR	Descrip/fisher	MI	AUC score	Step forward	Step backward	Subset cv=2	Laso Regularisation (c=	Laso Regularisation (c=0	Laso Regularisation (c=1	Elasticnet regularisation	RF importance	feature selection using	feature selection using	Boosted trees importa	Spike and slab	Total	Probalidad selección
1	1	1	1				1	1	1	1	1	1	1	1	1	12	0,800
2	1		1				1	1	1	1	1	1	1	1	1	11	0,733
3	1	1		1	1		1	1	1	1		1				9	0,600
4	1		1	1	1		1	1	1	1		1	1		1	11	0,733
5	1			1	1	1									1	5	0,333
6	1	1		1	1		1		1	1						7	0,467
7	1	1		1	1										1	5	0,333
8				1	1						1	1	1	1	1	7	0,467
9		1		1			1		1	1		1	1			7	0,467
10	1	1	1	1			1	1	1	1	1	1	1	1	1	13	0,867
11		1		1	1		1		1	1	1	1	1	1	1	11	0,733
12	1	1			1		1	1	1	1	1	1	1	1	1	12	0,800
13		1		1	1	1									1	5	0,333
14			1	1	1	1	1	1	1	1						8	0,533
15	1	1	1	1	1											5	0,333
16	1		1												1	3	0,200
17	1	1	1	1	1	1	1	1	1	1	1	1	1		1	14	0,933
18	1				1	1	1	1	1	1			1			8	0,533
19	1			1		1	1	1	1	1						7	0,467
20	1					1			1						1	4	0,267
21		1	1				1	1	1	1	1	1	1		1	10	0,667
22		1	1	1	1						1	1	1	1		8	0,533
23	1			1	1		1	1	1	1	1	1	1	1	1	12	0,800
24					1		1	1	1	1	1	1	1			8	0,533
25			1	1	1		1	1	1	1		1	1		1	10	0,667
26					1							1	1		1	4	0,267
27	1	1			1		1		1	1	1	1	1	1		10	0,667
28				1	1				1							3	0,200
29				1	1	1										3	0,200
30		1	1		1		1		1	1		1	1			8	0,533
31		1									1	1	1			4	0,267
32			1				1	1	1	1	1	1	1			8	0,533
33																0	0,000
34																0	0,000
total	17	16	13	19	22	8	20	15	22	20	14	20	20	9	17		

Figura 6: Matriz.

Cada columna representa un método de selección. La primera columna contiene todas las variables y se asigna un valor de 1 en cada método si la variable fue seleccionada por el mismo. Al final se crearon dos medidas, una representa el número total de veces que la variable fue seleccionada por los diferentes métodos y la última columna representa la probabilidad de selección de cada variable medida como la razón entre el número de veces que la variable fue seleccionada y el número total de métodos usados.

Con base en la tabla obtenida se utilizaron varios criterios para la selección de características, el primero fue tomar las variables que tuvieron una probabilidad de selección mayor a 0.50, el segundo fue tomar las variables con probabilidad de selección mayor a 0.55, 0.6, luego mayor a 0.65, a 0.70 y a 0.8 y este fue el resultado de la selección:

- **mayor al 55 % (18 variables)**: $x1$, $x2$, $x3$, $x4$, $x10$, $x11$, $x12$, $x14$, $x17$, $x18$, $x21$, $x22$, $x23$, $x24$, $x25$, $x27$, $x30$, $x32$.
- **mayor al 60 % (12 variables)**: $x1$, $x2$, $x3$, $x4$, $x10$, $x11$, $x12$, $x17$, $x21$, $x23$, $x25$, $x27$.
- **mayor al 65 % (11 variables)**: $x1$, $x2$, $x4$, $x10$, $x11$, $x12$, $x17$, $x21$, $x23$, $x25$, $x27$.
- **mayor al 70 % (8 variables)**: $x1$, $x2$, $x4$, $x10$, $x11$, $x12$, $x17$, $x23$.

- **mayor al 80 % (5 variables)**: $x1$, $x10$, $x12$, $x17$, $x23$.

Con base en lo anterior se construyeron cinco subconjuntos de bases adicionales, cada una con las variables seleccionadas por cada criterio de selección.

2.2. Selección del modelo

Una vez terminados los procesos de preparación y exploración de los datos y la selección de características se realizó el proceso de modelado. Se utilizaron 7 modelos de aprendizaje sobre los 5 subset mencionados anteriormente: Regresión logística, Gradient boosting, Decission Tree, Random Forest y suport vector machines y k nearest neighbors y Gaussian Process Classifier. Para evaluar el rendimiento de los modelos se analizó el ROC AUC en el subconjunto de test y también se analizó el accuracy. Cada modelo en cada conjunto de variables seleccionadas se corrió variando los hiperparámetros y con validación cruzada con kfold = 5. De cada modelo se escogió el mejor para cada subset tanto teniendo en cuenta la métrica de accuracy como la de auc.

Este proceso se corrió 1500 veces con cada modelo y se encontró que la regresión logística presentaba los mejores valores de AUC y Accuracy y las menores desviaciones.

Adicionalmente durante el proceso de modelado se tuvieron en cuenta algunos aspectos tales como:

Al usar la regresión logística, teniendo en cuenta la diferencia presentada entre los AUC de entrenamiento y los de validación, se aplicó una regularización variando un parámetro en el código en diferentes valores y se seleccionó el criterio que maximizaba el resultado el AUC. Este varía para cada subconjunto de variables.

En las máquinas de soporte vectorial se realizó una variación a los kernel, se utilizaron los siguientes: Lineal, sigmoide, polinomial y radial. Teniendo en cuenta que es un modelo costoso computacionalmente y que los resultados con los diferentes kernel no fueron los mejores se decidió utilizar el kernel lineal que arrojó los mejores resultados.

2.3. Resultados

Luego de tener el mejor modelo (regresión logística) y teniendo en cuenta la sensibilidad presentada al partir la muestra para el entrenamiento y la validación, con el fin de hacer el proceso de selección más robusto, se realizaron 1500 particiones de la base y se corrió el modelo seleccionado sobre los 5 conjuntos de variables, con el fin de identificar cual presentaba el mejor desempeño. El criterio utilizado para la selección del mejor conjunto, fue aquel donde se presentaba una mediana de AUC y Accuracy mayor y un mayor valor de la media de estas medidas restando su desviación estándar sobre las 1500 corridas. En este caso el conjunto seleccionado fue el de aquellas variables que tenían probabilidad de selección mayor o igual a 0.8 (x1,x10,x12,x17,x23). Como se observa el la figura 7 y 8.

	Modelo	Median: auc	Median: Accuray
0	[x1, x2, x4, x10, x11, x12, x17, x21, x23, x25...	0.659091	0.733333
0	[x1, x2, x3, x4, x10, x11, x12, x14, x17, x18,...	0.674603	0.733333
0	[x1, x2, x4, x10, x11, x12, x17, x23]	0.674603	0.733333
0	[x1, x10, x12, x17, x23]	0.674603	0.733333
0	[x1, x2, x3, x4, x10, x11, x12, x17, x21, x23,...	0.682540	0.733333

Figura 7: Resultados del problema binario

AUC	mean	std	dif
x1,x2,x4,x10,x11,x12,x17,x21,x23,x25,x27	0,65920	0,07608	0,58312
x1,x10,x12,x17,x23	0,66955	0,07694	0,59262
x1,x2,x3,x4,x10,x11,x12,x17,x21,x23,x25,x27	0,68403	0,08045	0,60358
x1,x2,x4,x10,x11,x12,x17,x23	0,67381	0,08257	0,59123
x1,x2,x3,x4,x10,x11,x12,x14,x17,x18,x21,x22,x23,x24,x25,x27,x30,x32	0,67215	0,08382	0,58834

Accuracy	mean	std	dif
x1,x10,x12,x17,x23	0,73596	0,06988	0,66608
x1,x2,x4,x10,x11,x12,x17,x21,x23,x25,x2	0,72431	0,07092	0,65339
x1,x2,x3,x4,x10,x11,x12,x17,x21,x23,x25,x27	0,74650	0,07335	0,67315
x1,x2,x4,x10,x11,x12,x17,x23	0,73718	0,07365	0,66353
x1,x2,x3,x4,x10,x11,x12,x14,x17,x18,x21,x22,x23,x24,x25,x27,x30,x32	0,71572	0,07607	0,63965

Figura 8: Relación entre la media y la varianza

3. Problema Continuo

3.1. Selección de variables

La selección de variables para este caso fue basado en cross-validation utilizando especialmente tres métodos, Recursive Feature Elimination (RFE), Lasso y Bayesian Model Averaging (BMA).

- BMA: Fue el primer método que se corrió, con priors sin información alguna, por lo que era una distribución uniforme para todas y esto indico un más o menos el número máximo de variables que pudiesen estar incluidas en el proceso generador de los datos. También se escogieron las variables con mayor probabilidad como un conjunto candidato al proceso generador de datos.
- RFE: tiene tres parámetros el primero es con que estimador desea hacer el ranking las variables, el segundo es cuantas variables seleccionar y el último cuantas características elimina en cada iteración. Tomando la cota superior, donde toda variable que en la salida del BMA fuese distinto de cero, sería contada para el parámetro máximo de este método, por lo cual se decidió correr este método escogiendo desde una variable hasta 15, con el estimador de la regresión lineal como método de ranqueo y siempre eliminando de a una variable, cada cantidad máxima de variables se corrió 100 veces, cada vez partiendo nuevamente el conjunto de entrenamiento y test, buscando una salida robusta. Este procedimiento genero unos conjuntos de variables candidatos.

- Lasso: Con una idea similar al anterior, se generó un arreglo que empezaba en 0.1 e iba hasta 1, para el parámetro alpha de esta regresión, y con cada valor de alpha se hicieron 100 corridas. Este procedimiento generó también unos conjuntos de variables candidatas.

Terminado este procedimiento se generó una matriz de 32x1284, 32 variables y 1285 conjuntos de variables únicos, con la salida de los tres métodos y se generó una prior distinta para cada variable que correspondía con la proporción de cuántas veces en los 1284 conjuntos había sido seleccionado cada variable y se corrió por última vez el BMA y se agregó el resultado de esa corrida. Finalmente, se corren todos los posibles conjuntos 1000 veces, cada vez particionando nuevamente el set de datos, y se sacan los indicadores promedio de todas las corridas, y con eso se hace una evaluación final.

3.2. Selección del modelo

Luego de analizar, los resultados obtenidos en el procedimiento anterior se seleccionan los mejores 13 conjuntos de variables y para estos se corre un grid search utilizando tres modelos: lineal, ridge y maquinas de soporte vectorial, y cada conjunto de variables se corrió 500 veces y de cada partición se guarda el mejor modelo junto el parámetro que lo obtuvo y así se obtiene la siguiente gráfica:

3.3. Resultados

Variables	Modelo_name	Parametro	Accuracy_mean	Accuracy_std	Accuracy_median	Count	MSE_mean	MSE_std	MSE_median
x12,x23,x24,x25,x32	svm	linear	0.776451	0.071557	0.800000	201	3.889560	0.837374	3.879599
x1,x23,x25,x32	svm	rbf	0.781931	0.058864	0.800000	107	3.930250	0.971606	3.866396
x6,x7,x12,x23,x24,x25,x32	svm	rbf	0.775000	0.065355	0.766667	216	3.908978	1.001383	3.878556
x23,x25,x31	svm	linear	0.763527	0.068926	0.766667	138	3.814295	0.966117	3.644562
x23,x25,x31	svm	rbf	0.766429	0.071508	0.766667	280	3.839625	0.938708	3.842631
x12,x23,x24,x25,x32	ridge	1.0	0.775661	0.066544	0.766667	189	3.943660	0.902762	3.823967
x1,x13,x23,x25,x31,x32	svm	rbf	0.766227	0.062859	0.766667	303	3.875697	0.827132	3.890582
x13,x23,x20,x25,x31,x32	svm	rbf	0.758248	0.065087	0.766667	293	3.836747	0.922052	3.906913
x1,x23,x25,x31,x32	svm	rbf	0.767888	0.067629	0.766667	273	4.002617	0.974876	4.026265
x6,x13,x20,x23,x25,x27,x31	svm	rbf	0.754581	0.074865	0.766667	171	3.845863	0.917604	3.741595

Figura 9: Resultados del problema continuo.

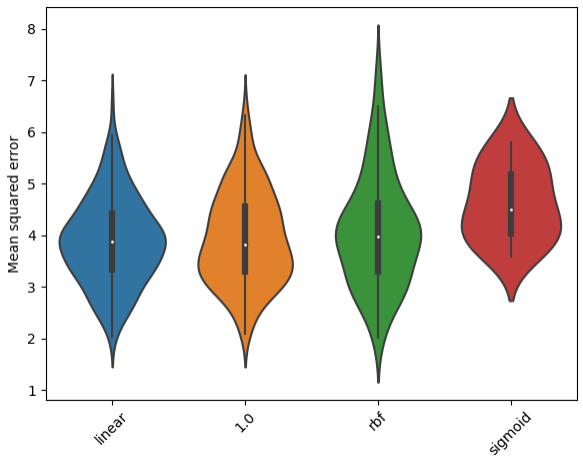


Figura 10: Distribución del Mean Square Error: x12,x23,x24,x25,x32

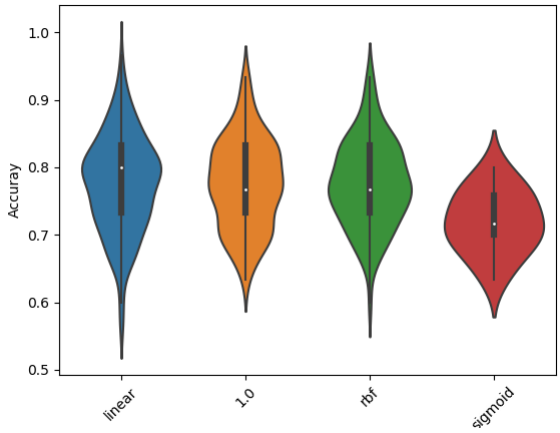


Figura 11: Distribución del Accuracy: x12,x23,x24,x25,x32

De la Figura 9 se calcularon dos columnas de ranking una por el MSE donde el más pequeño toma el número uno y lo mismo se hizo para el Accuracy pero de forma inversa, y luego la suma de los dos ranking nos da una señal de un conjunto que tiene buenos resultados en ambos scores, y también se validó que la varianza no fuera muy grande, que da una señal de robustez.

Por eso la decisión final de seleccionar las variables x12,x23,x24,x25 y x32 y el modelo es una máquina de soporte vectorial con un kernel lineal.

4. Problema Conteo

4.1. Selección de variables

Para el proceso de selección de variables en el problema de conteo se consideraron los métodos para selección de variables mencionados en las secciones anteriores. En todos los casos se observó que los conjuntos candidatos para usarse como regresores no generaban los resultados deseados respecto a la capacidad de predicción general y específica de para una regresión Poisson. Se procedió entonces explorar la relevancia de los regresores a partir de la evaluación de 8 algoritmos: Poisson Regression, Multilayer Perceptron Regressor, Multilayer Perceptron Classifier, Support Vector Machine Classifier, Support Vector Machine Regressor, Random Forest Regressor, Multiclass Logistic Regression y XGBoost. De aquí se observó que este último presentaba mejores resultados que los demás a pesar de no ser los deseados en la capacidad de predicción. De esta manera, se escogió el XGBoost para explorar una metodología heurística de búsqueda aleatoria.

El enfoque es bastante simple, se generaron 2.000.000 de combinaciones diferentes para la selección de variables. Estas combinaciones variaban en términos de cuántos regresores tomar y cuáles eran seleccionados. Así, para cada conjunto de variables se entrenó un modelo XGBoost (del que se darán detalles en la siguiente sección) de tal forma que al evaluar su capacidad de predicción, se encontrara aquella combinación de la muestra aleatoria que obtiene el mínimo error cuadrático medio (en adelante MSE) entre la predicción y el valor real de la variable respuesta. Dicha predicción es discretizada a partir del redondeo usual para variables continuas, tomando el entero más cercano al número real.

En este orden de ideas, se seleccionan las siguientes variables como los regresores para el pronóstico de la variable respuesta: $x4$, $x13$, $x25$, $x24$, $x27$, $x7$, $x14$, $x3$, $x2$, $x8$, $x22$.

4.2. Selección del modelo

Una vez seleccionadas las variables a implementar se hicieron pruebas con los modelos anteriormente mencionados. De igual manera que en los problemas anteriores, se prueban los Los resultados de la capacidad de predicción general y específica para cada algoritmo utilizando una parte de los datos que se excluyó para el proceso de selección de variables. Los resultados de los diferentes modelos y sus respectivas capacidades de predicción se muestran en la tabla 1. Con base en los resultados obtenidos por los modelos, se seleccionó el modelo de ensamblaje XGBoost como el adecuado para atacar este problema.

De manera general, el XGBoost es un algoritmo de ensamblado basado en árboles de decisión que implementa un framework de gradient boosting.

4.3. Resultados

	MSE	**CPE**[1]
XGBoost	1.0327	0.60
Multiclass Logistic Regression	1.4605	0.63
Random Forest Regressor	1.3416	0.56
SVM Regressor	1.3784	0.63
SVM Classifier	1.4719	0.63
Multilayer Perceptron Classifier	1.4944	0.60
Multilayer Perceptron Regressor	1.3038	0.63
Poisson Regression	1.2780	0.60

Cuadro 1: Resultados del problema de conteo.

Dentro de los resultados obtenidos, aquel que guarda la mejor relación entre la capacidad predictiva general (MSE) y la específica (CPE), como se mencionó anteriormente, fue el algoritmo XGBoost. Este resultado es interesante ya que de forma intuitiva, por las características de la variable de respuesta, se podría pensar en una regresión Poisson. Sin embargo, después de su evaluación con el conjunto de testeo, este no logra superar el desempeño del XGBoost en su capacidad de predicción general a pesar de lograr la misma capacidad específica.