

FACULTAD DE ESTUDIOS ESTADÍSTICOS

MÁSTER EN MINERÍA DE DATOS E INTELIGENCIA DE NEGOCIOS

Curso 2022/2023

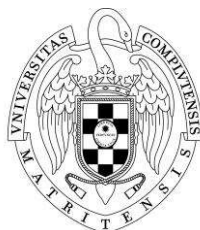
Trabajo de Fin de Máster

TÍTULO: Predicción del voto electoral en España mediante encuestadoras y variables fundamentales.

Alumno: Enric Palau Payeras.

Tutor: Javier Álvarez Liébana

Febrero de 2023



UNIVERSIDAD COMPLUTENSE
MADRID

Resumen

En tiempos de crisis como la COVID-19 o la anterior crisis financiera, se genera mucha expectativa y contemplación sobre el clima político y de gobierno. Especialmente, cuando se acercan las convocatorias de elecciones. Los medios de comunicación se hacen eco de los resultados de las casas encuestadoras y otros medios, para contestar la demanda generada por una sociedad adversa a la incertidumbre y descontenta con el contexto vivido.

El problema se atenúa cuando el resultado de las elecciones es muy diferente al esperado. Los medios que informan sobre las carreras electorales mediante datos son criticados. Olvidamos que no se trata de un sesgo malversado, ni de una negligencia por parte de la encuestadora. Realmente, estos sesgos son mayoritariamente sistemáticos a los que se les añade el factor de la incertidumbre.

En este estudio, pretendemos analizar el uso de variables tanto de contexto como de la información generada por las encuestadoras para comprender el origen del error en la predicción electoral y optimizar los resultados de estas predicciones generadas por los medios. Para ello, pasaremos por una fase de procesamiento y recopilación de encuestas y variables de contexto para finalmente llevar a cabo la predicción y los promedios.

Palabras Clave: Política, encuestadoras, sesgo, elecciones, porcentaje de voto

Abstract

In times of crisis such as COVID-19 or the previous financial crisis, expectations and contemplations are centered on political climate and government. Especially, when primaries get closer. The media takes advantage of the results generated by polling houses and other media to satisfy the demand generated by a society adverse to uncertainty and discontent with the lived context.

The problem is mitigated when the outcome of the elections is very different from what was expected. Media that report on electoral races through data are criticized. We forget that this is not a misappropriated bias, nor a negligence by the pollster. These biases are mostly systematic to which the uncertainty factor is added.

In this study, we intend to explore the use of both context variables and the information generated by the pollsters, to understand the origin of the error in the electoral prediction and optimize the results of these predictions generated by the media. To do this, we will go through a phase of processing and collecting surveys and context variables to finally go through the prediction and averages phase.

Keywords: Politics, pollsters, bias, elections, percentage of vote

Índice de contenido

Resumen.....	1
Abstract.....	1
Índice de contenido	2
Índice de tablas	5
1. Introducción.....	5
1.1. Estado del arte del estudio.....	6
1.2. Justificación del trabajo.....	7
2. Objetivos	7
2.1. Variable objetivo	8
3. Metodología empleada y estructura del estudio	8
4. Construcción de la base de datos	12
4.1. Datos de Wikipedia (encuestas)	12
4.1.1. Extracción de los datos de Wikipedia.....	13
4.1.2. Transformación y preprocesamiento de los datos de Wikipedia.....	14
4.2. Datos del Ministerio del Interior (censo y electorado).....	15
4.2.3. Extracción de los datos fundamentales	19
5. Promedio de Encuestas.....	21
5.1. Recogida de encuestas	22
5.2. Incorporar la incertidumbre	23
5.3. Ajustar el House Effect	23
6. Aplicación de la metodología SEMMA	26
6.1. Sample.....	26
6.2. Explore.....	26
6.2.1. Análisis Gráfico (visualización de datos).....	26
6.2.2. Análisis Gráfico mediante modelos de árbol.....	28
6.3. Modify (depuración y modificación de las variables)	30
6.4. Model	32
6.4.1. Criterio de bondad de ajuste elegido	33
6.4.2. Árboles de decisión	34
6.4.3. Bagging de Árboles y Random Forest	39
6.4.4. Gradient Boosting	44
6.4.5. Redes Neuronales	49
6.4.6. Máquinas de vector soporte (SVM)	61
6.4.6.1. SVM con kernel lineal	62
6.4.6.2. SVM con kernel polinomial	63
7. Evaluación comparativa entre modelos, casas, encuestas y promedios	67
8. Conclusiones	70
Bibliografía	71
Anexos.....	74

Índice de figuras

Figura 1: Estructura del estudio.....	9
Figura 2: Esquema proceso ETL	10
Figura 3: Promedio de encuestas	11
Figura 4: Esquema SEMMA.....	11
Figura 5: Datos de encuestadoras en la fuente de origen	13
Figura 6: "wiki_info" tabla para automatizar la extracción de datos.....	14
Figura 7: "national_surveys_with_n" histórico de encuestas no pivotado.....	15
Figura 8: "national_surveys_longer" histórico de encuestas pivotado.....	15
Figura 9: Resultado electoral según Wikipedia	16
Figura 10: "datos_reales", histórico de estadísticas electorales por carrera y partido .	18
Figura 11: Estimaciones de voto y factores de sesgo (2ª carrera del 2019, PSOE)	22
Figura 12: Días para las elecciones y rangos de evaluación	23
Figura 13: wing_effect_e de las encuestadoras (Carrera del 2019-11-10)	25
Figura 14: Ejemplo del efecto de promedios sobre las estimaciones (2019-11-10)	25
Figura 15: Errores, estimaciones de voto y voto real (distribución en el histórico)	27
Figura 16: Voto Real por carrera y partido (evolución sobre el histórico)	27
Figura 17: Ejemplo de árbol de decisión.	29
Figura 18: Filtro de variables, ejemplo árbol de decisión	30
Figura 19: Estudio comparativo entre métodos de imputación	31
Figura 20: Ejemplo de Early Stopping (arbol_1), validación cruzada simple (4 grupos) 35	
Figura 21: Validación cruzada repetida (4 grupos y 10 iteraciones) en árboles	36
Figura 22: Errores estimados en test (arbol_3).....	37
Figura 23: Predicción en test del % de voto por partido (arbol_3)	37
Figura 24: Estimaciones para las elecciones de 2023 (arbol_3).....	38
Figura 25: Ejemplo de OOB (bag_arbol_1), validación cruzada simple (4 grupos)	40
Figura 26: Validación cruzada repetida (4 grupos y 10 iteraciones) en bagging.....	41
Figura 27: Errores estimados en test (rf_bag_arbol_3).	42
Figura 28: Predicción en test del % de voto por partido (rf_bag_arbol_3).....	43
Figura 29: Estimaciones para las elecciones de 2023 (rf_bag_arbol_3)	43
Figura 30: Estudio de parámetros para GBM, validación cruzada simple (4 grupos). ...	45
Figura 31: Validación cruzada repetida (4 grupos y 10 iteraciones) en boosting.	47
Figura 32: Errores estimados en test (gbm_1).	47
Figura 33: Predicción en test del % de voto por partido (gbm_1)	48
Figura 34: Estimaciones para las elecciones de 2023 (gbm_1)	49
Figura 35: Esquema básico del modelo de red (concepto)	49
Figura 36: Criterios de parada (step_forward) con SAS	52
Figura 37: Criterios de parada (step_backward) con SAS	53
Figura 38: Criterios de parada (step_wise) con SAS.....	54
Figura 39: Validación cruzada repetida (4 grupos y 10 iteraciones) en regresión lineal	56
Figura 40: Estudio de parámetros para redes, validación cruzada simple (4 grupos) ...	57
Figura 41: Estudio de parámetros para redes, validación cruzada simple (4 grupos). ..	58
Figura 42: Validación cruzada repetida (4 grupos y 10 iteraciones) en redes.	59
Figura 43: Errores estimados en test (red_BIC_4).....	59
Figura 44: Predicción en test del % de voto por partido (red_BIC_4).....	60
Figura 45: Estimaciones para las elecciones de 2023 (red_BIC_4)	61

Figura 46: Parámetro C para SVM lineal, validación cruzada simple (4 grupos).....	62
Figura 47: Parámetros para SVM (kernel ²2), validación cruzada simple (4 grupos).....	63
Figura 48: Parámetros para SVM (kernel ²3), validación cruzada simple (4 grupos).....	64
Figura 49: Validación cruzada repetida (4 grupos y 10 iteraciones) en SVM.....	65
Figura 50: Errores estimados en test (SVM_pol_3).....	66
Figura 51: Predicción en test del % de voto por partido (SVM_pol_3)	66
Figura 52: Estimaciones para las elecciones de 2023 (SVM_pol_3).....	67
Figura 53: Ranking de encuestadoras de TheElectoralReport	68
Figura 54: Casas, encuestas, promedios y modelos; Evaluación en test	68

Índice de tablas

Tabla 1: Descripción general de la base de datos por bloques.	12
Tabla 2: Variables creadas como promedios.....	24
Tabla 3: Resumen de BBDD para metodología SEMMA.....	26
Tabla 4 : Modelos de árbol, parámetros y resultados en validación cruzada repetida .	35
Tabla 5: Modelos de árbol y resultados en test	36
Tabla 6: Bagging, parámetros y resultados en validación cruzada repetida	41
Tabla 7: GBM, parámetros y resultados en validación cruzada repetida.	46
Tabla 8: Selección de variables por bloques.	55
Tabla 9: Exploración de parámetros (redes neuronales)	56
Tabla 10: Redes, parámetros y resultados en validación cruzada repetida	58
Tabla 11: SVM, parámetros y resultados en validación cruzada repetida	64

1. Introducción

El clima político español, lleva experimentando grandes cambios sistemáticos desde la crisis financiera del 2008 (Guasch, 2019), pasando de un sistema prácticamente bipartidista a un sistema donde la estrategia de pactos entre partidos es clave. Pasar dos veces por urnas en el 2019, es un claro ejemplo de las disrupciones ocasionadas en el sistema político. El contexto generado por el COVID-19, ha hecho aún más inestable el equilibrio de las fuerzas de gobierno, y por ello cada vez hay más demanda e interés en conocer el escenario político actual y futuro.

Para contestar esa demanda de la sociedad, los medios de comunicación e incluso los mismos partidos políticos, utilizan las encuestas electorales. Hoy en día, contamos con muchas casas demoscópicas, periódicos y organismos tanto privados como públicos, encargados de realizar encuestas electorales con el fin de determinar la intención de voto del electorado en las próximas elecciones. Un ejemplo muy seguido en España es el Centro de Investigaciones Sociológicas (CIS). Otros, como TheElectoralReport se dedican a hacer promedios y rankings sobre la calidad de las encuestadoras. Y al final de esta “food chain feeding demand”, tenemos a aquellos que utilizan las encuestadoras y los promedios para hacer predicciones del futuro electoral e informar mediante datos a la sociedad. Uno de los mayores referentes es EL PAÍS y su periodista de datos Kiko Llaneras: “We have published eight models to predict elections in Spain, Mexico, Colombia, Netherlands, France and UK [...] I believe that no one has built so many in the last two years” (Llaneras, 2018).

Cuando el resultado esperado en las elecciones es muy divergente del resultado real, se critica la metodología utilizada por los agentes de este ecosistema mediático. Nosotros nos centraremos en este último aspecto. Se usarán técnicas de aprendizaje automático (Machine Learning) para predecir el error de estos medios y a su misma vez, optimizar las predicciones en el porcentaje de voto de los principales partidos en cada carrera electoral. Para la implementación de los modelos y el tratamiento de los datos usaremos software R y código SQL. Alternativamente, tomaremos software SAS para la selección de variables y parte de la exploración de los datos. El código completo aparecerá en un repertorio de GitHub (link en los anexos finales, punto A).

1.1. Estado del arte del estudio

La ciencia política como disciplina universitaria es aún más joven que las técnicas de Machine Learning. Ambos campos son muy novedosos y tienen un gran potencial de desarrollo. Aún así, pocos son los autores que pueden presumir de haber tratado la predicción de resultados electorales por más de 8 años. El academicismo de este campo de estudio es escaso, por lo que la mayoría de nuestras fuentes y bases de trabajo serán artículos periodísticos como EL PAÍS, TheElectoralReport o FiveThirtyEight.

En este contexto, cabe destacar que en países como Estados Unidos, la competitividad entre medios ha llevado a una gran mejora de los modelos utilizados y la extracción de datos necesarios. A pesar de ello, la divulgación de los conocimientos en este nicho y la transparencia de los procesos seguidos sigue siendo escasa.

A pesar de la carencia de trabajos en esta materia realizados en España, Kiko Llaneras con sus artículos de EL PAÍS y Endika Nuñez con los artículos de TheElectoralReport van a fundamentar mayoritariamente el estudio previo a la modelización. Otros autores serán de ayuda para contrastar y comparar los métodos más usados y proponer soluciones alternativas, como será el cómputo del House Effect, tratado por autores internacionales como Anthony B. Masters en Masters (2020).

1.2. Justificación del trabajo

La inestabilidad del clima político actual, hace cada vez más difícil predecir y comprender la evolución de las carreras electorales, mientras la demanda pública exige lo contrario: entender y hacer más transparente el sistema electoral. Por ello, es necesario incentivar la creación y estudio de modelos que puedan cubrir perspectivas más heterogéneas frente este problema.

Por otro lado, resulta casi imposible encontrar un proyecto académico sobre la predicción electoral en España que sea transparente y replicable, lo cual dificulta el estudio del problema planteado. Facilitar un proyecto fácil de seguir y reconstruir desde el proceso de extracción de datos hasta la evaluación de las predicciones, permitirá proliferar y explorar el conocimiento de este tópico nicho.

Finalmente, cabe remarcar que el escaso trabajo académico en este campo implicará adaptar muchos de los procesos llevados por grandes medios como EL PAÍS, The New York Times, FiveThirtyEight etc., a los recursos de un particular con limitaciones en los recursos computacionales, temporales y de experiencia. A pesar de ello, aproximar estos conocimientos al ámbito universitario es una gran oportunidad de ampliación de conocimiento que debe ser abordada para mejorar el sistema electoral actual.

2. Objetivos

A continuación listamos los diferentes objetivos que se van a tratar de resolver en esta investigación:

- **Automatizar un proceso de extracción, transformación y carga del histórico de datos electorales y casas encuestadoras en España.**

Al tratar problemas de Machine Learning, se requiere de bases de datos con volumetrías considerables y una de las mayores barreras es la disponibilidad de un histórico de datos. Por ello, en este proyecto recopilamos los datos necesarios desde el 1982. Las fuentes de datos son muy diversas y el dato de origen puede ser difícil de tratar, por eso otro de los retos planteados es la generación “automatizada” del histórico, dando transparencia y accesibilidad al mismo para próximos estudios.

- **Estudio comparativo de algunos de los métodos encontrados y propuesta de metodología propia y cercana al ámbito académico.**

Como ya hemos comentado, las principales metodologías en la actualidad son propuestas por grandes medios de comunicación. Acercar los métodos al ámbito académico y desarrollar una metodología propia (funcional), sería la piedra angular de este proyecto.

- **Creación e identificación de diferentes modelos y algoritmos de predicción, para determinar cuál es el más apropiado para el problema planteado.**

Al no constar de los modelos usados entre nuestros referentes en el estado del arte, vamos a tratar de comparar y optimizar todos los modelos posibles. Pasaremos de modelos más sencillos al uso de técnicas de Machine Learning para evaluar la balanza entre la calidad del modelo y su interpretabilidad.

2.1. Variable objetivo

Es fundamental aclarar que la predicción objetivo en nuestro proyecto es el error cometido por las encuestas. Sobre los datos históricos, ya disponemos del porcentaje de voto estimado y el voto real, pero en el caso de elecciones futuras del 2023, no tenemos ese porcentaje de voto real, por lo que tendremos un test en campo.

A nivel de modelos nuestra variable objetivo es el error de la encuesta, pero sumándolo al porcentaje de voto estimado, vamos a poder predecir el voto real obtenido por cada partido. Queda desarrollado el análisis completo en la sección 4.

Con esta última aclaración, queremos evidenciar que, el porcentaje de voto real obtenido por los principales partidos en cada una de las carreras electorales es también nuestra variable objetivo y deriva de predecir el error cometido por las encuestas:

$$\begin{aligned} \%Voto_{real} - \%Voto_{estimado} &= error \rightarrow \\ \%Voto_{real} &= error + \%Voto_{estimado} \end{aligned}$$

3. Metodología empleada y estructura del estudio

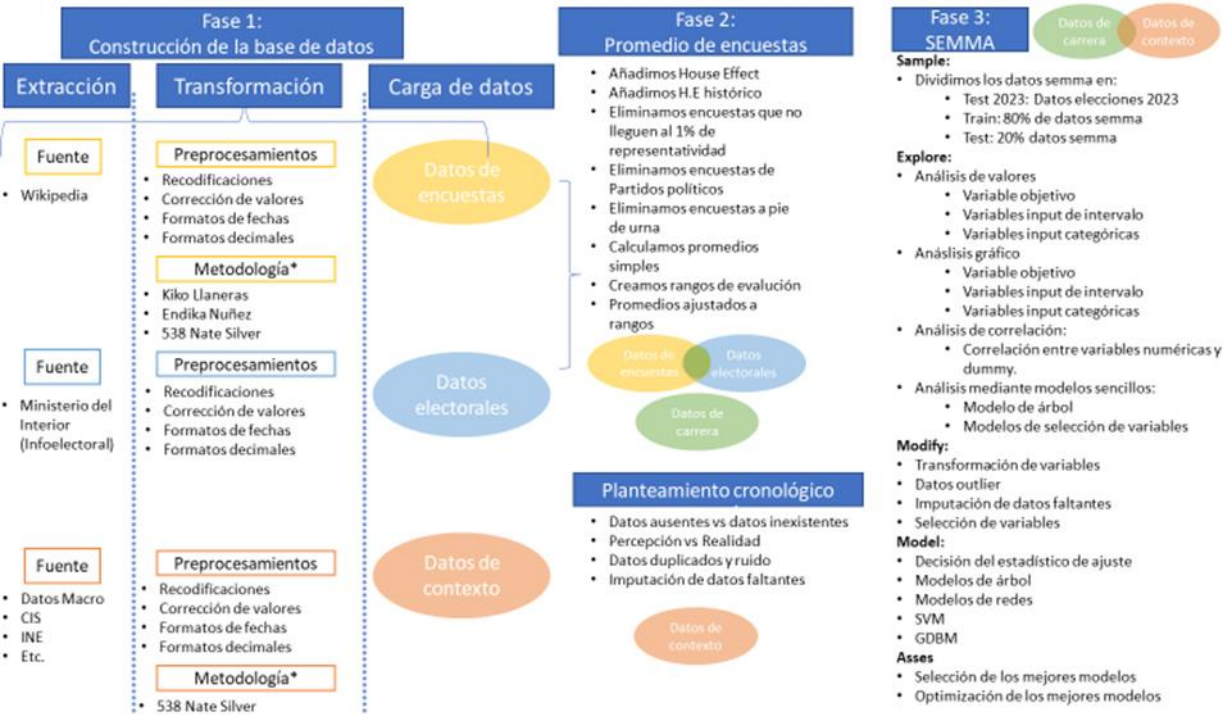
La predicción electoral puede seguir dos objetivos: predecir el Gobierno resultante o la estimación de voto. La predicción de Gobierno es la práctica más común e integra, la predicción de escaños y la formación de pactos (Ortega, s.f.). Nosotros nos centraremos en predecir el error de las encuestadoras y el porcentaje de voto que van a obtener los partidos que conforman cada carrera electoral. Por lo tanto, ya hemos definido nuestro objetivo a predecir (el error de la encuesta y el resultado electoral en porcentaje de votos) y nuestras fuentes de información o inputs: variables de contexto socio económico (fundamentales), resultados de encuestas y variables de carácter electoral. Ahora bien, cabe mencionar que no podemos predecir resultados electorales sin usar resultados de encuestas. El voto directo, puede ser explicado a través de muchos

factores tal y como resume este titular: “Claves para entender el resultado andaluz: sexo, edad, estudios y transferencias de voto” (Nuñez, 2022). Del mismo modo, se pueden hacer predicciones electorales usando sólo promedios de encuestas.

A partir de este punto, podemos plantear la siguiente cuestión: ¿por qué en este estudio nos centramos en combinar variables “fundamentales” y el promedio de encuestas? Los modelos de predicción electoral basados estrictamente en factores “fundamentales”, pueden funcionar bien sobre eventos ya vividos, pero pueden incorporar mucho ruido blanco o incluso resultados no plausibles cuando se testean en eventos futuros. Esto se debe a la gran variedad de técnicas e indicadores para medir el nivel económico a lo largo del tiempo frente al dato de la elección que incurre en una franja temporal cerrada. Tal y como ejemplifica Nate Silver en FiveThirtyEight: “One popular model based on second-quarter GDP, for example, implies that Biden is currently on track to win nearly 1,000 electoral votes — a bit of a problem since the maximum number theoretically achievable is 538.8” (Silver, 2020). Al mismo tiempo, no incorporar variables “fundamentales” puede ocasionar un incremento del sesgo y variabilidad del error al modelo, especialmente si las encuestadoras en general han fallado o no han sido correctamente seleccionadas (Silver, 2020).

Una vez tenemos claro cómo abordar el problema planteado, procedemos a ilustrar el esquema general del proyecto tal y como puede observarse a continuación:

Figura 1: Estructura del estudio



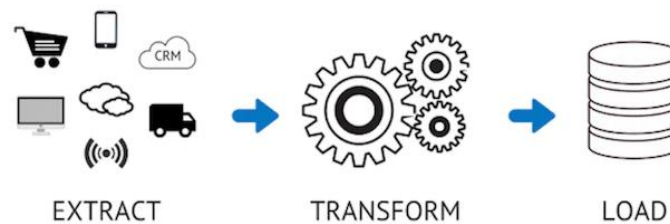
Fuente: Elaboración propia.

Tal como señalamos en la anterior imagen, las fases de desarrollo del proyecto serán: [1] Construcción de la base de datos, [2] Promedio de encuestas, [3] Planteamiento cronológico y [4] Aplicación de la metodología SEMMA.

La fase de construcción de la base de datos se basará en procesos ETL. El acrónimo inglés ETL, se refiere al proceso de extracción, transformación y carga de datos, como una secuencia básica para la integración de datos a un ecosistema de gestión de bases de datos (Appvizer, 2022). Como vemos en el anterior esquema, partiremos de extraer tres tipologías de datos distintas, cada una identificada con un color diferente: en amarillo datos de encuestas, en azul de las elecciones y en naranja datos de contexto. Las fuentes de estos datos son distintas, por lo que se procederá con tres operaciones de extracción y transformación adaptadas a cada una de ellas (segunda etapa de la Figura 2). Esta etapa preparatoria del dato es fundamental y muy costosa, ya que este es extraído en bruto de una fuente y puede no ser útil o incompatible con el software trabajado.

Una vez se ha terminado con la transformación, hemos procedido con la carga de los datos sobre nuestro entorno en el cual ya podemos trabajar (etapa 3 de la Figura 2).

Figura 2: Esquema proceso ETL

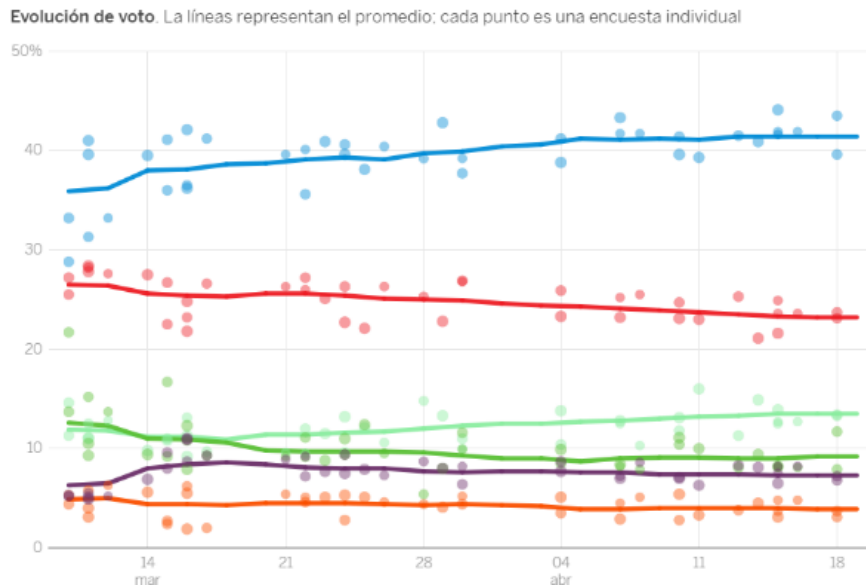


Fuente: (Appvizer, 2022).

Una vez tenemos todos los datos listos para trabajar, vamos a continuar con la segunda fase del método de trabajo aplicado en el proyecto: realización del promedio de las encuestas (ver en Figura 1 – Fase 2). En este punto procedemos a trabajar con los datos de encuestas (grupo amarillo) y datos de elecciones (grupo azul) de forma conjunta. El objetivo de esta metodología de trabajo es desarrollar un único dataset de las carreras electorales en España y las estimaciones de las encuestadoras, uniendo así el dato “real” de las elecciones con el de las estimaciones.

Por otro lado, nos inspiramos en los métodos de Endika Nuñez Larrañaga y Kiko Llaneras (entre otros) para elaborar un serie de estadísticos relacionados con las estimaciones de las encuestadoras. Según Endika, el promedio de encuestas “no es una predicción de lo que pueda ocurrir en unas eventuales elecciones, sino que ofrece la fotografía del estado de las encuestas en ese preciso momento” (Nuñez, 2022). De esta manera, siguiendo el estado del arte, definimos una serie de promedios y rangos de evaluación (días a comprender en la realización del promedio) que pasarán a ayudar a nuestras máquinas de aprendizaje a entender el comportamiento de las encuestadoras y sus estimaciones. A continuación, en la figura 3, ilustramos la idea de promediar encuestas.

Figura 3: Promedio de encuestas

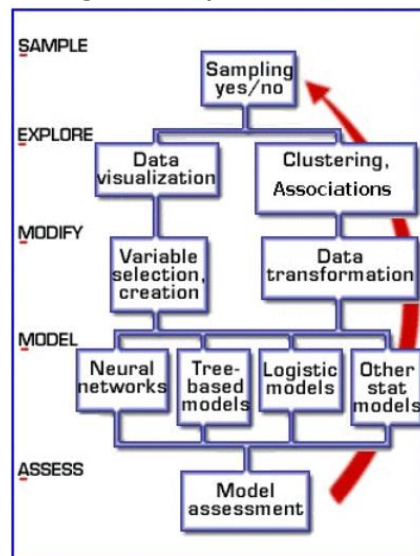


Fuente: (Llaneras, 2021).

Una vez creados los promedios, tenemos la base de datos de carrera electoral (dataset en verde) que se va a unir a nuestro conjunto de datos fundamentales (dataset en naranja), formando la base de datos que necesitamos para aplicar la metodología SEMMA, la cual constituye la fase 3 de la figura 1 en la que se estructura el estudio.

La metodología SEMMA es un metodo secuencial en el que cada uno de los pasos prepara un conjunto de datos para contruir máquinas de aprendizaje (SAS Institute Inc., 2017). Las diferentes etapas que constituyen dicha metodología son las que se presentan en la figura 4 y se detallan a continuación:

Figura 4: Esquema SEMMA



Fuente: (SAS Institute Inc., 2017).

- Sample: consiste en dividir la base de datos en muestras representativas al problema/objetivo planteado, siendo conjuntos de entrenamiento y testeo.

- Explore: se definirá la variable objetivo, se comprobará que la tipología de las variables es la correcta, se decidirá el tratamiento de datos ausentes, se realizará búsqueda de datos atípicos que puedan perjudicar el hallazgo del mejor modelo posible y se considerará la eliminación de variables.
- Model: consiste en la etapa de construcción de modelos para la predicción.
- Asses: evaluación de la calidad de los modelos y selección del mejor modelo mediante el criterio de bondad de ajuste.

Ya con una visión general del esquema del trabajo y una breve explicación de qué abarca cada una de las etapas que lo conforman, procedemos a entrar en una explicación más detallada de cada una de ellas.

4. Construcción de la base de datos

Para explicar la construcción de la base de datos (etapa 1 de la figura 1) debemos entender primero nuestro entorno de datos. En un primer momento, los datos se pueden clasificar según su sea origen: datos de encuestadoras (origen Wikipedia), datos electorales como el censo y porcentaje de voto (origen Ministerio del Interior) y datos “fundamentales” (origen fuentes diversas: p. ej. INE, el CIS, etc.). En resumen, partimos de los conjuntos de la tabla ilustrada a continuación:

Tabla 1: Descripción general de la base de datos por bloques.

Nombre de la Tabla	Fuente	Descripción
“national_surveys”	Wikipedia	Datos de encuestas y encuestadoras.
“votes_national_by_mun” “votes_national” “census_national_by_year”	Ministerio del Interior	Datos de censo y elecciones.
“BDfundamental”	CIS, INE, DatosMacro etc.	Datos de contexto socioeconómico, política, demografía etc.








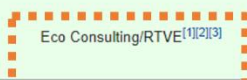
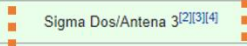
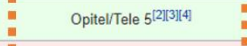
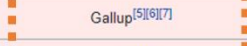

Fuente: Elaboración propia.

A continuación detallamos el proceso de extracción, transformación y carga de los datos que debe llevarse a cabo en cada uno de los módulos según sea su origen de datos.

4.1. Datos de Wikipedia (encuestas)

Los datos de encuestadoras proceden de Wikipedia y se encuentran tal y como se visualiza en la siguiente figura:

Figura 5: Datos de encuestadoras en la fuente de origen

Polling firm/Commissioner [hide]	Fieldwork date	Sample size	Turnout					PNV		BNG			Ni/C	Lead
2000 general election	12 Mar 2000	—	68.7	44.5 183	34.2 125	5.4 8	4.2 15	1.5 7	1.1 4	1.3 3	—	0.8 1	0.5 1	10.3
 Eco Consulting/RTVE ^{[1][2][3]}	12 Mar 2000	?	?[a]	42.1 173/180	32.9 122/126	7.5 10/13	4.0 14/16	1.9 8/9	0.9 4	1.6 4/6	—	1.0 2	0.5 1	9.2
			?	41.6 168/173	34.6 130/134	7.0 10/12	4.0 14/16	1.6 7	0.9 4	1.5 5/6	—	0.9 1	0.4 1	7.0
 Sigma Dos/Antena 3 ^{[2][3][4]}	12 Mar 2000	?	?	41.1 162/173	36.8 133/143	6.6 9/11	4.3 14/15	1.8 7	1.1 4	1.3 4/5	—	1.1 2	? 0/1	4.3
 Opitel/Tele 5 ^{[2][3][4]}	12 Mar 2000	?	?	42.0 168/173	37.0 131/136	7.5 12/13	4.5 15/16	1.5 7/8	? 4	? 4	—	? 1/2	—	5.0
 Gallup ^{[5][6][7]}	8–11 Mar 2000	1,228	?	42.7 173/178	33.9 128/133	7.8 11/13	4.3 14/15	1.6 6/7	0.7 3/5	1.5 4/5	—	0.7 1	—	8.8
 CIS ^[8]	7 Mar 2000	?	78.5	43.7 176	33.7 123	8.7 17	4.2 14	1.7 7	0.9 3	1.3 4	—	0.9 1	1.2 2	10.0

Fuente: (Wikipedia, 2000).

La idea del proyecto es predecir el porcentaje de voto obtenido (primera fila de la tabla), por los principales partidos políticos (logos del encabezado, rectángulo discontinuo azul), en las distintas carreras electorales desde 1982 hasta las próximas elecciones en 2023. Para ello, alimentamos los modelos con la estimación de voto (matriz de valores del recuadro verde), que las principales casas encuestadoras (etiquetas de las filas, recuadro discontinuo naranja) van a asignar a cada partido. También usaremos los valores de las columnas “Fieldwork date”, “Sample size”, “Lead” y “Turnout”.

4.1.1. Extracción de los datos de Wikipedia

Para extraer los datos de Wikipedia de manera automatizada, se han combinado una serie de funciones y tablas. Creamos una primera tabla con los valores de fecha de todas las elecciones (fecha entera y cada uno de los componentes año, mes y día) y usamos el valor de año para generar automáticamente los links a consultar. Esto se debe a que cada elección se encuentra en una página de Wikipedia diferente.

Cabe destacar que este paso requiere una revisión manual debido a ciertos acontecimientos, por ejemplo, encontramos casos como 2015 y 2019 con más de una tabla a extraer o las elecciones de noviembre de 2019 y futuras con un link diferente. Además, en esta primera revisión se ha tenido que llevar a cabo una recodificación para los principales partidos ya que, como podemos ver en la tabla de la figura 5, su categorización de origen (formato png) no es compatible con nuestro software.

A continuación se muestra la figura en la que se recoge de forma visual la explicación del proceso detallado en este apartado. Con la tabla anterior y una serie de funciones, definimos la extracción de los datos de Wikipedia:

Figura 6: "wiki_info" tabla para automatizar la extracción de datos

	date_elec	year	month	day	wday	link	pol_parties
1	1982-10-28	1982	10	28	4	https://en.wikipedia.org/wiki/Opinion_polling_for_the_1982...	c("UCD", "PSOE", "PCE", "AP", "CIU", "FN", "PA", "PNV", "HB", ...)
2	1986-06-22	1986	6	22	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_1986...	c("PSOE", "AP", "UCD", "PCE", "CIU", "CDS", "PNV", "HB", "ER...
3	1989-10-29	1989	10	29	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_1989...	c("PSOE", "AP", "CDS", "UCD", "CIU", "IU", "PNV", "HB", "EE", ...)
4	1993-06-06	1993	6	6	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_1993...	c("PSOE", "PP", "IU", "CDS", "CIU", "PNV", "HB", "PA", "UV", "...)
5	1996-03-03	1996	3	3	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_1996...	c("PSOE", "PP", "IU", "CIU", "CDS", "PNV", "CC", "HB", "ERC", ...)
6	2000-03-12	2000	3	12	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2000...	c("PP", "PSOE", "IU", "CIU", "PNV", "CC", "BNG", "EH", "ERC", ...)
7	2004-03-14	2004	3	14	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2004...	c("PP", "PSOE", "IU", "CIU", "PNV", "BNG", "CC", "ERC")
8	2008-03-09	2008	3	9	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2008...	c("PSOE", "PP", "IU", "CIU", "ERC", "PNV", "CC", "BNG", "UPY...
9	2011-11-20	2011	11	20	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2011...	c("PSOE", "PP", "IU", "CIU", "PNV", "UPYD", "ERC", "BNG", "C...
10	2015-12-20	2015	12	20	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2015...	c("PP", "PSOE", "IU", "UPYD", "CIU", "EH-BILDU", "PNV", "ERC...
11	2015-12-20	2015	12	20	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2015...	c("PP", "PSOE", "IU", "UPYD", "CIU", "EH-BILDU", "PNV", "ERC...
12	2015-12-20	2015	12	20	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2015...	c("PP", "PSOE", "IU", "UPYD", "CIU", "EH-BILDU", "PNV", "ERC...
13	2015-12-20	2015	12	20	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2015...	c("PP", "PSOE", "IU", "UPYD", "CIU", "EH-BILDU", "PNV", "ERC...
14	2016-06-26	2016	6	26	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2016...	c("PP", "PSOE", "PODEMOS", "CS", "IU", "ERC", "CDC", "PNV"...
15	2019-04-28	2019	4	28	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2019...	c("PP", "PSOE", "UP", "CS", "ERC", "PDECAT", "PNV", "PACMA...
16	2019-04-28	2019	4	28	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2019...	c("PP", "PSOE", "UP", "CS", "ERC", "PDECAT", "PNV", "PACMA...
17	2019-04-28	2019	4	28	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2019...	c("PP", "PSOE", "UP", "CS", "ERC", "PDECAT", "PNV", "PACMA...
18	2019-04-28	2019	4	28	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_2019...	c("PP", "PSOE", "UP", "CS", "ERC", "PDECAT", "PNV", "PACMA...
19	2019-11-10	2019	11	10	7	https://en.wikipedia.org/wiki/Opinion_polling_for_the_Nove...	c("PSOE", "PP", "CS", "UP", "VOX", "ERC", "JXC", "PNV", "EH-B...
20	NA	NA	NA	NA	NA	https://en.wikipedia.org/wiki/Nationwide_opinion_polling_f...	c("PSOE", "PP", "VOX", "UP", "CS", "ERC", "MP", "JXC", "PNV", ...)
21	NA	NA	NA	NA	NA	https://en.wikipedia.org/wiki/Nationwide_opinion_polling_f...	c("PSOE", "PP", "VOX", "UP", "CS", "ERC", "MP", "JXC", "PNV", ...)
22	NA	NA	NA	NA	NA	https://en.wikipedia.org/wiki/Nationwide_opinion_polling_f...	c("PSOE", "PP", "VOX", "UP", "CS", "ERC", "MP", "JXC", "PNV", ...)

Fuente: Elaboración propia.

4.1.2. Transformación y preprocesamiento de los datos de Wikipedia

Durante la extracción hemos tenido que considerar una serie de preprocesamientos:

- 1) Excluir la primera fila de datos reales, pues los extraemos de otra fuente.
- 2) Recodificar el nombre de los partidos mostrados como logos.
- 3) Recodificar el nombre de las casas encuestadoras con impurezas (p.ej. la nota numérica).
- 4) Transformar los valores de “-” o “?” en NA
- 5) Transformar los valores de la columna “Fieldwork date” en fecha de inicio del trabajo de campo y fecha de fin, si está vacía la fecha de fin le imputamos la misma que de inicio.
- 6) Generar nuevas variables de fecha como el número de días de trabajo de campo y los días que quedan para la celebración de las elecciones.
- 7) Si las encuestas han acabado el día de las elecciones las denominamos pie de urna (“exit_poll”) y eliminamos encuestas de un solo día de campo sin ser pie de urna.
- 8) Descartamos encuestas sin tamaño muestral.
- 9) Al leer los datos, los valores de votos y escaños se mezclan, así que nos quedamos solo con el número hasta el primer decimal.
- 10) Pasamos a NA los valores por encima de 100 o negativos.
- 11) Eliminamos encuestas realizadas por partidos como PP, PSOE, VOX, etc.
- 12) Unimos las 4 tablas de 2015 , las 4 tablas de abril de 2019, la tabla de noviembre de 2019 y las del resto de años.

El anterior proceso de extracción es aplicado también para los datos de las próximas elecciones de tal forma que puedan unirse en la misma tabla. Todo esto deriva en el siguiente resultado:

Figura 7: "national_surveys_with_n" histórico de encuestas no pivotado

	poll_firm	field_date_ini	field_date_end	n	n_days_field	days_to_elec	exit_poll	turnout	type_elec	type_surv	year_elec	date_elec	lead	UCD	PSOE	PCE	AP	CIU	FN	PA	PNV	HB	ERC	EE
1	SOCOMETRIX	2019-11-09	2019-11-10	4000	2	0	FALSE	NA	national	national	2019	2019-11-10	6.2	NA	26.1	NA	NA	NA	NA	NA	1.0	NA	3.5	NA
2	CELESTE-TEL	2019-11-08	2019-11-10	1100	3	0	FALSE	67.8	national	national	2019	2019-11-10	6.6	NA	27.4	NA	NA	NA	NA	NA	1.8	NA	3.6	NA
3	GESOP	2019-11-07	2019-11-09	963	3	1	FALSE	70.0	national	national	2019	2019-11-10	7.3	NA	26.7	NA	NA	NA	NA	NA	NA	NA	2.8	NA

Fuente: Elaboración propia.

La tabla resultante posee cerca de 50 columnas, de las cuales 37 son partidos. Por lo que deberemos seguir procesando el dato para que esta tabla pueda ser útil. A continuación se siguen listando los cambios de procesamiento:

- 13) El mayor cambio será pivotar los partidos y su porcentaje de voto en dos columnas únicas. Con esto pasaremos de tener 36 columnas de partidos a 2 columnas: el nombre del partido ("party") y el porcentaje de voto asignado ("est_surv_vote"). La tabla resultante se aproxima a las 100.000 observaciones y tiene 14 columnas.

El problema de pivotar radica en incrementar cada una de las observaciones de origen 36 veces (partidos evaluados) por cada conjunto de encuestas listado en la fuente origen, llevando a un número de observaciones excesivo. De nuevo debemos seguir procesando el dato, para que esta tabla pueda ser útil:

- 14) Creamos un id de encuesta (compuesto por la casa y las fechas de inicio y fin de trabajo de campo) para eliminar duplicados
- 15) Eliminamos partidos que no aparecen en las encuestas
- 16) Calculamos el número de encuestas de cada casa
- 17) Extraemos el primer y segundo partidos considerados como ganadores y los añadimos como "lead_party" y "lead_party2"

Estos 17 pasos dan lugar a la siguiente tabla:

Figura 8: "national_surveys_longer" histórico de encuestas pivotado

	id_survey	poll_firm	n_surveys_firm	poll_surveys_firm	field_date_ini	field_date_end	n	n_days_field	days_to_elec	exit_poll	turnout	type_elec	type_surv	year_elec	date_elec	lead	lead_party	lead2_party	party	est_surv_vote
1	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	UCD	6.6
2	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	PSOE	43.6
3	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	PCE	6.0
4	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	AP	24.5
5	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	CIU	2.4
6	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	FN	1.0
7	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	PNV	1.8
8	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	HB	1.1
9	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	ERC	0.8
10	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	EE	0.7
11	SOEMASA-1982-10-16-1982-10-19	SOEMASA	7	0.416667	1982-10-16	1982-10-19	18255	4	9	FALSE	78.5	national	national	1982	1982-10-28	24.6	PSOE	AP	CDS	4.0








Fuente: Elaboración propia.

Como se puede observar en la figura anterior, tendremos por observaciones el porcentaje de voto que una encuestadora estima para cada partido en las diferentes encuestas (según el trabajo de campo), que ha realizado para cada carrera electoral, llegando a rondar las 12.000 observaciones.

4.2. Datos del Ministerio del Interior (censo y electorado)

Recordemos que, en las tablas de Wikipedia, disponemos de una primera fila con el supuesto dato real del resultado de la elección:

Figura 9: Resultado electoral según Wikipedia

Polling firm/Commissioner <small>[hide]</small>	Fieldwork date	Sample size	Turnout					PNV		BNG			NI/IC	Lead
2000 general election	12 Mar 2000	—	68.7	44.5 183	34.2 125	5.4 8	4.2 15	1.5 7	1.1 4	1.3 3	—	0.8 1	0.5 1	10.3

Fuente: (Wikipedia, 2000).

Extraer este dato para construir el histórico de resultados electorales puede ser conveniente y sencillo teniendo en cuenta el anterior proceso de extracción de datos. A pesar de ello, optamos por explorar una fuente más oficial y que puede abrir más alternativas de enfoque para futuros proyectos. Esta fuente es la del Ministerio del Interior y su portal digital INFOELECTORAL (Ministerio del Interior, Gobierno de España, recuperado 7 julio 2022).

En esta fuente, el dato de origen se divide en municipios, partidos y carreras electorales, resultando en una tabla de casi 1,5 millones de observaciones. La mayoría de los autores como en Kiko Llaneras (2021), recomiendan transferir el efecto de los municipios sobre el porcentaje de voto, ya que esto facilita mucho derivar la predicción a un problema de escaños. Esta recomendación es muy lógica y sensata si consideramos que el resultado electoral es consecuencia del voto por municipio y su contexto (Perdomo, 2008). En este punto debemos recordar uno de nuestros objetivos principales: “[..] propuesta de metodología propia y cercana al ámbito académico”. Operar con varias tablas de más de 1 millón de observaciones se aleja de los recursos convencionales en el ámbito académico, por lo que en un principio limitaremos el estudio a datos nacionales y estimar el porcentaje de voto real.

Los valores de mayor interés para un modelo estatal son: fecha, año, mes y día de las elecciones, el registro del censo, la participación, votos registrados, votos blancos, votos nulos, votos obtenidos por cada partido, código del partido, denominación, siglas, votos, datos oficiales y concejales obtenidos. A su misma vez, mantendremos el valor de municipio y el código de la comunidad ("codigo_ccaa") para poder agrupar las anteriores métricas y obtener el dato estatal.

4.2.1. Extracción de los datos de INFOELECTORAL

A pesar de que la extracción de los datos en este caso no es tan compleja, pues podemos proceder con la descarga directa de los datos electorales del estado por municipios, hemos observado otros aspectos que han afectado esta etapa. La codificación de origen de los partidos es la principal peculiaridad, ya que cada municipio puede registrar valores erróneos o absurdos al tratarse de un proceso en su origen manual. Además, cabe tener en consideración que se trata de un histórico que remonta a 1982 y que los partidos no tienen el mismo nombre u ortografía en todas las comunidades y/o municipios debido a que en España conviven más de ocho lenguas. Todos estos factores identificados dificultarán la estandarización de los datos obtenidos.

4.2.2. Transformación y preprocesamiento de los datos de INFOELECTORAL

Tal y como hemos introducido en el apartado anterior, las discrepancias en los datos de origen entre municipios obligan a llevar a cabo un preprocesamiento de estos datos:

- 1) Creamos id por municipio, compuesto por los códigos de municipio provincia y comunidad.
- 2) Con los valores de la tabla "wiki_info" extraemos todos los elementos de las fechas de las elecciones.
- 3) Añadimos un 0 a los meses anteriores a octubre para pasar de "m" a "mm"
- 4) Recodificamos los partidos.
- 5) Sacamos los votos totales sumando los votos de las candidaturas y votos en blanco.
- 6) Sacamos la suma de votos totales a nivel nacional por elección y partido.
- 7) Sacamos los datos relacionados con la participación de cada elección.
- 8) Añadimos los totales de cada partido y nos quedamos solo con un registro por año y partido.

De todo este preprocesamiento, debemos destacar el cuarto paso en el que recodificamos los partidos, sin renunciar a uno de los objetivos del proyecto que es proporcionar un proceso de extracción y transformación de los datos automático. Por ello, utilizamos una función con varios "string_detec" que buscará las palabras clave que pueden componer el nombre del partido y lo recodificará en las siglas usadas para la tabla de datos de encuestas. Cabe mencionar que automatizar este proceso conlleva un riesgo de pérdida de observaciones y puede haber casos de partidos importantes que no sean detectados y no se incorporen a nuestra base de datos. El análisis exploratorio posterior es crucial para detectar esa pérdida de individuos relevantes.

Finalmente, extraemos dos tablas de interés con estadísticas electorales: un histórico de censo y participación por elección ("census_national_by_year") y otro histórico de votos por elección y partido ("votes_national"). Entre estas dos tablas deberíamos encontrar todas las variables input relacionadas con datos electorales. Nuestra variable objetivo "porcentaje de voto" deriva del ratio entre votos nacionales y los votos contados. Procedemos al siguiente paso de transformación para poder llevar a cabo el cálculo:

- 9) Unir ambas tablas para poder obtener nuestra variable objetivo y un histórico de datos por carrera electoral (tabla "datos_reales"). Y computamos el ratio con nuestra variable objetivo "real_vote".

A continuación, un fragmento del histórico de datos electorales con 166 observaciones y 10 variables tal y como podemos observar en la figura 10.

Figura 10: "datos_reales", histórico de estadísticas electorales por carrera y partido

	party	total_votes	national_votes	date_elec	census	votes_census	null_votes	empty_votes	turnout_census	real_vote
1	OTROS	558	921400	1982-10-28	26923504	21166545	16	2	78.61735	4.353095888
2	CDS	558	614591	1982-10-28	26923504	21166545	16	2	78.61735	2.903596218
3	PNV	558	394668	1982-10-28	26923504	21166545	16	2	78.61735	1.864583946
4	PCE	558	598574	1982-10-28	26923504	21166545	16	2	78.61735	2.827924916
5	AP	558	5601761	1982-10-28	26923504	21166545	16	2	78.61735	26.465164721
6	FN	558	111314	1982-10-28	26923504	21166545	16	2	78.61735	0.525895936
7	EE	558	105488	1982-10-28	26923504	21166545	16	2	78.61735	0.498371369
8	HB	558	207591	1982-10-28	26923504	21166545	16	2	78.61735	0.980750519
9	PSOE	558	10145645	1982-10-28	26923504	21166545	16	2	78.61735	47.932456620
10	UCD	669	1420092	1982-10-28	26923504	21166545	16	2	78.61735	6.709134627
11	CIU	2060	762147	1982-10-28	26923504	21166545	16	2	78.61735	3.600715185
12	ERC	2060	136342	1982-10-28	26923504	21166545	16	2	78.61735	0.644139136
13	IU	61	1973	1982-10-28	26923504	21166545	16	2	78.61735	0.009321313
14	CC	2418	44648	1982-10-28	26923504	21166545	16	2	78.61735	0.210936646
15	OTROS	544	1435673	1986-06-22	28924755	20170715	26	3	69.73513	7.117610853
16	AP	544	5242217	1986-06-22	28924755	20170715	26	3	69.73513	25.989247283
17	HB	544	231533	1986-06-22	28924755	20170715	26	3	69.73513	1.147867093

Fuente: Elaboración propia.

En este punto ya tenemos nuestra tabla de datos electorales, otra tabla de datos de encuestas y una variable objetivo, todo lo necesario para construir un modelo. Por otra parte, hay tres aspectos que debemos cubrir antes de modelizar: la construcción de una tabla de datos “fundamentales”, hacer un promedio de encuestas y por último realizar un profundo análisis exploratorio para asegurar la correcta construcción de la base de datos.

4.2.2.1. Datos de contexto (fundamentales)

Como ya hemos comentado, el uso de variables de “fundamentales” o de contexto puede beneficiar al modelo predictivo y es una práctica muy recomendada por medios como *The FiveThirtyEight*. Por otra parte, no debemos olvidar que las variables fundamentales, al ser en muchos casos índices relativos o indicadores interpretables, pueden aportar ruido a la relación con el resultado electoral: “las condiciones económicas explican solo alrededor del 30 por ciento de la variación en el desempeño del partido” (Silver, 2020).

Entonces, recopilaremos datos fundamentales desde 1980 a fin de poder complementar la base de datos “core” para el estudio, siendo esta el histórico de datos electorales y encuestadoras desde las elecciones del 1982. Listamos y comentamos los bloques temáticos de las variables incluidas en nuestro conjunto de datos fundamentales:

- **Medio ambiente**, agregando indicadores sobre generación y consumo eléctrico, emisiones de CO2, reservas de petróleo etc. Con la inclusión de variables fundamentales, perseguimos explorar el efecto que tiene la percepción contextual del elector sobre el voto. Entonces, incluir variables relacionadas con medio ambiente, nos resulta fundamental según las encuestas de IPSOS: “en España, del 74% al 88% de los potenciales electores estaría considerando votar a un partido que reduzca la factura energética e impulse la transición energética” (Grupo Crecimiento Verde, 2019). En este caso la fuente de datos principal ha sido Datosmacro.com, pues el histórico de sus indicadores medioambientales precede el 1975. Hemos recolectado hasta 13 indicadores ambientales de los

cuales hemos cargado 7 en la tabla final. Identificaremos los datos de este bloque con el prefijo “env_”.

- **Demografía y sociedad.** Ya contamos con la población y participación electoral, por lo que podemos considerar Infoelectoral como una de nuestras fuentes. Más allá, intentaremos considerar variables como la esperanza de vida o casos de violencia de género. De acuerdo con el informe de IPSOS (2019), la justicia social es una preocupación emergente entre los votantes en España. Hemos recolectado hasta 31 indicadores socio-demográficos de los cuales hemos cargado 12 en la tabla final. Identificaremos los datos de este bloque con el prefijo “pobl_”.
- **Economía.** El comportamiento del votante es complejo de modelizar, pero el voto económico es un factor con un largo recorrido de estudio. ¿Qué sucede cuando los electores tienden a ser racionales o generando proyecciones de futuro en función de las vivencias y las propuestas de los partidos? Fearon (1998), Kuklinski y West (1981) y Lewis-Beck y Skalaban (1989), como se citó en Sáez L., J.L., y Jaime C., A. M. (2014), plantean que el voto económico es un instrumento para seleccionar al que el individuo considera el mejor de los candidatos de la carrera. A pesar de ello, estos autores también remarcan, que el voto económico considerando más de un indicador sucede entre una gran minoría de votantes, quienes no sólo comprenden el funcionamiento general del sistema económico, sino que también se molestan en trasladarlo a los programas electorales presentados en la carrera en cuestión. Es por ello, que los indicadores económicos elegidos, son básicos tal como la inflación, el IPC, etc. Cabe considerar que incorporar variables económicas en exceso puede generar problemas de ruido y colinealidad. En definitiva, hemos recolectado hasta 20 indicadores económicos, de los cuales hemos cargado 9 en la tabla final. Identificaremos los datos de este bloque con el prefijo “eco_”.
- **Política y Gobierno.** Se trata de indicadores sobre la gestión del gobierno, como se han administrado los presupuestos generales del estado o cuál ha sido el partido gobernante. La casuística de este bloque es incorporar variables que describan cual ha sido el gobierno precedente y su gestión. Las fuentes de datos para extraer estas variables han sido varias. Por una parte, hemos utilizado la fuente Wikipedia (Wikipedia, 2022) para definir el partido gobernante precedente. Por otra parte, para los datos de gasto público y corrupción, hemos utilizado Datosmacro.com (Datosmacro, s.f.). Hemos recolectado y cargado 11 variables en la tabla final. Identificaremos los datos de este bloque con el prefijo “gov_”.

4.2.3. Extracción de los datos fundamentales

En este caso la extracción de los datos es más sencilla, pues se dispone de muchas fuentes de datos con la posibilidad de descargarlos por indicadores directamente. La cuestión es, qué fuentes seleccionar y qué indicadores pretendemos usar.

Hay muchas posibles fuentes como el CIS o el INE, entre otros, y no se excluye la incorporación de algunos de sus indicadores. A pesar de ello, en una primera fase del proyecto, a fin de tener cierta armonía entre las fuentes de datos, principalmente usaremos “datosmacro.com” como fuente de datos de contexto. El motivo de esta elección reside en la variedad de cálculos que la organización ofrece para un mismo indicador y el alcance de su histórico de datos.

A diferencia de los datos electorales y de encuestas, hay muchos indicadores que tienen un histórico muy restringido, por lo que deberemos prestar especial atención a la pérdida de información que puede suponer la incorporación de variables con poco histórico. Por otro lado, los indicadores de contexto pueden ser calculados de muchas maneras diferentes, por lo que podemos estar acogiendo el indicador correcto en nuestro modelo pero su método de cálculo podría estar generando ruido y colinealidad. Por lo tanto vamos a revisar las posibles variables a añadir, su lógica y diferentes métodos de cálculo.

Entre las variables de contexto, vamos a elegir aquellas que resulten mínimamente interesantes según el problema planteado. Por muy interesante que parezca explorar la relación entre el Índice de Paz Global y el porcentaje de voto, debemos aplicar cierto criterio de razón y causalidad para seleccionar aquellas variables que, acorde con el estado del arte y conocimiento propio, puedan resultar útiles para nuestros modelos.

Los datos resultantes de la extracción fueron cargados en distintos archivos Excel según el bloque temático de los indicadores, obteniendo finalmente un conjunto de datos fundamentales cargado en nuestro entorno de trabajo. En este hemos identificado cerca de 75 indicadores a nivel nacional, de los cuales 31 han sido incorporados a la tabla final del proyecto.

4.2.4. Transformación y preprocesamiento de los datos fundamentales

Depositamos los datos extraídos en Excels por bloque temático, siendo estos: datos_sociodemograficos, datos_gobierno, datos_medioambiente, datos_comercio (grupo de indicadores económicos), datos_impuestos (grupo de indicadores económicos) y datos_economia (grupo de indicadores económicos). A su misma vez, cada Excel tiene distintas hojas con cada una de las tablas de interés. Ahora, debemos hacer una revisión de los indicadores de mayor interés y aplicar el preprocesamiento necesario para poder incorporar todas las variables fundamentales en una misma tabla. Este proceso se elaborará a través de la propia herramienta de Excel y agruparemos los indicadores a nivel anual en un archivo final para la importación al Software R.

Debemos considerar ciertas peculiaridades observadas en estos indicadores, por ejemplo, algunos se miden a nivel interanual en un plazo de 4 años como la tasa de inmigración o emigración. Otros indicadores como el IVA existen a partir del 1986 y muchos otros indicadores serán calculados con más prudencia en estos últimos años que en los años anteriores al 2008. Otro factor para tener en consideración es que un mismo indicador puede ser expresado en términos absolutos (PIB en millones de euros) o relativos (PIB per cápita o variación del PIB) y deberemos elegir aquella expresión que

más favorezca o bien la efectividad de los modelos (capacidad predictiva) o su eficiencia (interpretabilidad). Por ello, el proceso de selección y transformación de variables en este conjunto puede ser muy extenso y complejo.

Otro problema a valorar con la inclusión de estas variables es su planteamiento cronológico respecto a la predicción que se propone. Cuando hablamos de modelos de “scoring” pasa algo similar. Predecir el cliente que ya ha abandonado el servicio, no tiene sentido. Del mismo modo, usar indicadores socioeconómicos en un contexto en el que el votante aún no ha percibido, sería incongruente para predecir el voto que va a recibir un partido. Según el indicador, las lecturas de estas variables se pueden basar en los dos años anteriores a la elección (por ejemplo, desde noviembre de 2018 hasta noviembre de 2020 para esta elección) o, alternativamente, se acoge el último estatus previo a las elecciones.

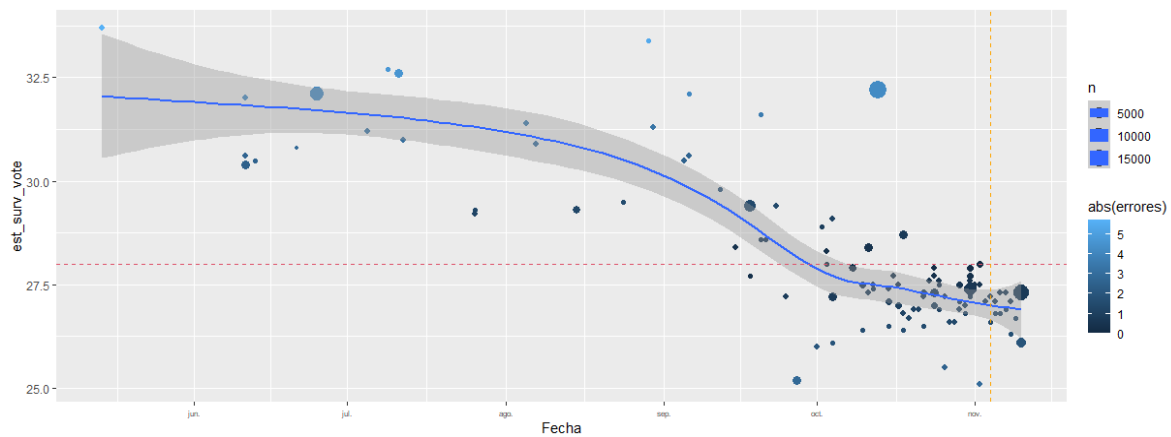
5. Promedio de Encuestas

A diferencia de la predicción electoral, el promedio de encuestas es una práctica común entre los medios. Tal como indica Endika: “Hay infinidad de promedios de encuestas circulando por ahí afuera [...], pero no todos tienen la misma metodología y transparencia” (Larrañaga, 2020). El objetivo del promedio de encuestas es generar una perspectiva sencilla sobre la tendencia de la intención de voto. Entonces, más que una predicción, es la fotografía temporal del estado de las encuestas (Larrañaga, 2020).

Antes de adentrarnos en el desarrollo de esta fase del proyecto, definiremos algunos conceptos básicos. En la introducción ya se diferenció entre predecir el Gobierno resultante y la Estimación de voto (Ortega, 2019). Pero debemos diferenciar también, entre la intención de voto y la estimación de voto. La intención de voto equivale a la respuesta del electorado a una pregunta como: “¿Si mañana se celebran elecciones, a qué partido votaría?” (Peinado, 2014). Habrá encuestados con una respuesta clara, pero aquellos que contestan “Indeciso” o “No contesta” entran en un proceso de imputación del dato, llamado “cocina”. Cada encuestadora hace este proceso de imputación de manera distinta en base a su propia metodología, añadiendo cierto nivel de estimación en los resultados finales ofrecidos. Cuando esa estimación de voto se aleja de lo que sucede en las elecciones celebradas, surge la crítica pública a la casa y el método de “cocina” aplicado. La diferencia entre la estimación de voto y el resultado real no tiene por qué ser una cuestión de “favoritismos” o malas prácticas (Masters, 2020). Ese sesgo se conoce como “House Effect”, es sistemático y a veces prolongado en el tiempo para determinados partidos y casas.

Otros factores fundamentales que pueden generar sesgo en las estimaciones de voto son: el número de encuestas y tamaño muestral, el número de días en el que se ha hecho el trabajo de campo, los días que queden para las elecciones (ver Figura 11), el diseño de la encuesta y la representatividad de la población encuestada. Por ello, añadiremos todas estas variables en nuestro estudio.

Figura 11: Estimaciones de voto y factores de sesgo (segundas elecciones del 2019, PSOE)



Fuente: Elaboración propia.

Debido a que las encuestadoras generan estimaciones, nosotros debemos recurrir al promedio de encuestas y el análisis exploratorio, para entender mediante gráficos y estadísticos esos sesgos sistemáticos y superarlos a la hora de construir nuestros algoritmos de predicción. Aprovechamos el anterior gráfico, para considerar el efecto de la incertidumbre. A menos días quedan para las elecciones, más oscuros son los puntos (menor es el error). A la hora de crear nuestros algoritmos, debemos excluir las encuestas a pie de urna, a fin de evitar causar sobreajuste indebido. Pretendemos estimar resultados futuros, un modelo basado en encuestas a 1 día, no es útil.

Hay muchos métodos de promedio de encuestas, pero el consenso dentro de nuestro estado del arte determina tres pasos genéricos en este proceso: [2.1] Recogida de encuestas, [2.2] Incorporar la incertidumbre, [2.3] Ajustar el “House Effect”.

5.1. Recogida de encuestas

Este paso quedó cubierto en la anterior sección en la que se ha detallado la construcción de la base de datos. Del mismo modo hemos cubierto gran parte del preprocesamiento y encuestas a incluir. Aún así, añadiremos un par de puntos para completar la explicación. “A priori no hay ninguna razón por la que una encuesta no deba formar parte del promedio, pero sí que hay unas pocas excepciones” (Larrañaga, TheElectoralReport, 2022). Esta misma cita se repite entre los mayores autores que abarcan el promedio de encuestas. Entre las excepciones a excluir, tenemos encuestas partidistas o patrocinadas entre militantes y partidos. Estas son encuestas con objetivos distintos al de informar a la población, ya que pretenden cumplir con funciones informativas para las estrategias de partido, por lo que no consideramos correcta la incorporación de estas en nuestros promedios. Tampoco incluimos las encuestas con una muestra poblacional insuficiente, del mismo modo tampoco las que no suelen presentar el “sample_size”. Por otra parte, cuando una encuestadora publica varios resultados de la misma encuesta con diferentes escenarios, utilizamos el escenario central o, de lo contrario, descartamos esas observaciones. Aquellas encuestas que en el análisis exploratorio demuestren evidentes signos de manipulación o malas prácticas, serán descartadas del promedio. Finalmente, también se han descartado aquellas encuestadoras que no representen más de un 1% del total de observaciones. Es básico

poder personalizar el “House Effect” de cada casa y si las encuestas realizadas son insuficientes, este indicador perderá significancia y nuestros modelos serán incapaces de interpretarlo correctamente.

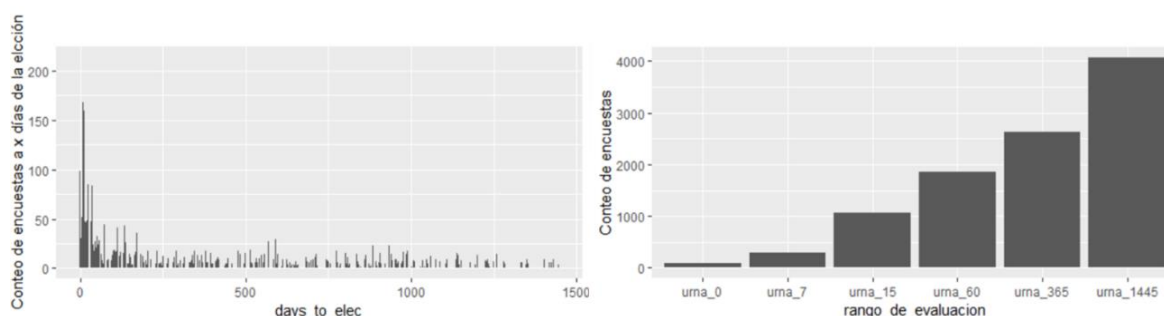
5.2. Incorporar la incertidumbre

Pretendemos hacer un modelo que sea más conservador cuando no hay elecciones convocadas y que se vuelva más agresivo a medida que vayamos acercándonos a una convocatoria electoral. Esto se debe a que la distancia de la fecha de elecciones puede afectar a la precisión y comportamiento de las encuestas, tal como representamos en la figura 11. Para poder incorporar la incertidumbre o sesgo que genera el tiempo restante para las elecciones, en las estimaciones definiremos los rangos de evaluación.

Estos rangos son ventanas temporales a las que se van a ajustar los promedios y determinan cuántos días de antelación a las elecciones se aceptaran. Debemos considerar que tenemos encuestas a 1445 días de las elecciones y que, a menor sea el número de días para las elecciones, más encuestas se elaboran (ver Figura 11 y 12). Con estos rangos pretendemos acometer las subidas y bajadas artificiales propiciadas por las encuestas “outlier”.

En nuestro caso, probamos con varios sets de rangos de evaluación, pero aquellos que más respetan la distribución definida son los mostrados en la siguiente figura:

Figura 12: Días para las elecciones y rangos de evaluación



Fuente: Elaboración propia.

Con los anteriores rangos crearemos flags binarios que identificarán, no solo en qué grupo se encuentra la encuesta, también ponderará los promedios que calculemos. En la fase de selección de variables observaremos atentamente, cómo se comportan los algoritmos al incorporar o no estas variables. Debemos recordar que para la construcción de modelos, deberíamos primero estudiar las encuestas a pie de urna y con un rango cercano a la fecha de las elecciones. Abusar de encuestas cercanas al evento electoral, puede concurrir nuestros modelos en sobreajuste.

5.3. Ajustar el “House Effect”

En esta fase, calcularemos una serie de promedios que informaran a los modelos de los comportamientos que tienen las encuestadoras y como suavizarlos. Para ello crearemos

un total de nueve tipos de promedio distintos, partiendo del más sencillo al más complejo tal y como se presentan en la siguiente tabla:

Tabla 2: Variables creadas como promedios

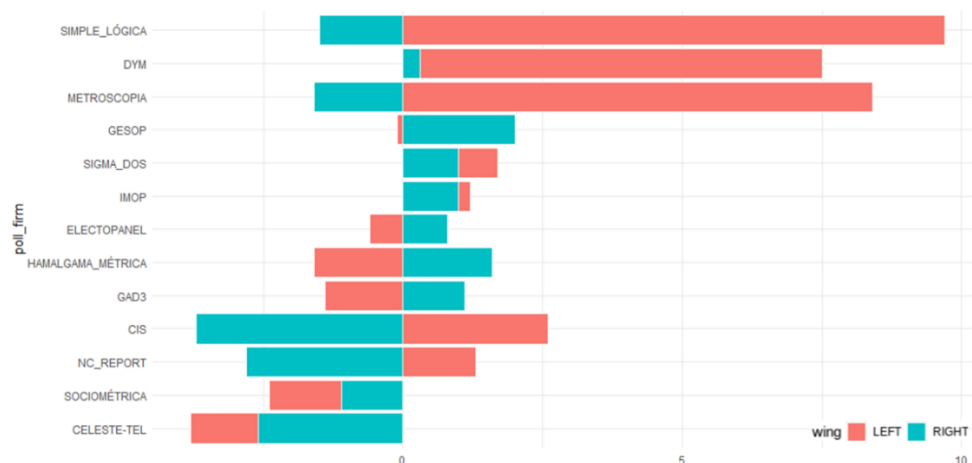
Promedio	Lógica
prom_general_partido	Promedio de todas las estimaciones de voto para un determinado partido.
prom_general_wing	Promedio de todas las estimaciones de voto para una ala determinada.
prom_casa_partido	Promedio de todas las estimaciones de voto de una casa encuestadora para un determinado partido.
prom_casa_wing	Promedio de todas las estimaciones de voto de una casa encuestadora para una ala determinada.
prom_carrera_partido	Promedio de todas las estimaciones de voto para un determinado partido y carrera electoral.
prom_carrera_wing	Promedio de todas las estimaciones de voto para una ala y carrera electoral
prom_carrera_casa_partido	Promedio de todas las estimaciones de voto de una casa encuestadora para un determinado partido y carrera electoral.
prom_carrera_casa_wing	Promedio de todas las estimaciones de voto de una casa encuestadora para una ala y carrera electoral determinadas.
house_effect_e	"Cuanto Infraestima o sobreestima una encuestadora a un partido sobre el promedio [...] se calcula por partido y carrera" $\text{prom_carrera_casa_partido} - \text{prom_carrera_partido}$
wing_effect_e	Cuanto Infraestima o sobreestima una encuestadora a un ala sobre el promedio de estimaciones. Se calcula por ala y carrera $\text{prom_carrera_casa_wing} - \text{prom_carrera_wing}$

Fuente: Elaboración propia.

Sobre el House Effect, se sabe que una de las formas para asegurar un modelo más robusto es calculando y ajustando los “House Effects” en nuestra tabla de datos de encuestadoras (Silver, 2012). El House Effect es relativo al promedio de la industria (prom_carrera_partido) y esta es una medida de centralidad. La centralidad no equivale a precisión, por lo que sólo juzgaremos la precisión respecto al porcentaje de voto real y no frente al promedio de la industria. Así, como la mayoría de los autores sugieren aplicar el “House Effect” por casa y partido, nosotros usaremos también un “House Effect” por posicionamiento del partido, pudiendo ser de Izquierda o Derecha exclusivamente.

A continuación ilustramos en la figura 13 el “wing effect” de la última carrera electoral a modo de ejemplo:

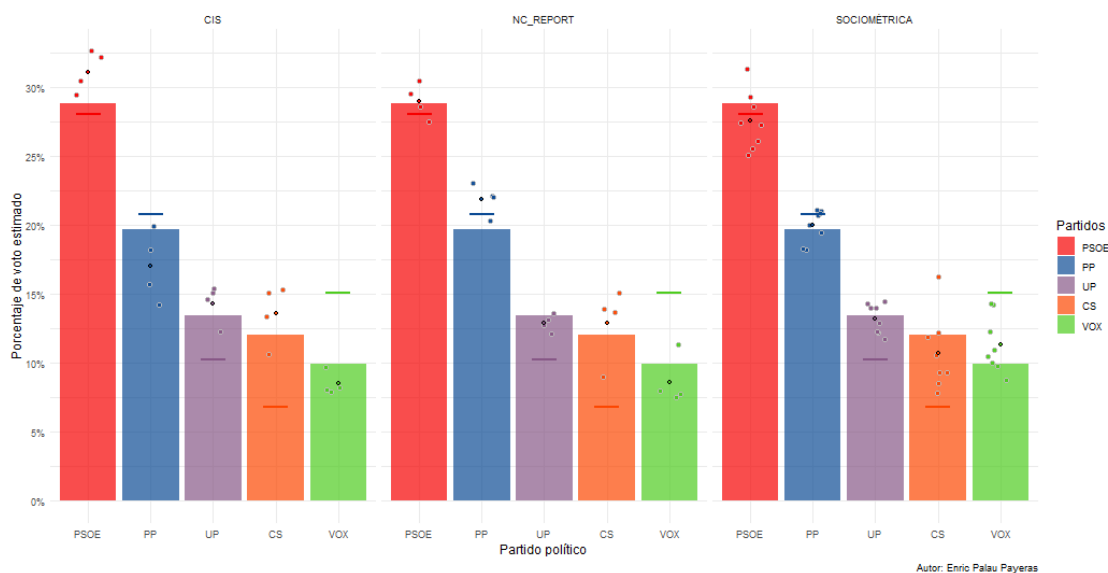
Figura 13: wing_effect_e de las encuestadoras (Carrera del 2019-11-10)



Fuente: Elaboración propia.

Por ejemplo, los house_effect y wing_effect desde las últimas elecciones se reflejan claramente en encuestadoras como NC Report o Gesop, tendiendo a mostrar un sesgo favorable al Partido Popular y/o al ala derecha. Por otro lado, el CIS mantiene un sesgo considerable en favor de Partido Socialista y al ala izquierda. Aunque en el caso del CIS, hablamos de un sesgo, que suele ir bastante relacionado con la coalición gobernante. En las últimas elecciones, la casa con un efecto más equilibrado fue Sociométrica, infravalorando casi por igual partidos de izquierda y derecha. Estos sesgos, no los consideramos perjudiciales para nuestros modelos, sino que además, si son continuados en el tiempo, serán más fáciles de predecir. Recordamos que, en este caso, no hablamos de sesgo real, hablamos de la desviación sobre el consenso de mercado (promedio de ala o partido por carrera). Reflejamos el anterior análisis en la siguiente figura a nivel de partidos, mediante algunos de los nuevos promedios que hemos creado para nuestra base de datos: Los puntos con contorno gris, son las estimaciones de cada encuesta; Los puntos con contorno negro, representan el promedio de la casa; La barra representa el promedio de mercado; Y la línea vertical equivale al voto observado.

Figura 14: Ejemplo del efecto de promedios sobre las estimaciones (Carrera del 2019-11-10)



Fuente: Elaboración propia.

6. Aplicación de la metodología SEMMA

Una vez tenemos la base de datos deseada para poder predecir, debemos aplicar la metodología SEMMA a fin de elaborar un estudio riguroso y efectivo, optimizando la posterior fase de modelización. A continuación abordamos individualmente cada una de las etapas de esta metodología, explicando lo que hemos hecho en cada una de ellas.

6.1. Sample

En primer lugar, aislaremos los datos de 2023 (test_2023), pues carecen de valores en nuestra variable objetivo y nos servirán como “puesta en práctica” de los modelos sobre un escenario real. Teniendo aún cerca de 11.036 observaciones, podemos permitirnos establecer como norma de partición destinar el 80% de datos a train y un 20% a test. Dentro de las 7.348 observaciones de entrenamiento, tendremos datos que servirán como validación para los modelos en función de las técnicas de control que definamos. De esta manera, nos quedaría la siguiente BBDD para el proceso SEMMA:

Tabla 3: Resumen de BBDD para metodología SEMMA

Tabla	Observaciones	Función
semma	9.185	Base de datos Input
split_semma	9.185	Base a particionar
train_semma	7.348	Base de entrenamiento
test_semma	1.837	Base de test
test_2023	1.851	Base de datos de 2023 (test real)

Fuente: Elaboración propia.

Al tratar este problema con una estratificación sencilla, abandonamos al 100% el enfoque del problema como una serie temporal. Indagaremos más en este aspecto en la próxima sección de análisis.

6.2. Explore

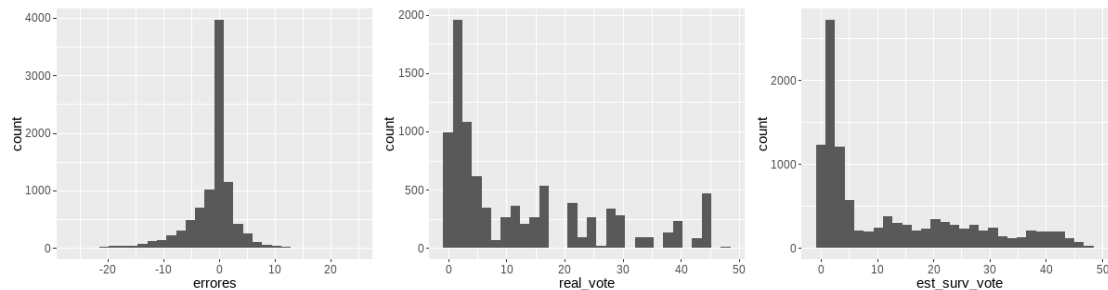
Revisaremos la correcta carga de los datos, concretamente fijándonos en que tomen valores lógicos. En este apartado es importante entender la naturaleza de las variables y de los individuos. Al tener 77 variables y 11.036 observaciones, limitaremos el análisis documentado a estudiar el comportamiento de las encuestadoras en el histórico. Después trataremos de entender aquellas variables que puedan ser más o menos útiles para el problema planteado. También aseguraremos que en la tabla final no queden: errores en los valores, valores atípicos y/o datos faltantes. Justificaremos cada una de las decisiones tomadas, pero limitaremos la documentación gráfica para respetar el límite de páginas establecido para este trabajo de investigación.

6.2.1. Análisis Gráfico (visualización de datos)

Estas son las distribuciones que toman el error, la estimación de voto y el voto real en el total de nuestra tabla (ver figura 15): El error, tiene una distribución bastante simétrica y centrada. El error MAE (Mean Absolute Error) de las encuestadoras en todas

las elecciones generales de España varía entre 2 y 4 puntos porcentuales. Esto lo interpretamos como una desviación de las encuestas con respecto a los resultados oficiales por un margen de hasta 4 puntos porcentuales.

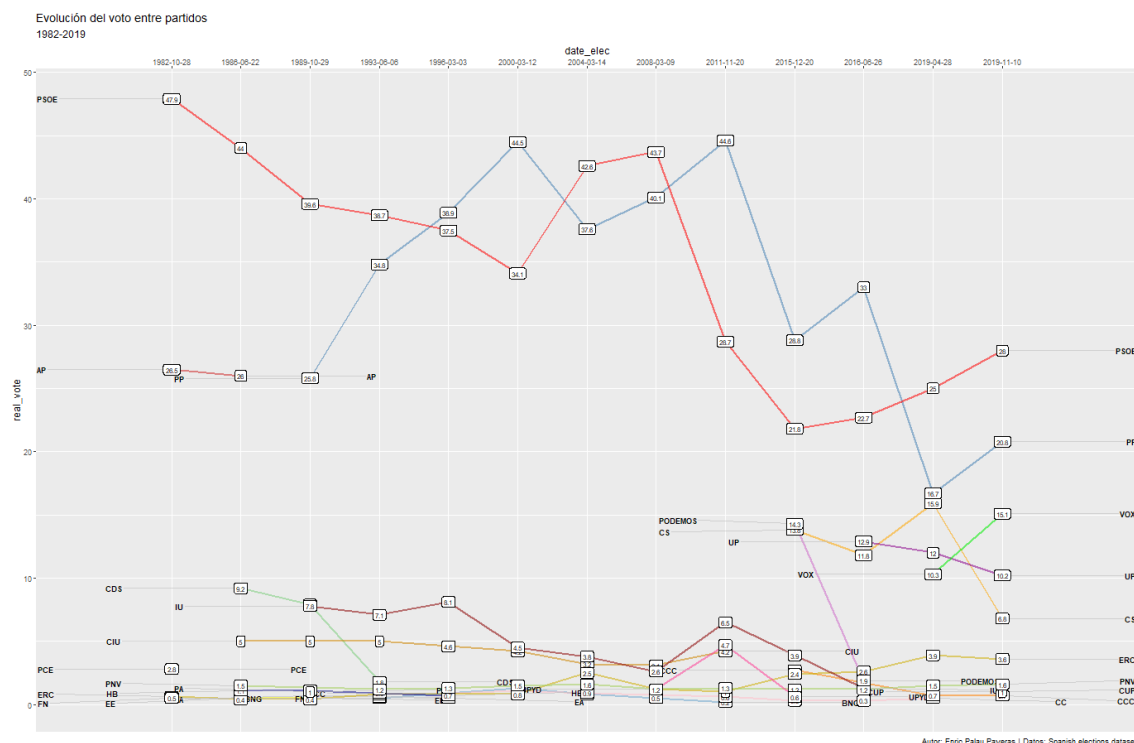
Figura 15: Errores, estimaciones de voto y voto real (distribución en el histórico)



Fuente: Elaboración propia.

Como podemos observar, el análisis exploratorio convencional no es aplicable a este problema. A pesar de tener dos variables objetivo-continuas, el porcentaje de voto real se repite tantas veces como encuestas hayan valorado al partido en cuestión. Lo mismo pasará entre las variables de contexto. Es decir, no tratamos con una serie temporal, pues cada carrera es única y a su misma vez en las 14 carreras, al tener una relación 1:N entre la tabla de datos de encuestas y la de datos electorales, se genera esa “duplicidad” de las variables continuas como el voto real. Como solución, entrenaremos con 14 carreras, pero no podemos hablar de una serie temporal pues el orden de los individuos nos es irrelevante. Proponemos estudiar el voto de los partidos, a través del histórico (Figura ampliada en los anexos, punto C) para entender su evolución:

Figura 16: Voto Real por carrera y partido (evolución sobre el histórico)



Fuente: Elaboración propia.

Con el anterior gráfico de pendientes y series temporales, pretendemos estudiar el comportamiento del voto entre carreras y partidos. El análisis exploratorio convencional suele centrarse en estudiar la relación entre variables, pero aquí debemos considerar la relación entre individuos. Como ya hemos inferido el voto, este está condicionado por sí mismo y no sólo eso, sino que además muchas de las variables seleccionadas dependen también de la forma de gobierno resultante. Además, en este caso apreciamos cierto nivel de estacionalidad entre el voto del PP y del PSOE. También se identifica como la tendencia de este factor ha seguido, aunque a menor escala, desde las elecciones de 2016 con la aparición de Ciudadanos, VOX, Podemos y Unidas Podemos. Como se comentó en la introducción del trabajo, el voto de los principales partidos en España ha presentado un comportamiento específico. Con el tiempo hemos pasado de un sistema casi bipartidista a uno en el que partidos como VOX o UP son esenciales para asegurar la mayoría mediante pactos al PP o al PSOE. Por lo tanto, el partido y la carrera electoral son dos variables cruciales para la estimación de voto.

Para hacer un análisis sencillo, corto y a la vez representativo, hemos usado varios gráficos multivariante con ejemplos tal como las figuras ilustradas en la anterior sección de promedio de encuestas, las cuales, también consideraremos como parte de este análisis exploratorio. Pero, adicionalmente incluiremos como herramientas de análisis algoritmos sencillos (como el árbol de decisión) e interpretables y los propios métodos de selección de variables.

6.2.2. Análisis Gráfico mediante modelos de árbol

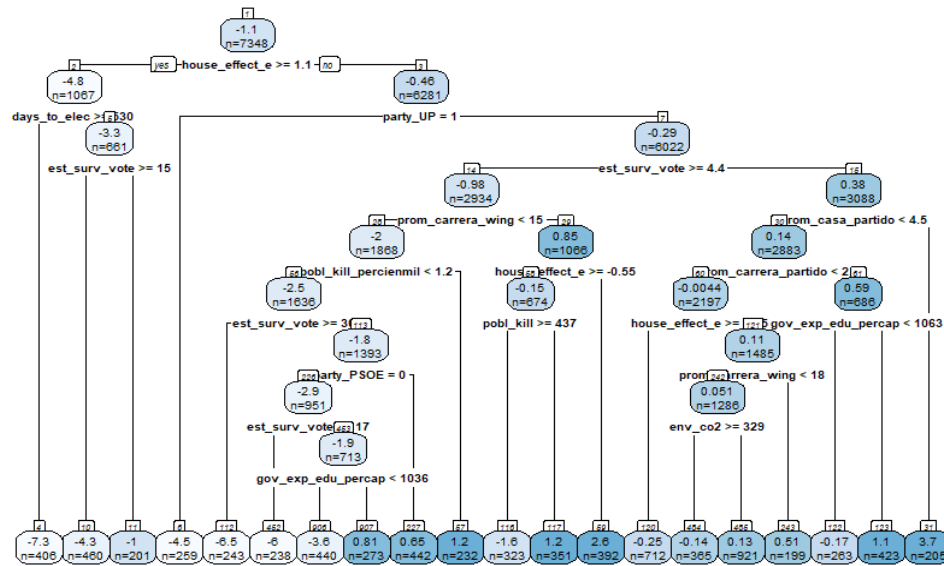
El uso de algoritmos sencillos para el análisis exploratorio es muy recomendable en casos dónde la interacción entre las variables es compleja. Con este fin introducimos brevemente nuestro primer algoritmo en la sección de modelos, los árboles de decisión (figura 17).

Los árboles de decisión son modelos predictivos basados en reglas binarias (generalmente, pero puede modificarse) con las que se van dividiendo las observaciones, según la variabilidad causada en la variable respuesta a predecir. Estos modelos permiten introducir variables numéricas, categóricas e incluso con datos faltantes. Es más, son modelos con pocos parámetros. Estos rasgos los hace modelos fáciles de aplicar e interpretar. Con este modelo, no pretendemos encontrar las mejores predicciones, por ahora simplemente usaremos este modelo como herramienta de exploración y ejemplo introductorio a la fase de modelización. También nos permitirá adelantar una serie de ideas básicas para los próximos modelos que también se basan en la idea de los árboles (Bagging de Árboles, Random Forest, GBM). Con finalidad exploratoria, prestaremos especial atención a la estructura del árbol (figura 17) y la importancia de las variables (figura 18).

El concepto del árbol parte del nodo raíz, el cual representa el 100% de la muestra de entrenamiento. Este se extiende en dos ramas que van a generar dos subnodos (nodos hijo). La reiteración de este proceso es finita y acaba en las hojas o nodos terminales. Una vez construido el árbol, la predicción se realiza en cada nodo terminal. Lo que nos

permite hacer de este modelo una herramienta de exploración, son las normas de decisión que se toman entre nodos para llegar a la predicción final.

Figura 17: Ejemplo de árbol de decisión.

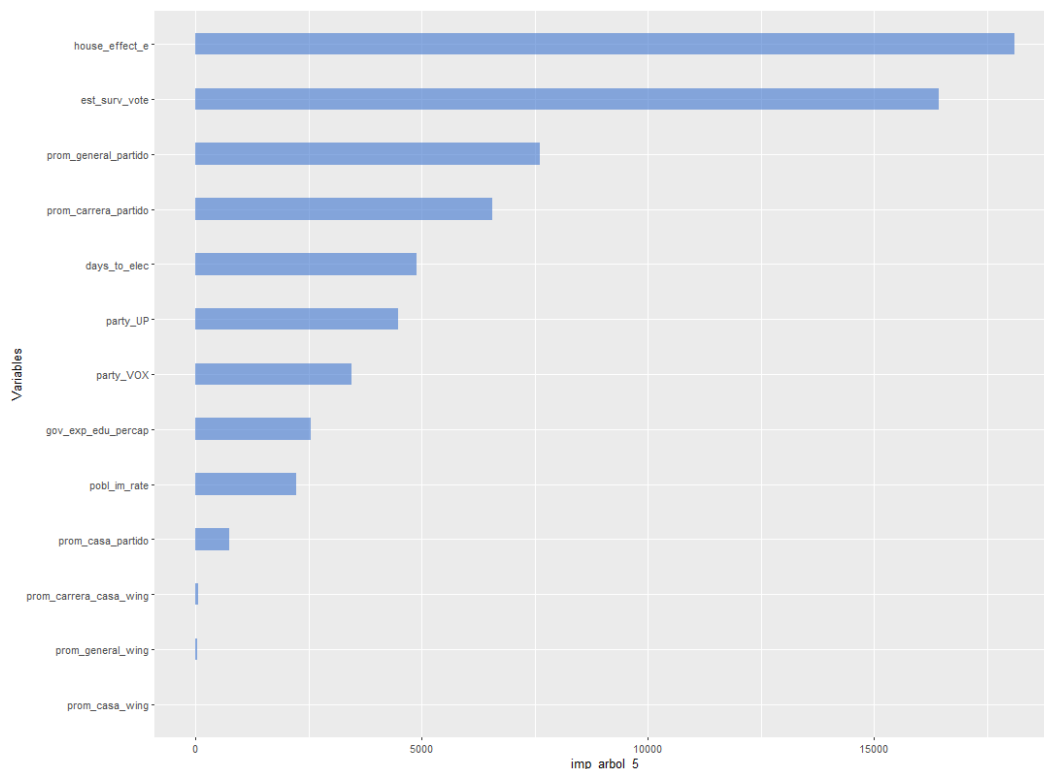


Fuente: Elaboración propia.

En nuestro ejemplo, cada nodo padre se divide en dos ramas y dos nodos hijos. De 7.348 observaciones, el árbol divide dos grupos según el “House Effect”. Si éste es mayor o igual a 1.1, el árbol realiza la predicción con dos decisiones más como máximo: ¿quedan más de 530 días restantes para la elección? y ¿es la estimación de voto superior al 15%? Con esto, el árbol ya segmenta un 14,5% de las observaciones. Obviamente, no será el modelo más efectivo, pero ofrece mucha interpretabilidad e información básica. La reiteración de las variables relacionadas con las encuestas hacen obvio lo básicos que son estos atributos para predecir. A pesar de ello, vemos como factores como la muestra poblacional (altamente defendido en el estado del arte) no ha sido considerado como un factor de decisión. Destacan también las variables de promedios de encuestas, siendo el “House Effect” la primera norma de decisión, lo cual contrasta lo planteado en la anterior sección de promedios. El grupo de variables con menos presencia son las variables fundamentales. Aún así, parece que el uso de estas era muy conveniente. Nuestro árbol llega a crear hasta 5 normas de decisión basadas en variables de gestión de Gobierno, sociedad y medio ambiente. En la figura 18, vemos el ranking y selección de variables resultante en otro árbol.

Podemos apreciar que este ranking resultante es muy similar al del anterior análisis. El “House Effect” es el principal factor de decisión, seguido de la estimación de voto y otros promedios. Eso deja en primer lugar las variables de encuestas y promedios. Hay partidos en particular (ambos principales partidos de la actualidad) que pertenecen a la extrema de ambas alas y son factores de gran importancia en este modelo. Finalmente seguimos teniendo varias variables fundamentales relacionadas con la población y gestión de gobierno que toman una importancia considerable. Disponemos el listado de variables completos en los anexos (Anexos, punto 2).

Figura 18: Filtro de variables, ejemplo árbol de decisión



Fuente: Elaboración propia.

6.3. Modify (depuración y modificación de las variables)

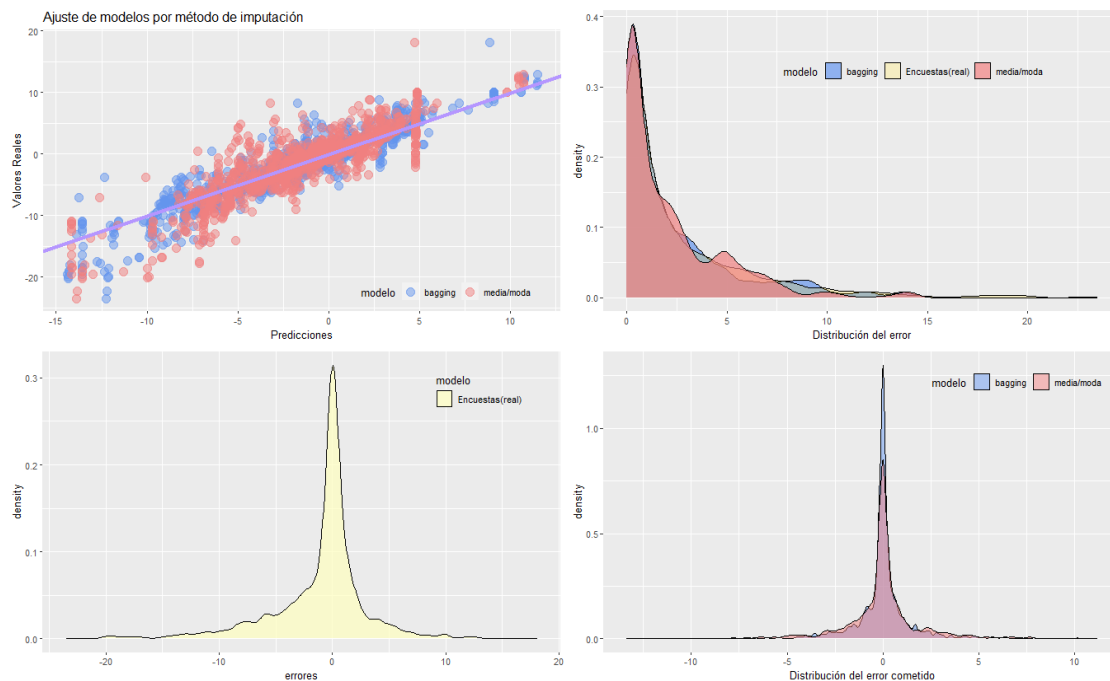
La transformación de variables para su correcta integración en la tabla final ya la aplicamos en la fase de ETL, por lo que en esta fase nos centraremos en el tratamiento de datos faltantes. A pesar de habernos librado de algunas encuestas “outlier” o poco representativas y de haber tratado las variables fundamentales con el planteamiento cronológico oportuno, seguiremos teniendo datos faltantes en la tabla final. Tener datos “missing”, por defecto, nos supone una pérdida de información, ya que tenemos menos observaciones válidas. Algunos modelos funcionan igual de bien a pesar de su presencia, por ejemplo, los árboles. Aún así, vamos a tratarlos para ganar interpretabilidad y armonía entre todos los modelos construidos. Podemos optar por: la eliminación de estos datos (tanto de variables como de observaciones), recategorizarlos (discretización en las variables continuas y/o aplicar una categoría válida en las categóricas) o aplicar la imputación.

En nuestro caso, como no deseamos perder más información, empezaremos estudiando la viabilidad de imputar estos datos. A la hora de imputar podemos usar: estadísticos de localización (según los estadísticos de simetría, desviación, etc.), modelos o la imputación aleatoria (opción descartada pues queremos controlar el proceso). Por lo tanto, estudiaremos si es más conveniente imputar por estadísticos de localización o por modelos cada una de las variables afectadas.

Puesto que tenemos muchas variables con valores “missing”, no podremos exponer un estudio de la imputación individual (siguiendo la distribución) por cada variable. Lo que

haremos es crear dos conjuntos de modelos: uno de datos imputados con estadísticos de localización y otro de datos imputados por modelos (en nuestro caso, del paquete `recipes` con la función `step_impute_bag`, de R). Así, podremos comparar como serían las distribuciones de las variables entre tipos de imputación y también podremos ver como ajustan las predicciones de un mismo algoritmo con los dos conjuntos de imputaciones.

Figura 19: Estudio comparativo entre métodos de imputación



Fuente: Elaboración propia.

Tras aplicar las dos técnicas de imputación, vemos que el MAE conseguido en los modelos con variables imputadas por estadísticos es ligeramente peor que el de los modelos con imputación por Bagging, siendo respectivamente de 1.2 y 0.9. El MAE equivale a la media en términos absolutos de las distancias que tenemos entre cada uno de los puntos del gráfico superior izquierdo y la línea continua lila.

Nos adentraremos en el algoritmo y criterio de bondad de ajuste elegidos en la próxima sección de modelos. Realmente, buscamos obtener una distribución del error estimado lo más similar posible a la del error observado. Por lo tanto, los cuatro gráficos mostrados en la figura 19, nos sirven para evaluar dicho objetivo. El gráfico inferior izquierdo, es un diagrama de densidad con el error observado (representado en amarillo) y es esa distribución la que buscamos replicar. El gráfico superior derecho, contrasta la similitud del error en términos absolutos, entre el error cometido por las encuestas y el error estimado por nuestros modelos. Podemos observar cómo las áreas entre el modelo de variables imputadas por Bagging y la de las encuestas, es ligeramente más pequeña que el área entre modelos con imputación por estadísticos y el error cometido por las encuestas. Finalmente, tenemos el gráfico inferior derecho. Este diagrama de densidad evalúa el error cometido por los modelos a la hora de predecir el error de las encuestas. Buscamos el método que consiga mantener la mayoría de las estimaciones en error = 0 y ese sería el conjunto imputado por Bagging. En conclusión,

procedemos a imputar las variables con valores “missing” mediante Bagging en vez de media y moda.

También recurriremos a transformaciones de las variables relacionadas más con necesidades que presentan los algoritmos que vamos a construir. Por ejemplo, a la hora de entrenar modelos basados en redes neuronales, debemos transformar los datos, al menos, convirtiendo en “dummys” las variables nominales y estandarizando las numéricas. Esto no supone un problema para otros algoritmos como los árboles de decisión y por ello que dejaremos nuestro conjunto de datos con estas transformaciones.

“Dummificar” o convertir a binarias (“one hot encoding”) es la transformación que aplicamos a nuestras variables categóricas y consiste en transversar las categorías como columnas, dejando como valor de fila unos y ceros. Si la fila corresponde a una valoración del PP, la columna “party_PP” tendrá un 1, de lo contrario será 0. Ya hemos recurrido a técnicas similares para la creación de rangos de evaluación.

Normalizamos las variables numéricas porque su escala y la magnitud de su varianza pueden influir de manera “injusta” sobre algunos modelos, dejando al modelo predictores que, aunque no sean los que más relación tienen con la variable respuesta, tengan mucho peso sobre las predicciones.

Con esto, las modificaciones previas y las variables de nueva creación de secciones anteriores, tendríamos la “receta” paso a paso para construir una base de datos útil para la elaboración de máquinas de aprendizaje enfocadas a la predicción electoral.

6.4. Model

Nuestro procedimiento para la construcción de modelos, por norma genérica, consistirá en usar el conjunto de entrenamiento para construir y parametrizar nuestros modelos, y utilizar un conjunto test para evaluar los resultados obtenidos en función de la bondad de ajuste elegida.

En el entrenamiento primero buscamos combinaciones hiper parámetros adecuadas mediante la validación cruzada simple. En un principio sería mejor usar la validación cruzada repetida de entrada. Pero con el volumen de datos tratado, supone un coste computacional que no podíamos asumir. Por lo que optamos por el siguiente patrón de estudio. En la primera validación cruzada simple: se subdivide el conjunto train_semma (en 4 grupos siempre) para que sean utilizados alternamente en cada iteración a modo de validación. Con esta primera validación de bajo coste computacional tendremos una selección de parámetros que serán optimizados con las técnicas oportunas de cada algoritmo. Adicionalmente, para las mejores selecciones de hiper parámetros, aplicamos la validación cruzada repetida. Similar a la simple, pero no sólo hacemos subgrupos del conjunto de entrenamiento, sino que reiteramos el proceso tantas veces como consideremos oportuno. Para aplicar cierta armonía, siempre usaremos 4 grupos y 10 iteraciones. Nuestro objetivo es hacer comparables los resultados de los diferentes modelos del estudio. Luego usaremos el conjunto test (no test_2023) para evaluar cómo

funcionan nuestros algoritmos con observaciones nuevas, de las que no ha aprendido anteriormente. Finalmente, usaremos el conjunto de test_2023 para reflexionar y estudiar la puesta en producción del modelo. Por lo que en total, nuestro proceso de modelaje pasa por 4 validaciones distintas, entre las cuales nuestra mayor prioridad es mantener el equilibrio entre sesgo y variabilidad sin incurrir en sobreajuste.

6.4.1. Criterio de bondad de ajuste elegido

Entendemos por criterio de bondad de ajuste la métrica que va a indicar como de bueno es un algoritmo, prediciendo la variable objetivo (error de las encuestas). El criterio óptimo depende del caso de estudio y la intención del investigador. Deberíamos estudiar varios criterios, pero por cuestiones de complejidad, en este proyecto se atenderá primordialmente un único criterio. Al tratar un problema de regresión, tenemos una gran variedad de criterios, entre ellos: MSE, RMSE, MAE, Rsquared.

El Error Cuadrático Medio (MSE), equivale a la media de los errores al cuadrado (MSE) y es un criterio excelente para evaluar los modelos, si no fuera por estar expresado en unidades al cuadrado. La solución es el Root Mean Squared Error (RMSE), la raíz cuadrada del MSE. Interpretamos el RMSE como la desviación estándar de la varianza inexplicada y tiene la ventaja de estar en las mismas unidades que la variable objetivo. Por otro lado, tenemos el Mean Absolute Error (MAE), valor absoluto de la media de los errores. A diferencia del Mean Squared Error, el Mean Absolute Error es más restrictivo, penalizando las desviaciones grandes ya que deja los errores al cuadrado. De lo explicado, podemos ver que RMSE penaliza la predicción del último valor más fuertemente que MAE. En general, RMSE será mayor o igual que MAE.

Si el modelo predice igual de bien o igual de mal sobre todas las observaciones (lo cual depende en gran parte de las variables), éste se ve más aventajado en el MSE y RMSE frente al MAE. El MAE es más favorecedor para modelos que tengan bajo error de predicción pero que predigan muy mal algunas observaciones concretas. Esto se reflejará al evaluar los modelos del proyecto y hemos usado la guía de Rodrigo, (2020).

Atendiendo nuestro problema, elegimos el MAE. Entre cerca de 11.000 encuestas sobre los mismos partidos, es de esperar que cualquier modelo vaya a tener un error generalmente bajo pero inestable y muy caracterizado en función del partido, la encuestadora y el momento en el que se hicieron las encuestas (entre muchos otros factores por descubrir). El criterio debe castigar los errores más “outlier”. Con estos argumentos dejamos clara nuestra elección del MAE como criterio de bondad de ajuste. Por otro lado, es la métrica más usada en el sector, por lo que nos dará mucha facilidad a la hora de comparar e interpretar resultados. El error MAE de las encuestas realizadas a menos de dos meses de las elecciones generales celebradas en España el 10 de noviembre de 2019 (por ejemplo) fue de 4,4 puntos porcentuales. Sabiendo que estimamos el error de las encuestas en España, teniendo un histórico de las elecciones desde 1982 y que el error absoluto medio de los promedios de encuestas ronda los dos puntos por partido llegando casi a siete puntos en partidos mayoritarios. A la hora de parametrizar y mejorar nuestros modelos, este puede ser un primer objetivo para superar.

6.4.2. Árboles de decisión

Ya cubrimos el concepto del árbol de decisión en la sección de exploración. También vimos el concepto del árbol y cómo se comporta con las variables. En esta sección nos centraremos en los parámetros de estos algoritmos y su eficiencia a la hora de predecir. Concretamente, usaremos los árboles de regresión, que son el subtipo de árboles de predicción que se aplica cuando la variable respuesta es continua. En términos generales, en el entrenamiento de un árbol de regresión, las observaciones se van distribuyendo por bifurcaciones (nodos) generando la estructura del árbol hasta alcanzar un nodo terminal. Cuando se quiere predecir una nueva observación, se recorre el árbol acorde al valor de sus predictores hasta alcanzar uno de los nodos terminales. La predicción del árbol es la media de la variable respuesta de las observaciones de entrenamiento que están en ese mismo nodo terminal.

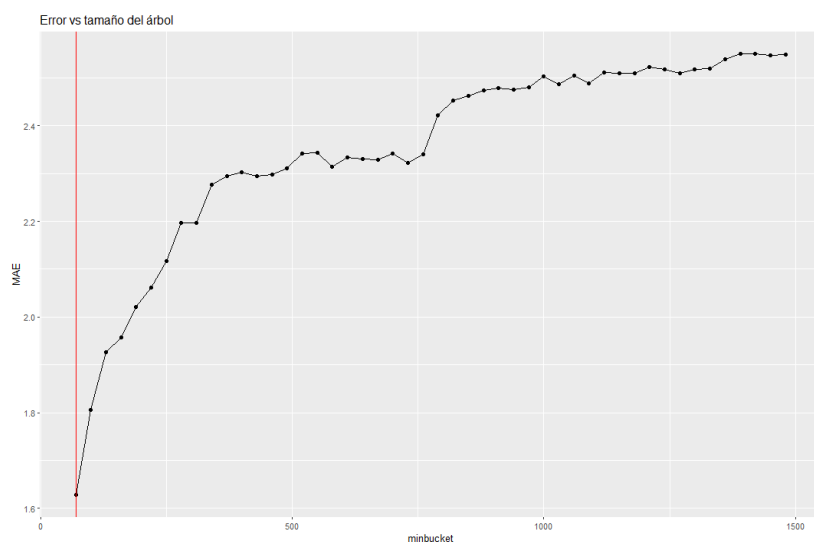
6.4.2.1. Parámetros y rangos estudiados: Early Stopping y Prunne

Procedemos a la primera exploración de parámetros con la validación cruzada. A continuación listamos los diferentes parámetros de los árboles y cuál ha sido nuestra selección:

- **method:** criterio de división del árbol y usamos F de Snedecor (“annova”), que toma la división que varíe más la media de la variable dependiente entre los grupos (mayor F).
- **cp:** es el parámetro de poda y penaliza el número de nodos finales a fin de evitar el sobreajuste. Sin este parámetro, el árbol tiende al máximo sobreajuste. Primero lo fijamos $cp=0$ y en segundo lugar exploraremos el uso de la poda (“pruning”).
- **minbucket:** número de observaciones mínimas en cada nodo final o en otras palabras, complejidad del árbol. Si las observaciones mínimas de los nodos finales son muy bajas, el árbol puede estar sobre ajustando (árbol complejo). Por lo contrario, si es alto, los árboles serán más sencillos y tienden a tener un error o sesgo muy elevado. Primero, aplicamos “Early Stopping” con “minbuckets” desde un 1% (70) hasta un 20% (1500) de las observaciones. Segundo, dejamos árboles grandes para aplicar la poda.
- **maxsurrogate:** el árbol no usa todas las variables, va seleccionando las mejores (método de selección “embedded”). Además, hay variables que usa más que otras y permite elaborar un índice de importancia, basado en la suma de la mejora en cada división de las que ha participado. Marcando este parámetro a $maxsurrogate > 0$ filtramos las variables no usadas.

Con el rango de parámetros que hemos comentado, exploramos varias selecciones óptimas para finalmente construir y evaluar nuestros modelos. Ilustramos como ejemplo la exploración de parámetros con “Early Stopping” en validación cruzada. Comparamos sesgo en MAE con el parámetro de complejidad del árbol (“minbucket”).

Figura 20: Ejemplo de Early Stopping (arbol_1), validación cruzada simple (4 grupos)



Fuente: Elaboración propia.

Con “Early Stopping”, partimos de un “minbucket” de 70 (coincidiendo con el parámetro óptimo). A fin de evitar caer en un mínimo local, repetimos esa exploración en rangos de valores más cercanos (saltos de “minbucket” de 2 en 2 entorno “minbuckets” de 70 y 100) y se nos sugiere de nuevo el “minbucket” de 70 y 150 adicionalmente. Por otro lado, la exploración de parámetros con poda se aplica con otro algoritmo, pero la idea es la misma: intentamos hallar el árbol más simple (de menor tamaño, mayor “bitbucket”) que logre los mejores resultados de predicción. Resumimos los modelos resultantes en la tabla 4 de la siguiente sección .

6.4.2.2. Evaluación y comparativa de modelos entrenados

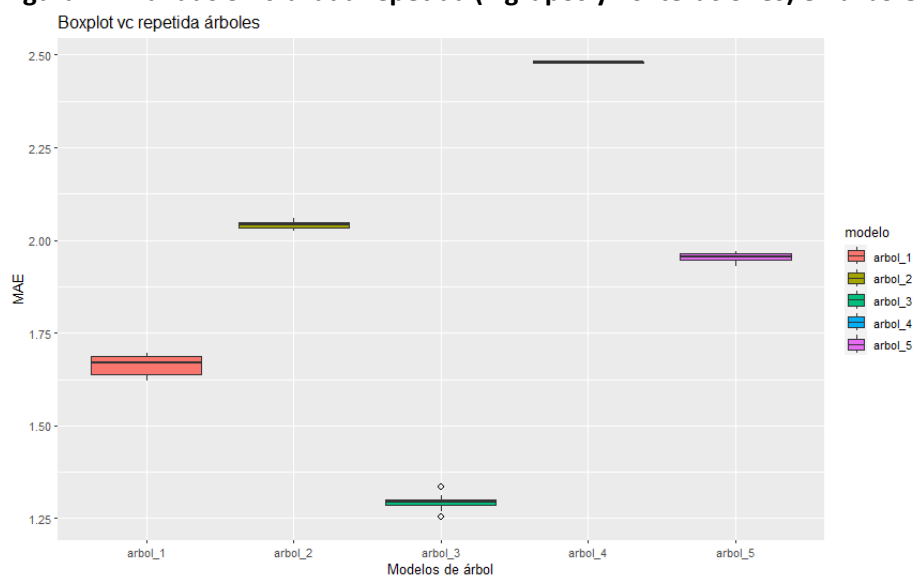
Tabla 4 : Modelos de árbol, parámetros y resultados en validación cruzada repetida

Model name	Minbucket	MAE	RMSE	R^2
arbol_1	70	1.629	2.801	0.567
arbol_2	199 (poda)	2.102	3.37	0.371
arbol_3	36 (poda)	1.31	2.43	0.674
arbol_4	1020	2.479	3.95	0.139
arbol_5	150	1.904	3.207	0.432

Fuente: Elaboración propia, 2022

Procedemos a explorar el sesgo y varianza de los modelos que hemos construido, mediante la validación cruzada repetida. Tenemos el resumen de los modelos y sus resultados en un gráfico de cajas (ver figura 21).

Figura 21: Validación cruzada repetida (4 grupos y 10 iteraciones) en árboles



Fuente: Elaboración propia.

El modelo con un menor MAE es aquel con un valor de “minbucket” igual a 36. Aunque la varianza de este modelo parece superior al árbol con “minbucket” igual a 199 y es el único modelo con errores “outlier”. En términos de variabilidad, el mejor modelo sería el árbol con menor “minbucket”, pero queda desestimado por su alto sesgo. Debemos mejorar la variabilidad que presentan los árboles con “minbucket” =70 y 199. En términos de sesgo, el árbol con “minbucket” = 36 es la mejor opción.

Finalmente, tomamos como sujeto a optimizar los tres primeros modelos planteados. Los tres modelos han pasado por poda y/o “early stopping”, lo cual prueba la importancia de las técnicas de control frente estos algoritmos que tienden al sobreajuste. En los posteriores modelos de Bagging y Random Forest se tomarán como base de mejora estos resultados.

6.4.2.3. Evaluación en test y 2023

Como segundo método de validación observaremos el error de nuestros modelos en test. Exponemos el resultado en test de los modelos planteados mediante el MAE en la tabla 5 pero evaluaremos gráficamente un único modelo. El ejemplo seleccionado, será siempre aquel con un mejor comportamiento en test y se estudiará su comportamiento en las elecciones de 2023.

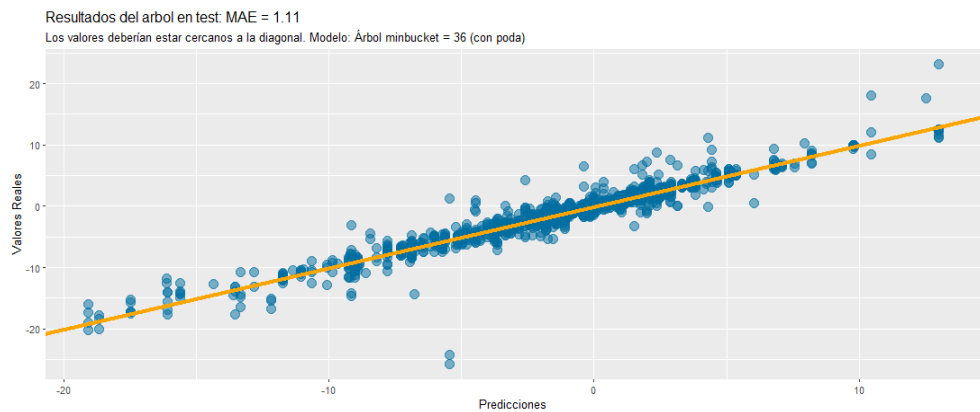
Tabla 5: Modelos de árbol y resultados en test

Model name	Minbucket	MAE
arbol_1	70	1.48
arbol_2	199 (poda)	1.95
arbol_3	36 (poda)	1.11
arbol_4	1020	2.47
arbol_5	150	1.86

Fuente: Elaboración propia.

Vemos que los resultados en test son estables, sobre lo que ya hemos visto en la comparativa mediante validación cruzada repetida. Ilustramos en la figura 22 los resultados de nuestro mejor modelo (arbol_3, “minbucket” = 36).

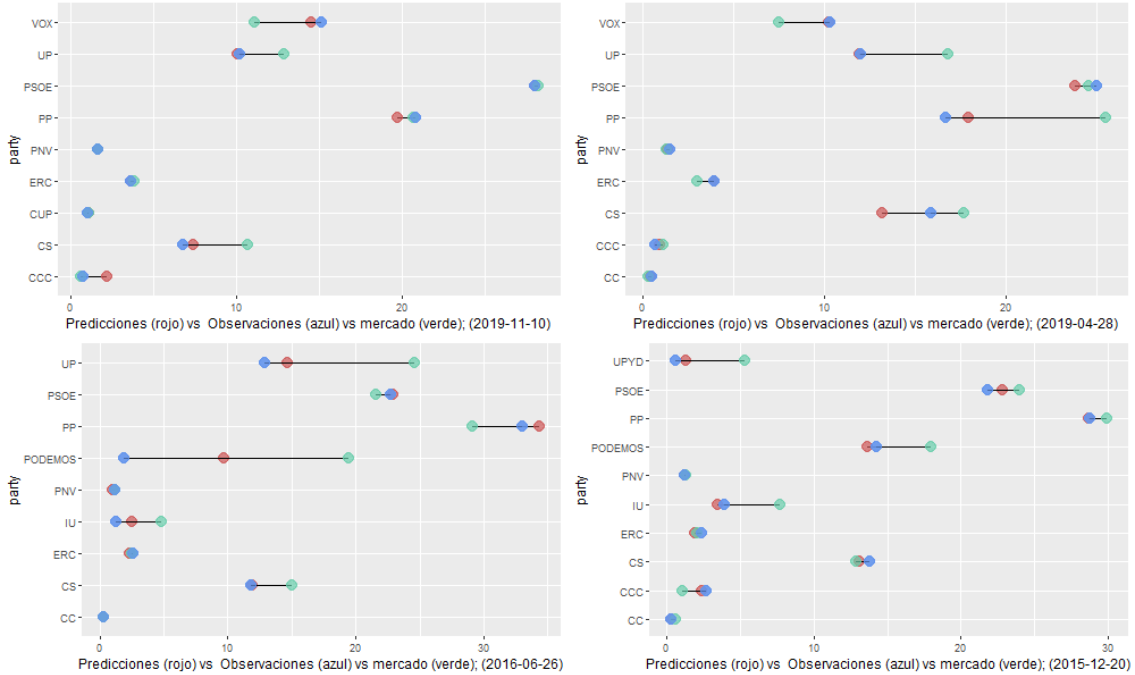
Figura 22: Errores estimados en test (arbol_3)



Fuente: Elaboración propia.

Usamos primero el gráfico de dispersión para ilustrar la distancia entre los valores de error observados (nuestra variable objetivo es el error de la encuesta) y el error predicho (sesgo de la encuesta según el modelo). La línea naranja, representa el ajuste perfecto. A más se acerquen los puntos a esa recta, mejor es el ajuste del modelo. En este caso vemos que el modelo no es muy bueno y tenemos muchos errores “outliers” (puntos que no siguen la tendencia), por lo que tenemos un modelo sesgado y con mucha variabilidad. En términos de estimación de voto, los resultados en test serían los siguientes (ver figura 23) para las últimas 4 carreras electorales:

Figura 23: Predicción en test del % de voto por partido (arbol_3), últimas 4 carreras

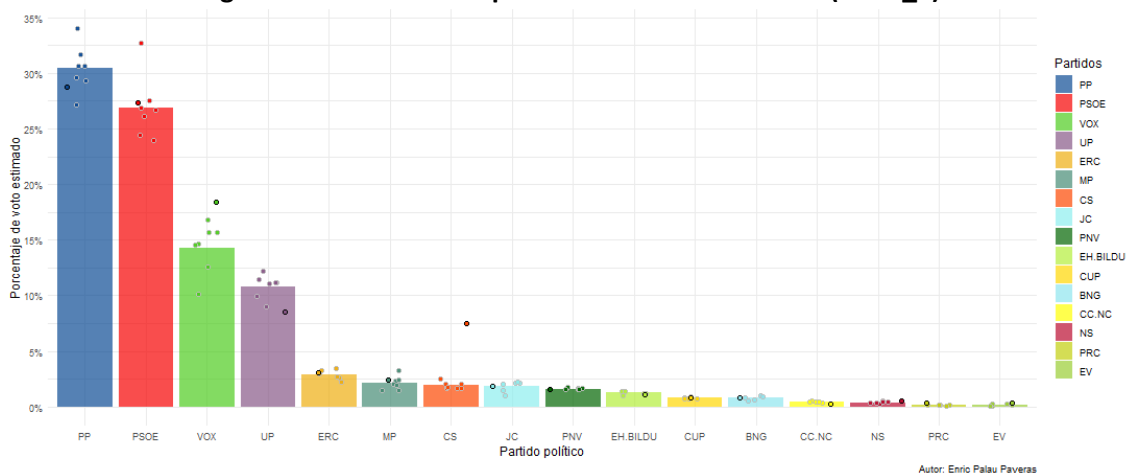


Fuente: Elaboración propia.

En el anterior gráfico Dumbbell, comparamos, la estimación de nuestro modelo (punto rojo) con la estimación sugerida por el promedio de mercado (punto verde), mediante la distancia (error) que tengan con el valor obtenido (punto azul) por el partido en la carrera. Los casos en los que no hay distancia o un solo punto, es porque el error fue ínfimo. El MAE de las encuestas evaluadas, es del 2.23, mientras que nuestro MAE es del 1.3. Es cierto, que no hemos incurrido a ninguna selección específica para esta evaluación. Pero con esto ya vemos que, para mejorar el promedio de mercado, no se necesita mucho más que una recogida de encuestas elaborada, datos de contexto y un modelo sencillo. Hay cierta mejoría respecto al promedio de encuestas, pero tampoco aseguramos que en todos los casos las superemos. Se aprecia claramente los casos más difíciles de predecir para ambos nuestro modelo y las encuestas: la formación de nuevos partidos, pues cuando apareció “Unidas Podemos”, dejando “Podemos” fuera de la candidatura, la predicción de ambos partidos fue relativamente mala; Y generalmente se predicen mal los partidos pequeños y de centro incluyendo Ciudadanos.

Finalmente, estudiamos los resultados de nuestro modelo para 2023 (figura 24). En los anexos (punto E), tenemos la tabla resumen. Cabe destacar que la muestra de encuestas para este conjunto está condicionada al momento de extracción de datos. Actualmente, es imposible obtener encuestas a menos de 300 días de antelación de las elecciones.

Figura 24: Estimaciones para las elecciones de 2023 (arbol_3)



Fuente: Elaboración propia.

En el anterior gráfico, las barras equivalen al promedio de mercado. El promedio de mercado en este caso es calculado con las encuestas con un tamaño muestral mayor a 30 y en un rango menor a 365 días previos a las elecciones. Los puntos con borde blanco, equivalen a las encuestas. Y los puntos con el borde negro, son la estimación de nuestro modelo. Según nuestro modelo de árbol ganador, en 2023, el partido con una mayor estimación de voto será el PP con un 28,8% , seguido del PSOE con un 27,3%. Estos están seguidos de los partidos de extrema VOX (18,4%) y UP (8,55%). Queda claro, que el modelo distingue bien el tamaño del partido (entre otros factores). Al plantear un problema de regresión, deberíamos interpretar los resultados considerando siempre el error del modelo. Es más, los resultados entre algoritmos pueden ser muy dispares, por lo que estos modelos no deben tomarse como una predicción definitiva, sino que más bien nos sirven de guía para interpretar con razón propia un posible futuro electoral.

6.4.3. Bagging de Árboles y Random Forest

A fin de superar las limitaciones de los árboles, usaremos modelos con el mismo concepto base y con el método de Ensamblado (combinando múltiples modelos en uno nuevo). Nuestro primer algoritmo de ensamble será el Bagging de árboles y después trataremos el Random Forest.

El Bagging se basa en la siguiente idea: sobre la tabla de tamaño “N”, se seleccionan “N” o “n<N” observaciones con reemplazamiento (o sin) de los datos originales y se aplica un árbol, del cual se obtienen las predicciones para todas las observaciones originales N (Portela, 2022). Este proceso se repite las m veces que estimemos para promediar las predicciones resultantes. Con las submuestras deberíamos poder reducir la dependencia de los datos iniciales para la construcción del modelo, optimizando así la varianza del modelo y bajando el sobreajuste y sesgo. Es decir, esperamos que este modelo supere a los árboles si usamos parámetros similares.

Por otro lado, los modelos Random Forest, se pueden considerar una modificación del Bagging que consiste en incorporar la aleatoriedad en las variables utilizadas para segmentar cada nodo del árbol. El proceso es prácticamente igual, pero a diferencia del que se hace en Bagging, cada vez que se abre un nodo seleccionaremos p variables de las k originales y de esas p elegidas se escoge la mejor para llevar a cabo la partición en ese nodo. Luego, se obtienen las predicciones del mismo modo que en Bagging.

6.4.3.1. Parámetros y rangos estudiados: Out Of Bag

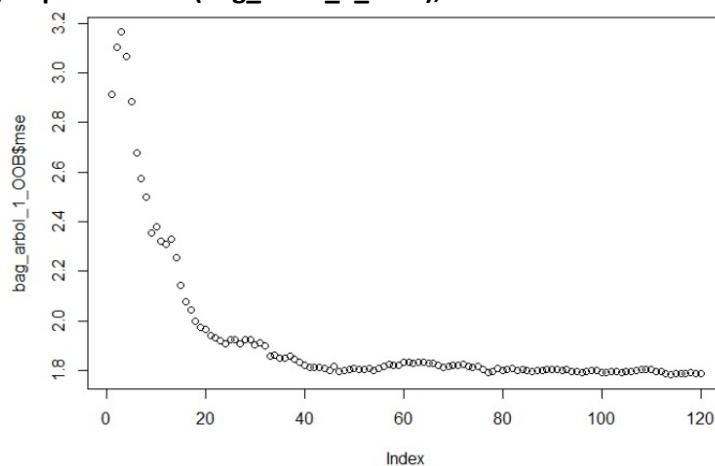
A continuación ilustramos los diferentes parámetros del bagging de árboles.

- **n tree**: siendo este el número de iteraciones “m” o árboles a promediar. Haremos uso del “Out Of Bag” para estimar este parámetro tras definir el resto de parámetros en los diferentes modelos Bagging planteados.
- **m try**: equivale al número de variables predictoras del modelo y en nuestro caso sería mtry=129. Posteriormente, pasaremos al estudio de este parámetro con los modelos de Random Forest.
- **nodesize**: tamaño máximo de los nodos finales, siendo éste un parámetro de complejidad tal como el “minbucket” en árboles. Para la determinación del “nodesize” nos basaremos en los anteriores árboles (“nodesize”= 36 y 70) y nuevas funciones.
- **sampsiz e**: tamaño de la muestra n vs N para la que realizaremos los árboles. Si este parámetro no es utilizado, pero hay un replace=TRUE el sampsiz e=7348 (todas las observaciones). Si especificamos este parámetro, debemos tomar un “n” igual o menor a $(1-(1/k))*N$, siendo k el número de grupos (que siempre dejamos en 4). En nuestro caso, el “sampsiz e” máximo es igual a 5.511.

- **replace:** Si se utiliza o no reemplazo (aunque por ahora, dada la complejidad de los datos pensamos que mantenerlo es la mejor opción). Siempre fijaremos `replace=TRUE`.

Empezamos con la búsqueda de parámetros óptimos, usando un bucle con un primer entrenamiento de validación cruzada simple. Consideramos que `n` tiene que estar entre 300 y 1.200. El máximo (5.500) es bastante inferior al “`sampsize`” máximo, definido anteriormente $(1-(1/k))*N=5.510$ y el “`sampsize`” mínimo viene definido por una cuestión de optimización en términos de coste computacional. Del mismo modo, los saltos de “`sampsize`” se dan de 100 en 100. Por otra parte, el “`nodesize`” se establece entre el 1% y el 10% de nuestras observaciones, escogiendo sólo los valores de 10 en 10. Con los resultados del bucle, pasaremos por OOB las iteraciones (`ntree`) y el número de variables para segmentar (`mtry` sólo para random forest). A continuación, ilustramos un ejemplo de gráfico de la exploración de parámetros en la figura 25.

Figura 25: Ejemplo de OOB (`bag_arbol_1_OOB`), validación cruzada simple (4 grupos)



Fuente: Elaboración propia, 2022

El OOB nos proporciona el error cometido en las observaciones que no caen en la muestra, y por tanto puede ser tomado como “observaciones test” dentro del entrenamiento con el MSE, como error a medida que avanzan las iteraciones (parámetro `ntree`). Regularemos también el `mtry` con esta técnica, para construir los modelos de Random Forest. Recordamos que el número de árboles no es crítico para el sobreajuste, puesto que a partir de los 120 árboles (en este ejemplo) se estabiliza la reducción de test error. Podemos ver todos los modelos de Bagging y Random Forest) resultantes en la próxima sección (Tabla 6).

6.4.3.2. Evaluación y comparativa de modelos entrenados

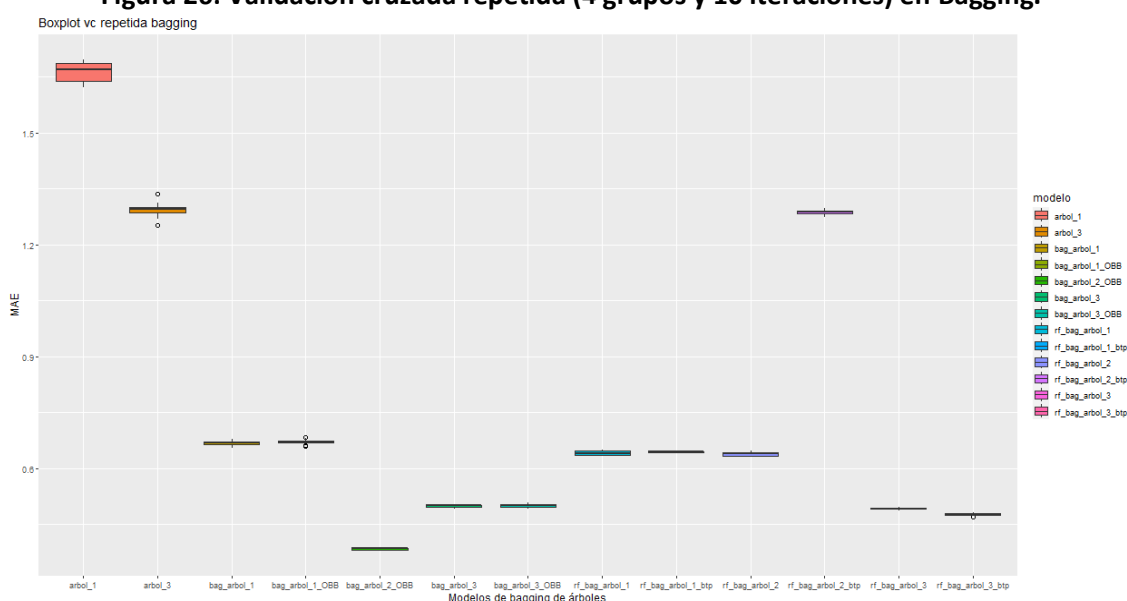
Al probar en validación cruzada repetida con los modelos que consideramos oportunos tras la exploración de parámetros, vimos que se genera una mejora continua entre árboles Bagging y Random Forest. Ilustramos el sesgo y varianza de los modelos a través del siguiente boxplot de la figura 26. También ilustramos los anteriores modelos de árbol para comparar la mejora entre algoritmos.

Tabla 6: Bagging y Random Forest, parámetros y resultados en validación cruzada repetida

Model name	ntree	nodesize	mtry	RMSE	MAE	R^2
bag_arbol_1	120	70	129	1.357	0.66	0.905
bag_arbol_1_OOB	77	70	129	1.363	0.671	0.904
rf_bag_arbol_1	120	70	126	1.353	0.664	0.906
rf_bag_arbol_1_btp	77	70	90	1.269	0.642	0.922
bag_arbol_2_OOB	230	23	129	0.901	0.383	0.957
rf_bag_arbol_2	230	23	77	1.27	0.637	0.924
rf_bag_arbol_2_btp	300	3	77	2.3	1.287	0.772
bag_arbol_3	600	40	129	1.072	0.498	0.941
bag_arbol_3_OOB	190	40	129	1.072	0.499	0.941
rf_bag_arbol_3	600	40	83	1	0.491	0.951
rf_bag_arbol_3_btp	190	40	78	1.07	0.494	0.940

Fuente: Elaboración propia.

Figura 26: Validación cruzada repetida (4 grupos y 10 iteraciones) en Bagging.



Fuente: Elaboración propia.

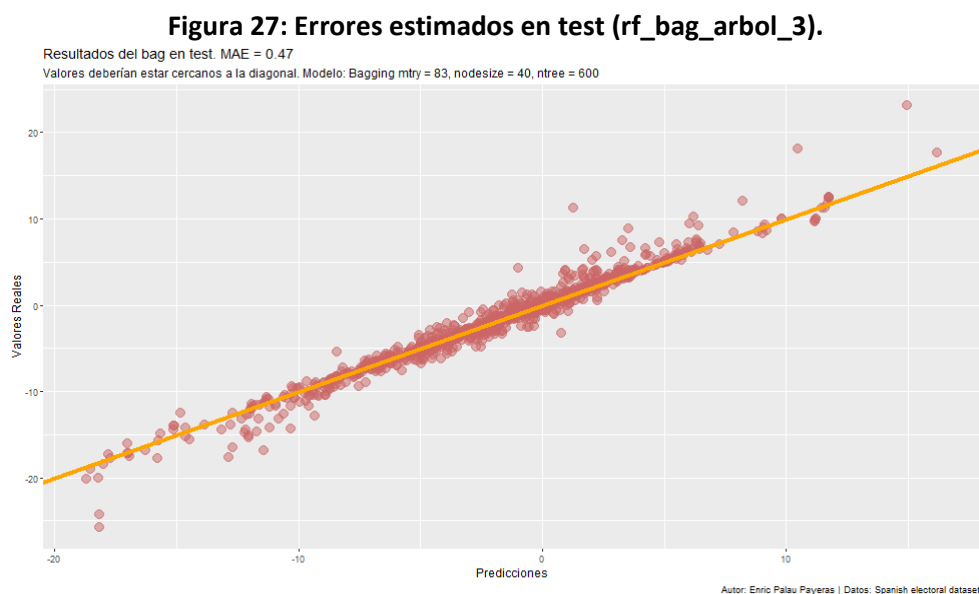
Como hemos adelantamos al explicar el concepto del modelo de Bagging, se ha mejorado tanto en sesgo como varianza el performance de los árboles iniciales. Considerando que arbol_1 y arbol_3 un tamaño máximo de los nodos finales muy similar al de los siguientes modelos de bagging, podemos hacer una comparativa justa de la mejora entre ellos. En el caso del arbol_3, vemos como de una variabilidad explicada de apenas el 60%, pasamos a explicar más del 94% con Bagging y rozamos el 95% con Random Forest.

En términos de MAE sucede lo mismo, el arbol_3 tenía un MAE de 1.1 y provocamos una decaída a un MAE de 0.5 (circa) en Bagging y 0.41 en Random Forest. Similar a la filosofía del promedio de encuestas para reducir la varianza, aquí en un enfoque predictivo lo que hacemos es obtener múltiples muestras de la población (o el símil al uso de Bootstrapping), ajustar un modelo distinto con cada una de ellas, y hacer la media de las

estimaciones. Por eso, con este proceso reducimos tan considerablemente sesgo y varianza. Por otra parte, la mejora entre Bagging y Random Forest no es tan brusca y es algo que podríamos esperar en el problema planteado. La relación entre nuestras predictoras es muy compleja y, por lo tanto, el conjunto de árboles está poco correlacionados. La mejora de Random Forest se fundamenta en una selección aleatoria de “m” predictores antes de evaluar cada división, dejándonos árboles divergentes. En nuestros caso, hay margen de mejora en este ámbito, pero afecta más en términos de variabilidad que sesgo.

6.4.3.3. Evaluación en test y 2023

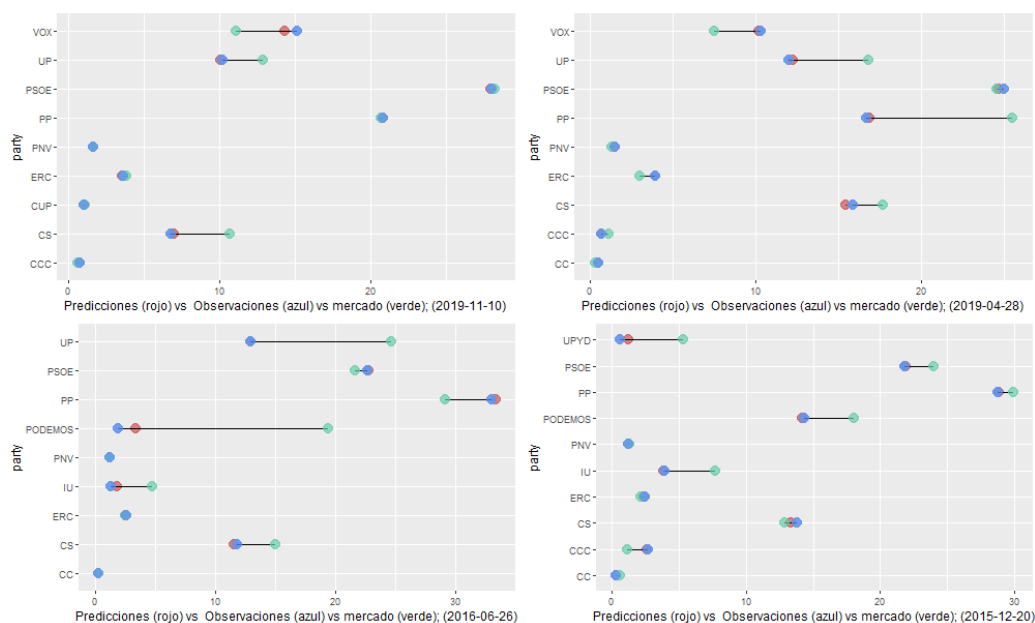
Por otro lado, vemos como nuestro modelo ha conseguido mejorar con creces la predicción del error cometido en las encuestas en test respecto al anterior modelo de árbol. Ilustramos en la figura 27 los resultados en un gráfico de dispersión. Vemos que, en este caso, el modelo tiene dificultades para predecir errores muy extremos. Hasta que no superamos los errores en un 10% el modelo ajusta casi a la perfección. A partir de ese rango (+/- 10%) los errores se hacen más difíciles de predecir. Es más, el modelo hace más extremos los errores “outlier”. Vemos como se traduce esto en términos de porcentaje de voto (ver figura 28). Por lo pronto, ya podemos concluir que el modelo de Random Forest ha mejorado con creces sobre el anterior modelo de árbol, pasando de un error MAE del 1,11 al 0,47. También ilustramos en un gráfico de Dumbbell (ver figura 28) la predicción a nivel de partido.



Fuente: Elaboración propia.

En términos de estimación de voto, nuestro modelo muestra valores mucho más cercanos al “outcome” real de las cuatro carreras seleccionadas. Es más, las dificultades para predecir factores como el de Podemos en 2016, persisten (igual que en árboles). Por otra parte, vemos que se reduce contundentemente la distancia entre observaciones y nuestras estimaciones. En la mayoría de los casos, esta no supera ni un punto porcentual. Tiene sentido viendo que el MAE ha bajado a un 0.56 frente al MAE del 2.23 de las encuestadoras y el 1.3 de nuestro anterior modelo de árbol.

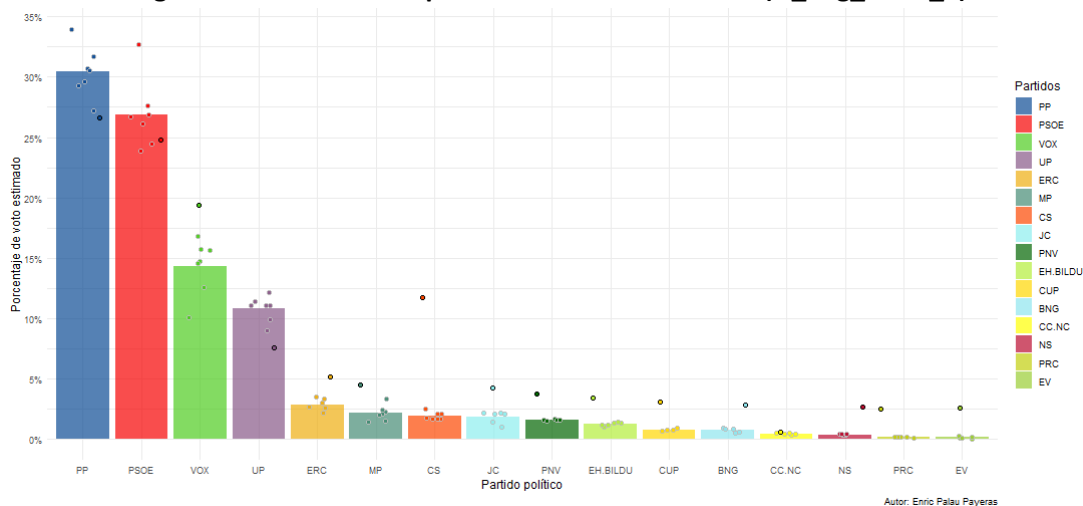
Figura 28: Predicción en test del % de voto por partido (rf_bag_arbol_3), últimas 4 carreras.



Fuente: Elaboración propia.

Finalmente, estudiamos el comportamiento del modelo para las elecciones de 2023. Según nuestro modelo de Random Forest ganador, en 2023 el partido con una mayor estimación de voto será el PP con un 26,64% de votos, seguido del PSOE con un 24,78% de votos. Estos partidos van seguidos de VOX en tercer lugar, acogiendo ya un 19%. Es el segundo modelo que nos ofrece estimaciones muy optimistas para este partido, si consideramos que el voto récord de VOX ha sido del 15,1%. También nos sorprende lo mucho que se desvía la estimación de nuestro modelo para CS, con respecto a las encuestas. Unidas Podemos decae en un 8% lo cual se alinea con el anterior modelo. Por lo general, nuestro modelo tiene un house_effect bastante positivo, para los partidos pequeños mientras que es más restrictivo con los principales partidos. Los resultados estimados entre los principales partidos se muestran en la siguiente figura 29.

Figura 29: Estimaciones para las elecciones de 2023 (rf_bag_arbol_3)



Fuente: Elaboración propia.

6.4.4. Gradient Boosting

Otro método alternativo al Bagging, es el Boosting. Los modelos Gradient Boosting están formados por un conjunto de árboles de decisión, entrenados de forma secuencial, de forma que cada nuevo árbol trata de mejorar los errores de los árboles anteriores. La predicción de una nueva observación se obtiene agregando las predicciones de todos los árboles individuales que forman el modelo. Y se diferencia de las dos versiones anteriores por ir modificando las predicciones en la dirección de decrecimiento dada por el negativo del gradiente de la función de error.

Al final hacemos una construcción de árboles reiterativa en la que vamos modificando ligeramente las predicciones iniciales, buscando el minimizado de los residuos en un parámetro “V”. Al plantear muchos árboles consecutivamente, conseguimos un ajuste progresivo y que los árboles se corrijan entre ellos. En este caso, dado que si no establecemos parada los datos tienden a ajustar al 100% el dato, es primordial usar “Early Stopping” para evitar el sobreajuste y marcar así un número de iteraciones de parada.

6.4.4.1. Parámetros y rango estudiado:

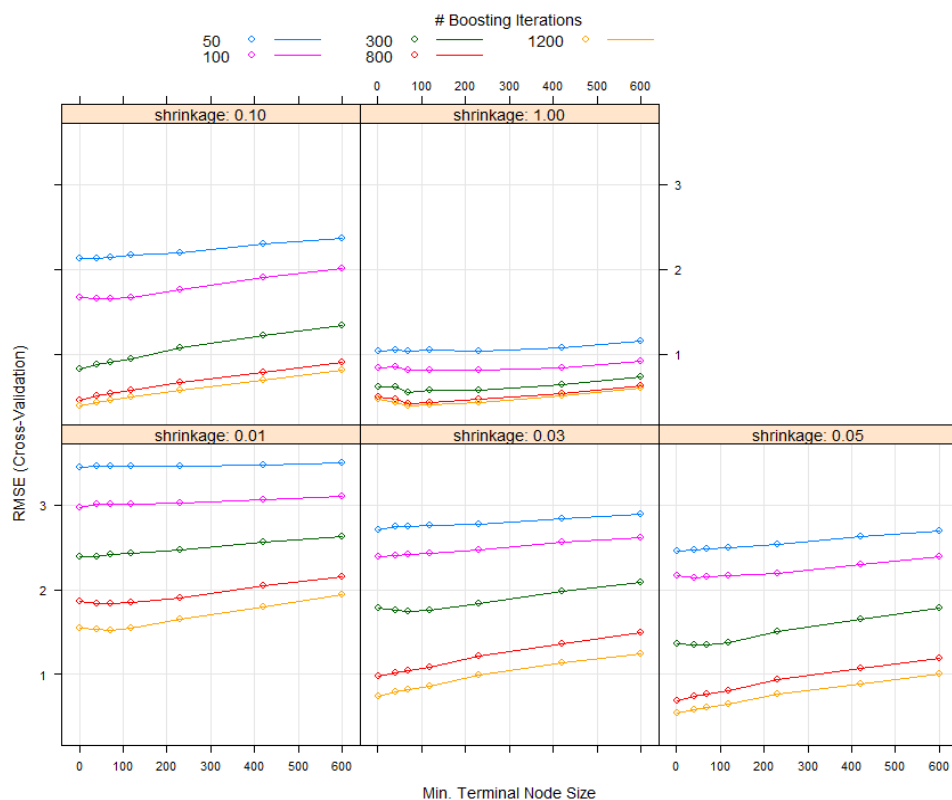
El método gbm (en caret) tiene los siguientes hiperparámetros básicos:

- **n.trees (“B”)**: igual que con Bagging y Random Forest representa el número de iteraciones (árboles). A mayor n.trees fijemos mayor sobreajuste. Lo limitaremos según el “learning rate” o “shrinkage”.
- **interaction.depth (“d”)**: complejidad de los árboles (“weak learner”) o número de divisiones. Siempre lo fijaremos en 2.
- **Shrinkage (“λ”)**: o “learning rate”, controla la influencia de los modelos sobre el conjunto del ensemble. Representa el ritmo de aprendizaje y cuanto menor sea el “shrinkage”, mayor “n.tree” se necesita (buscaremos el mínimo posible). Nuestro “shrinkage” varía desde 0.001 (valor por defecto) hasta 1. De manera que podamos ver si tiende a los parámetros más bipolares o si se ubica en términos medios evitando así un sobreajuste entre los “n.trees” y este parámetro.
- **n.minobsinnode**: como el “nodesize” de Random Forest y Bagging o el “minbucket” en árboles, equivale al número mínimo de observaciones para dividir los nodos. Nosotros exploraremos este parámetro según los resultados previos y el uso de bucles.
- **distribution**: En nuestro caso siempre asignaremos esta función de coste como “gaussian” pues es una de las únicas opciones para problemas de regresión.

- **bag.fraction (subsampling fraction):** Reconocemos este parámetro como la fracción de observaciones del set de entrenamiento seleccionadas de forma aleatoria para ajustar cada “weak learner”. Al estudiar el Gradient Boosting, fijamos “bag.fraction”=1, si es inferior a 1 tratamos Stochastic Gradient Boosting.

Entonces, entendemos que este modelo depende de tres hiperparámetros esenciales (“B”, “d” y “λ”) más la complejidad de los árboles empaquetados. A diferencia de con Bagging y Random Forest, donde los árboles son independientes, debemos tener mucho cuidado con el número de árboles en Gradient Boosting, ya que es un modelo aditivo que se alimenta de los residuos (gradientes) entre ajustes de árboles. Entonces, es fácil caer en un sobreajuste. En nuestro caso, lo limitamos a un máximo de 1.200. Viendo que siempre obtenemos los mejores parámetros entre 1.200 y 800 ajustaremos los rangos en esos valores con un estudio más preciso. Por otro lado, con relación a las iteraciones seleccionaremos el “λ” siempre estudiaremos este valor por debajo de 1 con la intención de provocar un aprendizaje lento y evitar el sobreajuste. Para el tamaño del árbol y su complejidad partimos de que fijamos las divisiones en 2 y basaremos el rango de “n.minobsinnode” en valores relativamente cercanos para los ya explorados en nuestros anteriores modelos. Eso sí, siempre que sea factible intentaremos elegir, el menor “learning rate”, mayor “n.minobsinnode” y menor número de iteraciones posibles. Resumimos la anterior exploración en un gráfico resultante del bucle de parámetro en validación cruzada en la figura 30:

Figura 30: Estudio de parámetros para GBM, validación cruzada simple (4 grupos).



Fuente: Elaboración propia.

A continuación ilustramos los resultados de la exploración den un resumen numérico en la Tabla 7 y el gráfico en la Figura 31. Recordemos que alternamente al Bagging, tenemos métodos Boosting, por lo que también pretendemos superar los anteriores modelos de árbol planteados.

6.4.4.2. Evaluación y comparativa de modelos entrenados

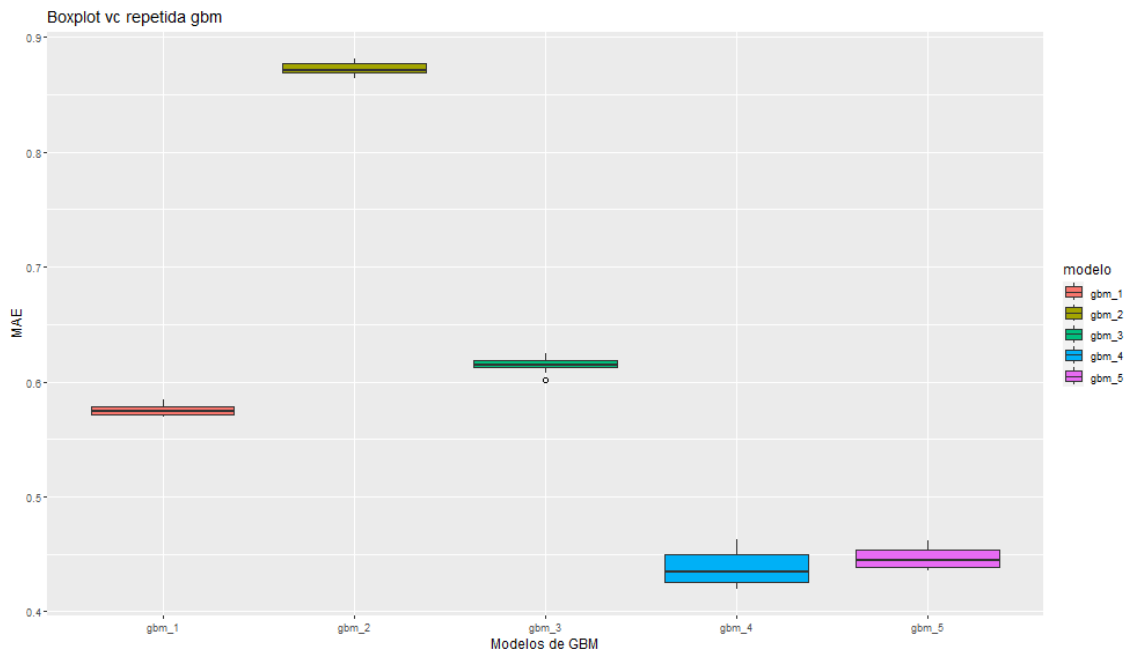
Tabla 7: GBM, parámetros y resultados en validación cruzada repetida.

Model name	shrinkage	n.minobsinnode	n.trees	RMSE	MAE	R^2
gbm_1	0.1	82	1000	0.931	0.575	0.954
gbm_2	0.05	70	800	1.396	0.872	0.903
gbm_3	0.1	70	800	0.989	0.614	0.948
gbm_4	0.8	40	800	0.666	0.437	0.975
gbm_5	0.5	150	800	0.688	0.446	0.973

Fuente: Elaboración propia.

Evitamos combinar un número de árboles grande con tasas de aprendizaje rápidas. Entre los modelos resultantes, el que menos sobreajuste debería generar es el modelo gbm_2, pues usamos árboles relativamente sencillos con un ritmo de aprendizaje lento y el mínimo de iteraciones dentro de nuestro margen óptimo (entre 800 y 1.200). A pesar de su buena parametrización, los resultados en términos de error son bastante peores que el del resto de modelos. Aún así, mejora mucho con respecto a los árboles individuales planteados. Los modelos gbm_1 y gbm_5, a pesar de tener un ritmo de aprendizaje algo alto, tienen árboles sencillos y un número de iteraciones equilibrado. Comparando su MAE con el resto de modelos serían ambos muy buenos ejemplos para estudiar su comportamiento en test. El modelo gbm_4 se podría considerar el mejor en términos de error, pero no lo consideramos óptimo pues se compone por árboles algo complejos y tiene una tasa de aprendizaje muy elevada, pudiendo llevar a un modelo sobreajustado. Resumimos este análisis en un diagrama de cajas en la siguiente figura:

Figura 31: Validación cruzada repetida (4 grupos y 10 iteraciones) en boosting.

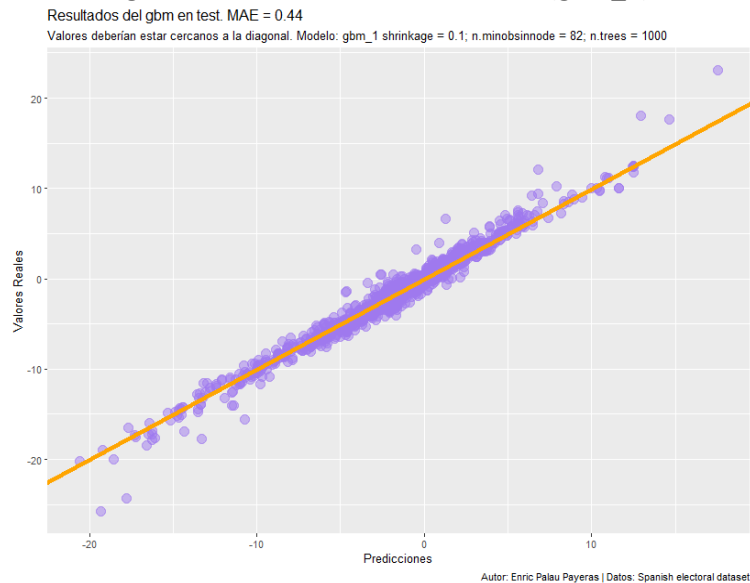


Fuente: Elaboración propia.

Vistos los niveles de error y varianza, consideramos que el gbm_1 y gbm_5 son buenos ejemplos para exponer en test. En este caso, al tener niveles de error tan bajos, optaremos por el modelo con menos variabilidad siendo este el gbm_1.

6.4.4.3. Evaluación en test y 2023

Figura 32: Errores estimados en test (gbm_1).



Fuente: Elaboración propia.

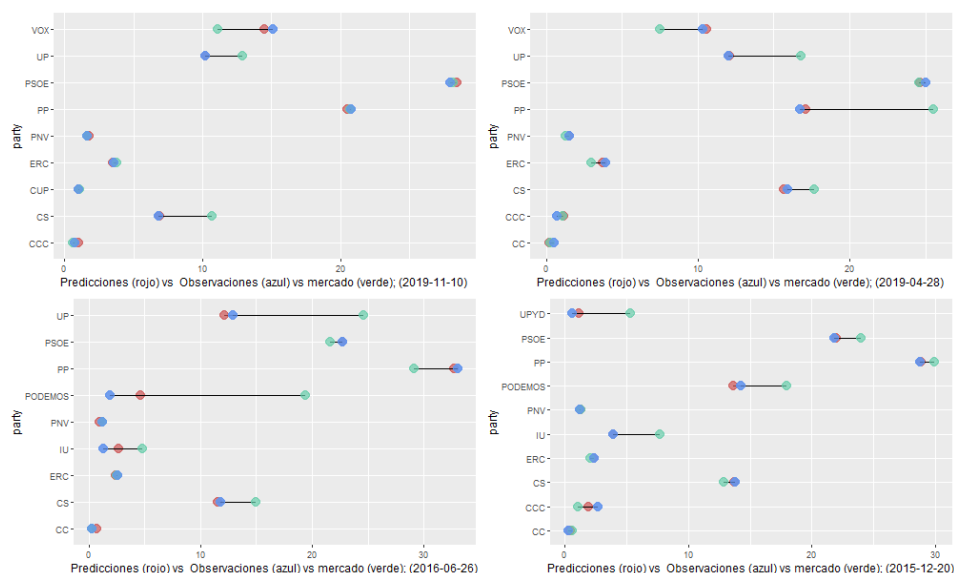
Pasamos a evaluar las predicciones en test del gbm_1. En primer lugar, hay una pequeña mejoría en término de error MAE. Hemos pasado de un árbol con un MAE del 1.11 a un Random Forest con un 0.47. En este caso, tenemos un MAE del 0.44. Por lo que quedaría contratada la siguiente hipótesis: “tendremos una mejora continua entre los modelos

de árboles, Bagging, Random Forest y Boosting”. Evaluamos observaciones de error contra estimaciones de error en un gráfico de dispersión en la figura 32. También evaluamos en un gráfico dumbbell en la figura 33, el cual muestra las estimaciones de voto ofrecidas por nuestro modelo.

Si evaluamos el modelo en términos de predicción del error cometido en las encuestas, vemos como hay un muy buen ajuste entre las predicciones en un intervalo de error absoluto inferior al 15%. Fuera del intervalo empiezan a proliferar unos pocos errores “outlier”. En este caso, vemos como se ha mejorado el problema con los errores extremos. A pesar de ello, queda un pequeño margen de mejora en términos de sesgo.

En estimación de voto los resultados en test de nuestro GBM, para las últimas 4 carreras son muy similares, a las que nos ofreció el anterior Random Forest. El error MAE de nuestro GBM en estimaciones de voto es del 0.6, algo peor que el previo Random Forest, pero muy por encima de las encuestas y los promedios. Debemos recordar que el MAE obtenido al predecir errores no es comparable al MAE en estimación de voto. Y por ello siempre evaluamos nuestros modelos en test con dos MAE distintos (uno para cada variable objetivo).

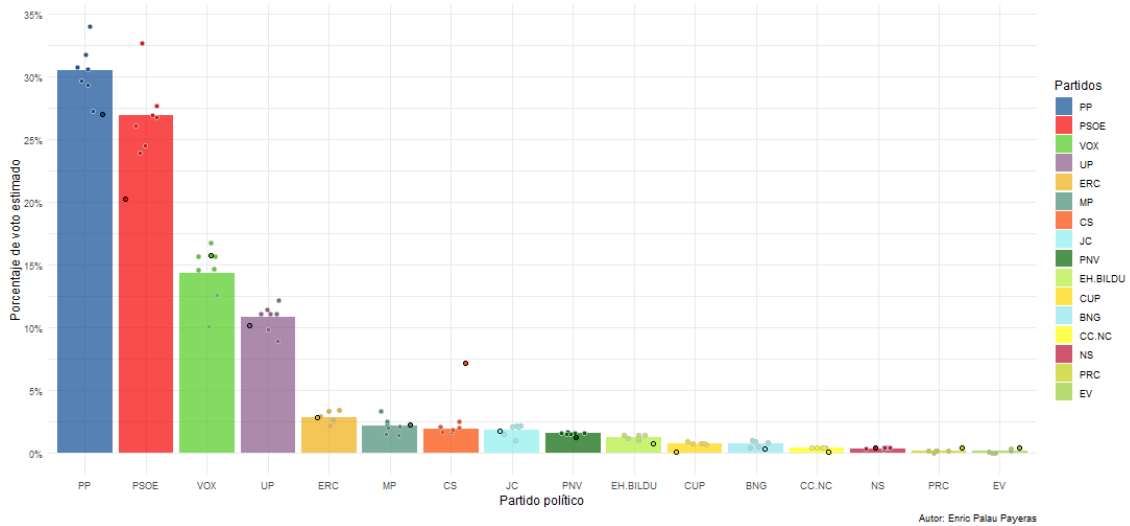
Figura 33: Predicción en test del % de voto por partido (gbm_1), últimas 4 carreras.



Fuente: Elaboración propia.

A continuación evaluamos el modelo en sus estimaciones para 2023. Según nuestro modelo Boosting ganador, el partido con una mayor estimación de voto será el PP con un 26,95%, seguido del PSOE con un 20,87%. Topamos ya un mínimo en la estimación de PSOE. Teniendo también a VOX con un 15% y UP con un 10,16%, vemos que todos los modelos han dado una clasificación similar, por lo que ya podemos empezar a denotar una lectura de consenso. CS vuelve a tener un house_effect positivo. Esto es un factor a estudiar, pues muchos modelos persisten a la hora de estimar votos directos superiores al 10% cuando para 2023 hay expectativas de disolución del partido. Otros, penalizan este partido con estimaciones inferiores al 1%. Generalmente, este es el modelo que más se alinea con las encuestas y el promedio de mercado.

Figura 34: Estimaciones para las elecciones de 2023 (gbm_1)

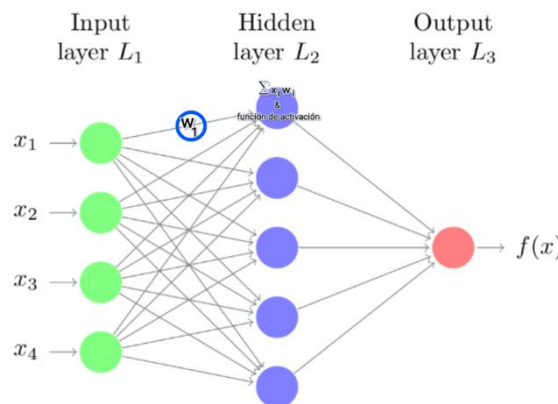


Fuente: Elaboración propia.

6.4.5. Redes Neuronales

Las redes neuronales, son algoritmos que parten de la idea del cerebro humano. Ilustramos en la figura 35 y describimos este concepto. La información aportada por las variables predictoras (“ x_i ”, x_1 , x_2 ,...) sería la entrada de información mediante los sentidos (vista, tacto,...) y la percepción que recibe el humano sería el output (“ $f(x)$ ”). Entonces, tenemos capas de neuronas conectadas entre ellas mediante pesos. Por lo que, como mínimo, tenemos una capa de entrada (puntos verdes o L_1), otra capa oculta o intermedia (los puntos azules, L_2 , identificados como “ H_i ” y se pueden aumentar) que resume linealmente la información asignando pesos (“ w_{ij} ”) y la capa de salida (punto rojo L_3).

Figura 35: Esquema básico del modelo de red (concepto)



Fuente: Computer Age Statistical Inference 2016.

Entonces, además de la capa de entrada, la capa oculta y la de salida vamos a tener otros parámetros. Uno de ellos son las conexiones o pesos entre capas de neuronas (el anterior “ w_{ij} ”). Otro parámetro sería la función de activación que aplicamos a los nodos ocultos para e la red pueda aprender relaciones complejas en los datos y no de forma lineal. Finalmente, tenemos una serie de variables explicativas a modo de input.

Además, es importante entender cómo funcionan las neuronas. Dentro de cada neurona con la información de entrada se generan sumas ponderadas (multiplicamos valor de entrada “ x_i ” por el peso “ w_i ” de la conexión) y para superar meras combinaciones lineales se configura la función de activación.

Antes de proceder con la parametrización, destacamos dos aspectos sobre las redes y los datos de entrada. Como ya adelantamos en la sección de “Modify”, hemos aplicado la binarización (“one-hot-encoding”) sobre nuestras variables categóricas, ya que las redes solo permiten inputs numéricos. También exclusivamente para este algoritmo, hemos recurrido al centrado, es decir, le hemos restado a todas las observaciones (en sus valores numéricos) la media de la variable en cuestión. De esta forma dejamos la media de los predictores numéricos en 0 y se centran valores en torno al origen.

6.4.5.1. Selección de variables

Retomamos otro apunte de la sección de ETL y de la de “Modify”. No todos los predictores recogidos, son útiles. Por ello en la fase de ETL y Explore hemos ido creando y descartando variables explicativas. Al ser el modelo de redes altamente sensible a los datos de entrada y no tener un proceso de filtrado interno, haremos un estudio comparativo entre sets de variables tentativos (como en la fase de Explore con el modelo de árbol). Para ello tenemos dos métodos de selección de variables: Filtros y Wrappers.

6.4.5.1.1. Filtros estudiados

Los modelos planteados hasta ahora (árboles, Bagging de árboles, Random Forest y GBM) filtran las variables con un ranking de importancia según la relación con la variable objetivo (“errores” de las encuestas). Ya hemos visto un ejemplo bastante ilustrativo en la sección de explore (ver figura 18). Esto se debe a que todos los modelos aplicados se basan en árboles. Los filtros generan un “ranking” de variables a fin de cuentas. Pero tenemos dos inconvenientes con estos filtros: no son relativos a otras predictoras, pudiendo elegir dos variables que signifiquen lo mismo, y no ofrecen un criterio de corte. Es decir, a partir de qué valor de importancia u orden debemos considerar desechar variables. Por defecto, consideramos siempre como punto de corte un aporte igual o inferior a 0. Reflejamos un resumen de los sets resultantes de estos métodos en la próxima sección.

6.4.5.1.2. Wrappers estudiados

A fin de complementar los inconvenientes de los filtros, también hemos recurrido a otra técnica de selección de variables: los “wrappers”. Estos son métodos de búsqueda secuencial y buscan entre distintas combinaciones de variables. Estos no llegan a evaluar todos los sets tentativos y pueden pasar por alto buenas combinaciones de variables. Por otra parte, hay relaciones matemáticas muy directas o derivadas del azar, también pueden tender a selecciones basadas en “sobreajuste” o casualidad. Utilizaremos SAS para la aplicación de estas técnicas.

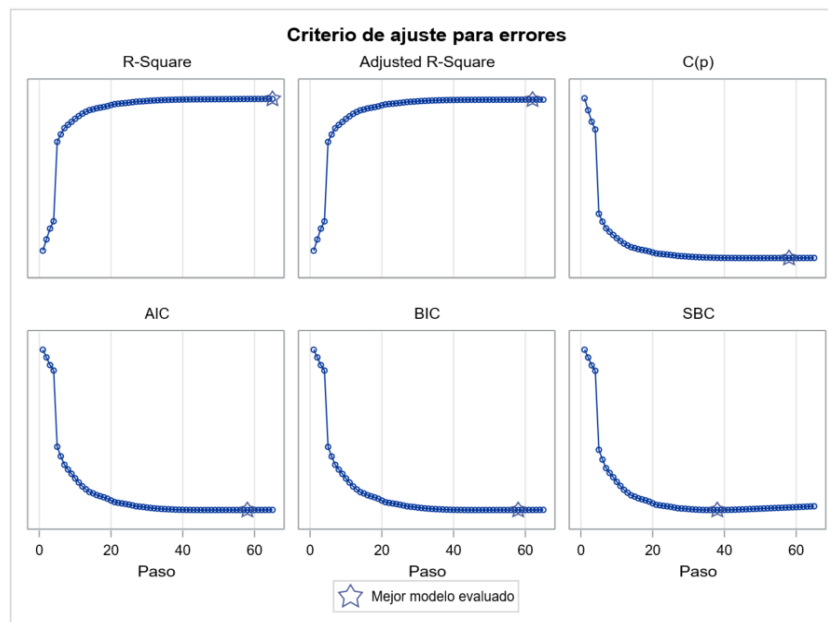
Usamos la regresión lineal múltiple para descubrir relaciones entre variables y ofrecer una combinación de predictores óptima a nuestro modelo. Realmente buscamos el modelo lineal que mejor prediga la variable dependiente a partir de subconjuntos de variables independientes. Esta inspiración se ha cogido de (Beal, 2007).

Estos modelos requieren de una dirección y un criterio de parada. Los criterios de parada en los que nos centraremos son AIC, BIC y SBC. El BIC (Bayesian Information Criterion) y el AIC (Akaike Information Criterion) se resumen en las siguientes funciones; $BIC = -2L(\text{modelo}) + \ln(n)k$; $AIC = -2L(\text{modelo}) + 2k$. Entendemos el parámetro “k” como unidad de parámetros a estimar (en el caso de la regresión lineal $k=p+2$, ya que debemos estimar $p+1$ coeficientes y la varianza residual), y $L(\text{modelo})$ la log-verosimilitud del modelo definida como $L(\text{modelo}) = \ln(P(\text{datos observados} | \text{modelo estimado}))$ teniendo como diferencia entre modelos la penalización (Liébana, 2021). Por otro lado, tenemos derivado de AIC el SBC, pero usa un multiplicador de $\ln(n)$ para k en lugar de una constante 2 al incorporar el tamaño de la muestra n (Beal, 2007). Todos nos ofrecen una medida que balancea la calidad del modelo frente al número de predictores empleados. Adicionalmente, SAS nos permite añadir diagnósticos por paso y generales como “análisis de error y variabilidad”, incluyendo estadísticos como el coeficiente de determinación ajustado (R^2) y la C_p de Mallow. Nosotros repetiremos el proceso y su análisis para tres métodos heurísticos comunes: selección directa, selección inversa y regresión paso a paso. Lo haremos en función de los tres criterios de parada comentados, según los estadísticos de diagnóstico.

Por lo general, el proceso consiste en los siguientes pasos: importamos los datos a SAS (en este caso hemos aplicado SAS Enterprise Guide, SAS Enterprise Miner y SAS base), corregimos las tipologías de las variables y asignamos sus funciones, configuramos el modelo de regresión y el output deseado (informe de estadísticos por selecciones de variables). Crearemos un paso final para exportar automáticamente el resultado en formato HTML a fin de poder conservar el informe del modelo completo (ver anexos para consultar el código SAS). Ilustramos los resultados del estudio por las direcciones aplicadas y los criterios de parada.

- **Stepforward:** Con este criterio de dirección lo que hacemos es estudiar los distintos criterios de parada a medida que se van añadiendo variables al “dataframe”. Partimos de 0 variables hasta llegar a un nivel de significación mínimo de 0.5 para la entrada en el modelo. El nivel de significación es ajustable pero por defecto SAS aplica el 0.5. Si vemos selecciones insuficientes o incoherentes bajaremos el nivel de significación. Una vez se añade una variable, esta se queda en el modelo. Ilustramos los resultados.

Figura 36: Criterios de parada (step_forward) con SAS

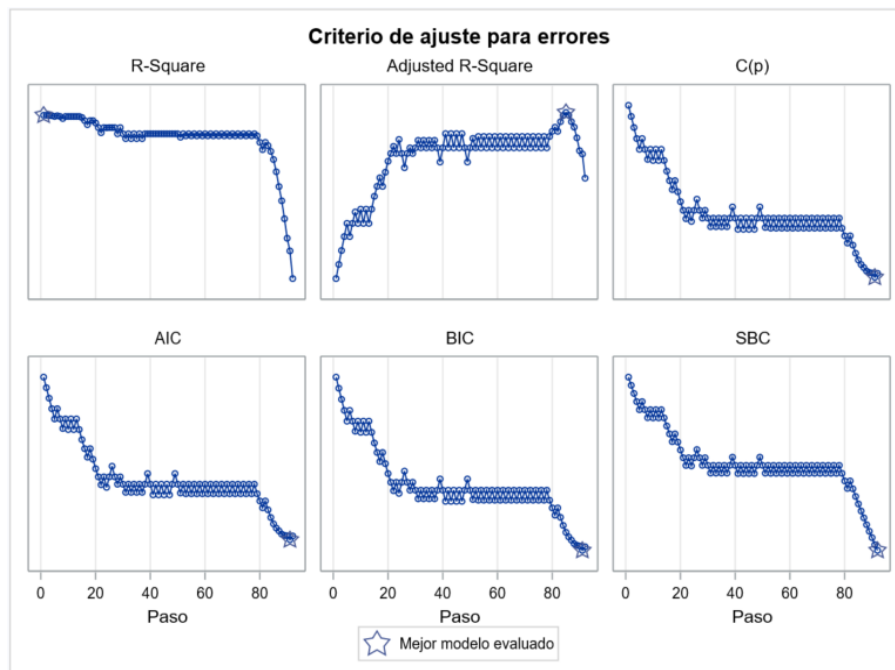


Fuente: Elaboración propia en SAS ('SASApp', Linux).

Según el criterio de información Akaike (AIC) el número óptimo de variables a incluir es 58 igual que con el criterio de información bayesiano de Sawa (BIC). El criterio de información de Schwarz (SBC) es más estricto limitando la selección de variables a 40. Las variables seleccionadas por estos tres métodos son: en primer lugar variables relacionadas con las encuestas como, los días para las elecciones, la estimación de voto, el efecto de la casam etc; luego tenemos los promedios de encuestas y una gran variedad de variables fundamentales relacionadas con el grado de corrupción del gobierno previo, el ingreso fiscal per cápita, la variación del PIB, etc.

- **Stepbackwards:** Con este criterio de dirección lo que hacemos es estudiar los distintos criterios de parada a medida que se van restringiendo variables del "dataframe". Partimos de 133 variables y sólo permanecen aquellas que superen el 0.1 de significación. A partir de allí vamos retirando variables por pasos hasta que según c llegamos a un nivel de significación mínimo del 0.5 para la entrada en el modelo. El nivel de significación es ajustable pero por defecto SAS aplica el 0.5. Si vemos selecciones insuficientes o incoherentes bajaremos el nivel de significación. Ilustramos los resultados en la figura 37.

Figura 37: Criterios de parada (step_backward) con SAS



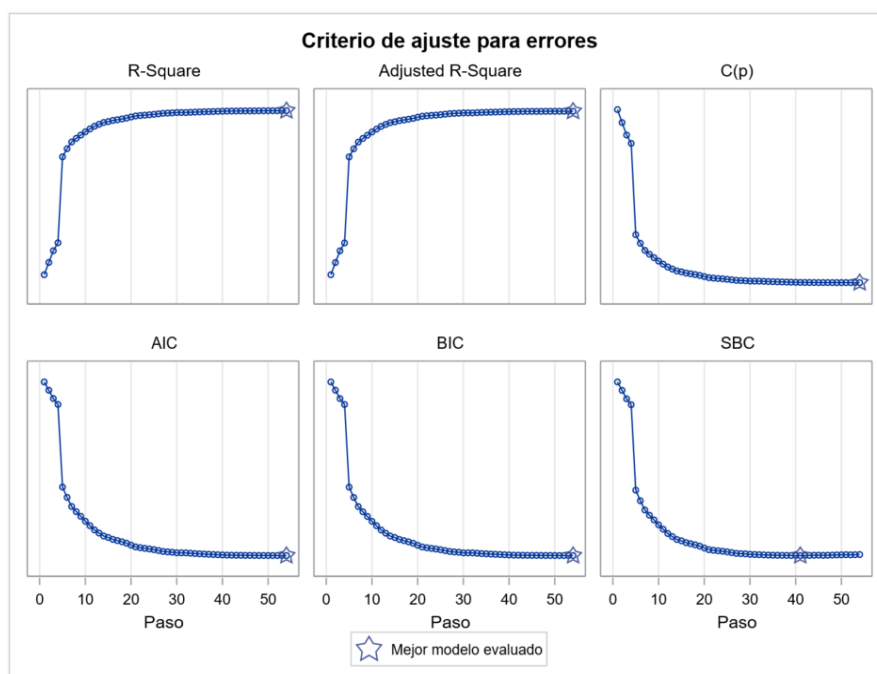
Fuente: Elaboración propia en SAS ('SASApp', Linux).

Según el criterio de información Akaike (AIC), el número óptimo de variables a incluir es 58 igual que con el criterio de información bayesiano de Sawa (BIC). El criterio de información de Schwarz (SBC) es más estricto limitando la selección de variables a 57. Entre las variables desestimadas: tenemos 27 de las 38 variables fundamentales, todas las variables de días para las elecciones (urnas y días de trabajo de campo) y todas las variables de gobierno.

- **Stepwise:** Con este criterio de dirección estamos agregando una variable por paso y evaluando los valores p de los estadísticos F de las variables ya incluidas para ver si exceden el nivel de significación configurado (0.15). Si es así, elimina esa variable y agrega otra al modelo. Solo después de realizar esta verificación y eliminar las variables identificadas se agrega otra variable. Ilustramos los resultados en la próxima figura.

En la figura 38 vemos que, según el criterio de información Akaike (AIC) el número óptimo de variables a incluir es 45 y 46 según el criterio de información bayesiano de Sawa (BIC). El criterio de información de Schwarz (SBC) es más estricto limitando la selección de variables a 37 variables. De nuevo entre las variables seleccionadas predominan los promedios de encuestas, variables sobre las propias encuestas y por último algunas variables fundamentales, concretamente dos por cada bloque temático.

Figura 38: Criterios de parada (step_wise) con SAS



Fuente: Elaboración propia en SAS ('SASApp', Linux).

6.4.5.1.3. Elección y estudio de los sets resultantes

Hemos acogido las selecciones de los modelos de árboles, Bagging, Random Forest y GBM, desarrollados en las anteriores secciones. También hemos aunado las selecciones de variables obtenidas con filtros elaborados en SAS. Entre todos los métodos tenemos que los 23 modelos han seleccionado `est_surv_vote`, `party`, `days_to_elec`, `prom_casa_partido`, `prom_carrera_wing`, `prom_carrera_partido`, `prom_general_partido`. Esas 7 variables pertenecen a las grupos de datos de encuestas y promedios, por lo que podemos concluir que, tal como se ha indicado en el estado del arte, la predicción de resultados electorales mediante encuestas es mucha más efectiva que el uso exclusivo de variables de contexto.

Sobre las variables fundamentales, hasta 21 modelos han llegado a seleccionar variables como: la tasa de desempleo, la esperanza de vida, corrupción del gobierno, la formación del gobierno, el ingreso fiscal per cápita y la población. En definitiva, el uso de promedios aventaja mucho los algoritmos y la información aportada. Adicionalmente, usar variables de contexto puede ser arriesgado en términos de ruido, pero parece ser que si son bien seleccionadas pueden tener un aporte muy consistente. Resumimos la selección de variables entre los diferentes modelos en la tabla 8. Igual que en el esquema de estructura del estudio presentada al principio del trabajo en la figura 1, resaltamos en naranja los datos de contexto, en amarillo los datos de encuestas, en azul los datos de carrera y en verde los promedios y otros estadísticos.

Para los próximos modelos y a modo comparativo, usaremos dos sets de variables. El segundo set se va a componer de las variables seleccionadas (considerando las variables “dummy”) por más del 50% de los modelos planteados. Hemos usado 23 modelos, por

lo que consideramos hasta 12 modelos como punto de corte. El segundo conjunto se basará en una de las selecciones de variables resultante de nuestros filtros. Para comparar los modelos de manera justa recurriremos al uso de la validación cruzada repetida. Entre los modelos ilustrados en la próxima figura, sólo destacaremos aquellos filtros con mejores resultados.

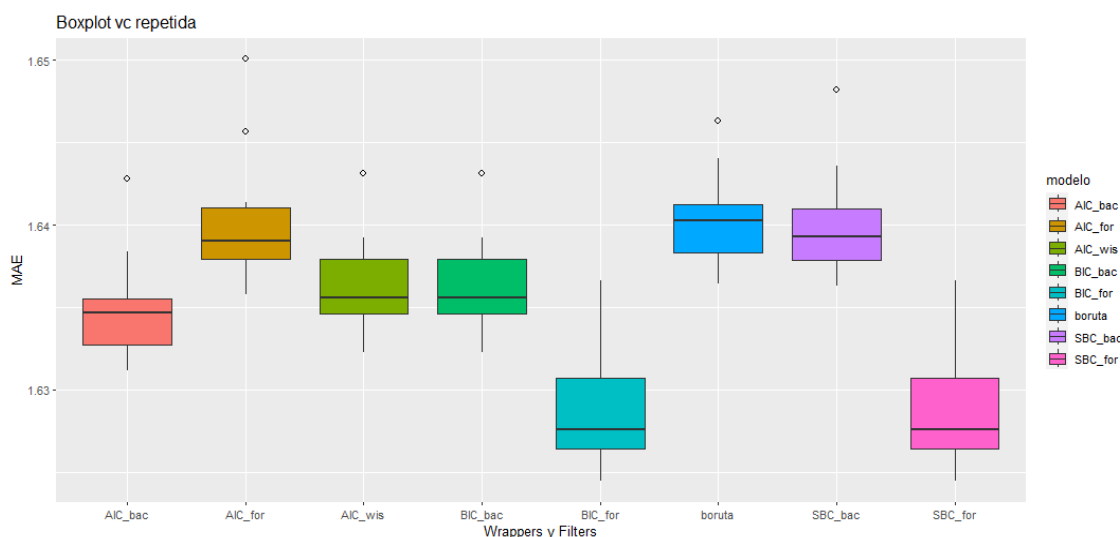
Tabla 8: Selección de variables por bloques.

> 50% de modelos		<= 50% de modelos	
est_surv_vote	23	env_co2	12
party	23	pobl_em_rate	12
days_to_elec	23	env_kwh_consum_percap	12
prom_casa_partido	23	prom_carrera_casa_wing	11
prom_carrera_wing	23	eco_rate_avg	11
prom_carrera_partido	23	pobl_fem_porc	11
prom_general_partido	23	gov_exp_edu_percap	11
pobl_pobreza_rate	21	wing	11
house_effect_e	20	pobl_suicide_percienmil	10
prom_general_wing	20	pobl_kill_percienmil	10
pobl_suicide	20	gov_exp_war	10
poll_firm	20	n	9
eco_unemployment	18	wing_effect_e	9
pobl_life_expectancy	18	eco_pib_percap	9
gov_cor_rate	18	eco_smi	9
gov_pre	18	env_co2_percap	9
eco_fisc_ing_percap	18	gov_exp_pib	9
pobl	18	gov_exp_san	9
lead2_paty	18	gov_exp_war_percap	9
urna_15	18	gov_exp_edu	9
lead_party	18	eco_fisc_ing	8
porc_surveys_firm	16	pobl_idh	8
pobl_im_rate	16	env_gwh_consum	8
prom_casa_wing	15	env_gwh_prod	8
eco_pib_var	15	env_gwh_prod_renovable	8
eco_deficit	15	urna_365	7
pobl_densidad	15	eco_debt_percap	7
pobl_kill	15	urna_7	6
year_elec	14	urna_60	5
prom_carrera_casa_partido	14		

Fuente: Elaboración propia.

El primero, equivale a la comparativa en “boxplot” siguiente. Entre los mejores sets, tenemos el de BIC_for y SBC_for, ambos muy similares. Hemos decidido usar todas las variables de ambos sets teniendo un conjunto total de 40 variables (considerando las variables “dummy”). El segundo modelo o set de variables corresponde a las variables de la columna izquierda del anterior gráfico.

Figura 39: Validación cruzada repetida (4 grupos y 10 iteraciones) en regresión lineal



Fuente: Elaboración propia.

6.4.5.2. Función de Activación y capas ocultas

Tanto la función de activación como el número de capas ocultas serán explorados de forma muy limitada. El uso de varias capas ocultas es un recurso recurrente en técnicas de Deep Learning, pero por el umbral de nuestro proyecto no vamos a cubrir más que una única capa oculta. Por otro lado, la función de activación con el paquete de R que utilizamos ("caret") está limitada a una función de activación logística o lineal. Recordemos que la función de activación añade un paso más a cada nodo de la red, permitiendo resolver problemas de forma no lineal. Entonces, cubriremos este algoritmo, pero sin posibilidad de explorar estos dos parámetros esenciales.

6.4.5.3. Parámetros y rangos estudiados

En este apartado, comprobaremos como se ajustan a las redes nuestros mejores sets de variables. Para ello tendremos que encontrar primero los parámetros más adecuados y luego aplicar las redes. En nuestro caso, trabajaremos con los dos sets de variables determinados en la selección anterior. Resumimos en la siguiente tabla los rangos y parámetros explorados.

Tabla 9: Exploración de parámetros (redes neuronales)

Parámetros	Valores explorados
Máximo de iteraciones, maxit	Estudiado con Early Stopping, con un máximo de 1200.
Parámetro de regularización, decay	Siempre (<1), y provocamos que por iteración los pesos decaigan convergiendo en el mínimo. Estudiado entre 0,01 y 0,9.
Número de nodos en la capa oculta, size	Núm. parámetros = $h(k + 1) + h + 1$

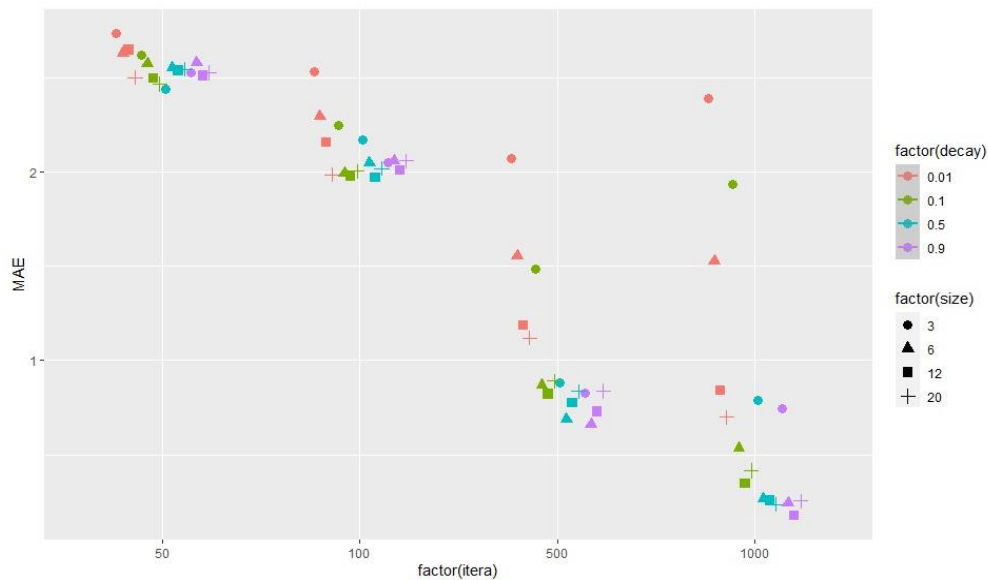
Fuente: Elaboración propia.

Set de variables 1:

Puesto que disponemos de 7.348 observaciones y estimamos que podríamos trabajar con 30 observaciones por parámetro (máximo de $7348/30 \approx 245$ parámetros) y 40 variables ("k"), calculamos el número de nodos:

$$\text{Núm. parametros} = h(k+1)+h+1 \rightarrow 245 = h(40)+h+1 \rightarrow 245 = 41h+1 \rightarrow h \approx 6 \text{ nodos}$$

Figura 40: Estudio de parámetros para redes, validación cruzada simple (4 grupos)



Fuente: Elaboración propia.

Tras explorar en validación cruzada los parámetros comentados, vemos que, efectivamente, a más iteraciones, más sobreajuste, generando más relación entre el "decay" y el error. En vez de 1.200 iteraciones nos limitaremos a construir modelos con 500 y/o 1000 iteraciones. Desestimamos el uso de "decays" por debajo del 0.1.

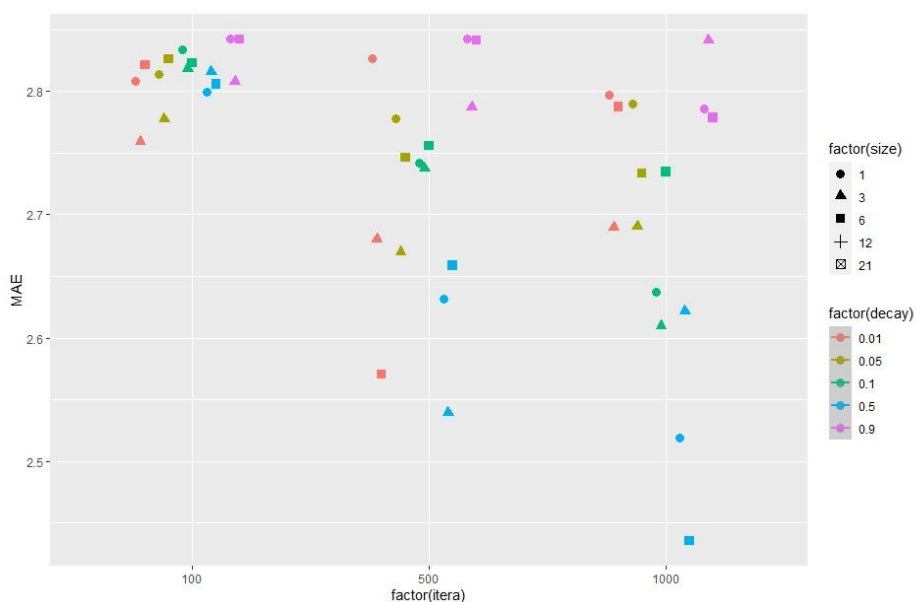
Set de variables 2:

Al igual que con el anterior set, buscamos 245 parámetros pero con 75 variables ("k"). Nos quedan 3 nodos y nuestro parámetro "size" girará en torno a este valor "h":

$$\text{Núm. parámetros} = h(k+1)+h+1 \rightarrow 245 = h(75)+h+1 \rightarrow 245 = 76h+1 \rightarrow h \approx 3 \text{ nodos}$$

El contraste entre parámetros en términos de error es bastante grande, por lo que podemos permitirnos limitar la construcción de los modelos en "h" de 6 para una ponderación de 0.01 y 0.5 en 500 y 1000 iteraciones respectivamente. Resumimos los modelos constuidos a continuación en la tabla 10.

Figura 41: Estudio de parámetros para redes, validación cruzada simple (4 grupos).



Fuente: Elaboración propia.

6.4.5.4. Evaluación y comparativa de modelos entrenados

Entre las dos exploraciones de parámetros previas los modelos construidos son:

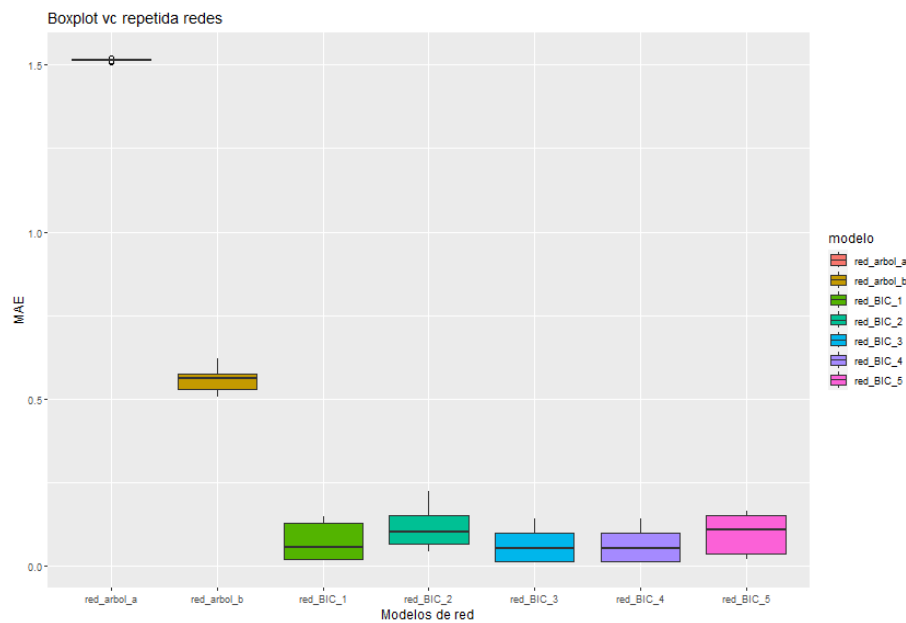
Tabla 10: Redes, parámetros y resultados en validación cruzada repetida

Model name	decay	size	iteraciones	RMSE	MAE	R ²
red_arbol_a	0.01	6	500	2.071	1.575	0.789
red_arbol_b	0.1	6	1000	1.136	0.590	0.856
red_BIC_1	0.9	12	1000	0.589	0.099	0.988
red_BIC_2	0.5	12	1000	0.588	0.109	0.995
red_BIC_3	0.5	6	1000	0.507	0.099	0.95
red_BIC_4	0.5	20	1000	0.503	0.099	0.971

Fuente: Elaboración propia, 2022.

Destaca el bajo nivel de sesgo que tienen los modelos con el primer set de variables en validación cruzada repetida, pues es muy inferior al nivel de error que logramos en los anteriores modelos. Tendremos que estudiar sobre test qué set de variables funciona mejor. Representaremos a modo de ejemplo el mejor modelo de redes en test en la próxima sección.

Figura 42: Validación cruzada repetida (4 grupos y 10 iteraciones) en redes.



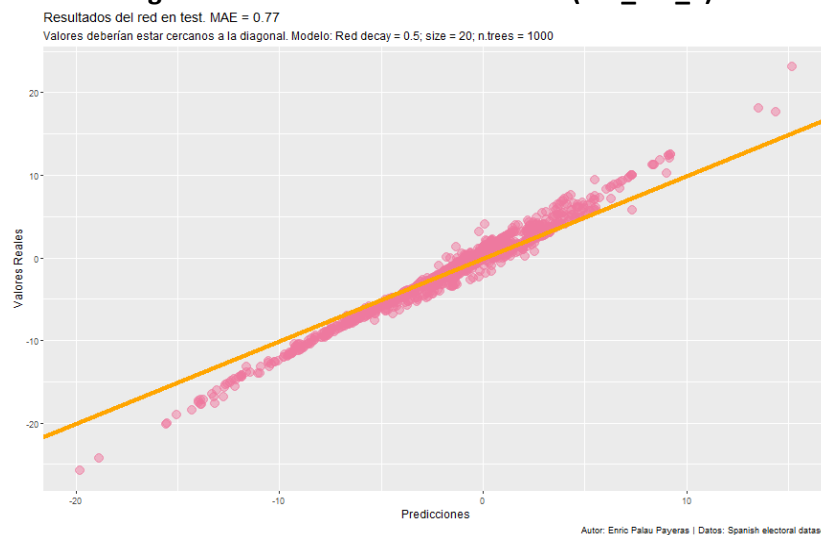
Fuente: Elaboración propia, 2022.

En términos de variabilidad, mantenemos que la elección correcta sería el modelo `red_arbol_b` con un “decay” de 0.1, un tamaño de 6 nodos y 100 iteraciones. Los niveles de MAE parecen coherentes y su balance con la variabilidad explicada es bueno. Por otro lado, no vemos excesos de diferencia entre los sesgos de redes con el primer set de variables. Sí que hay diferencias en términos de variabilidad y por eso escogemos la `red_BIC_4`.

6.4.5.5. Evaluación en test y 2023

Con el primer set de variables estudiamos los resultados de una red con 20 nodos en la capa oculta 1000 iteraciones y un “decay” del 0.5. En términos de estimación de errores por encuesta obtenemos los siguientes resultados:

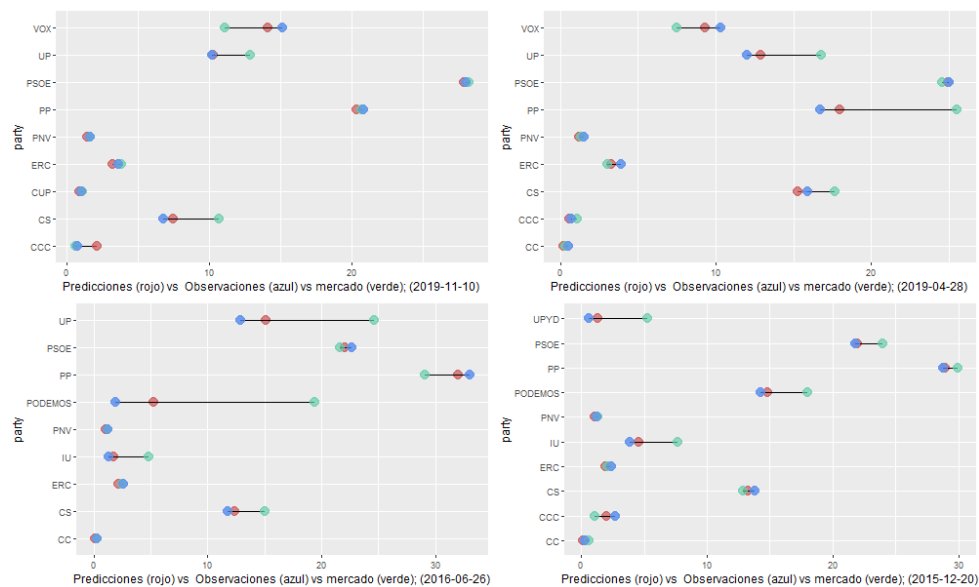
Figura 43: Errores estimados en test (`red_BIC_4`)



Fuente: Elaboración propia, 2022.

Los resultados de este modelo no son los peores, con un MAE en test de 0.77. Teniendo en cuenta la baja capacidad de parametrización que tenemos en R con “caret”, hay mucho margen de mejora y optimización. Aún así, valorando los resultados en test, este sería el segundo peor modelo ejemplificado, por delante de los árboles de decisión. No sólo tenemos un sesgo considerable, si no que a la hora de parametrizar hemos caído en un claro sobreajuste. El MAE obtenido en validación cruzada repetida y el de test no tienen nada que ver con el de esta fase. En test, vemos como a partir del error absoluto del 5% (o menos), el modelo siempre infraestima el error cometido. No hablamos de errores “outlier” pero sí de un sesgo muy sistemático que no apreciamos en la anterior validación. Por ejemplo, si el error de la encuesta es una infraestimación del 20%, el modelo predice una infraestimación del 16%. Puede ser que, si la encuesta tiene un error de sobreestimación del 20%, el modelo predice un 16%. Estos errores tan sistemáticos no los hemos podido detectar en ningún modelo más que en este.

Figura 44: Predicción en test del % de voto por partido (red_BIC_4), últimas 4 carreras

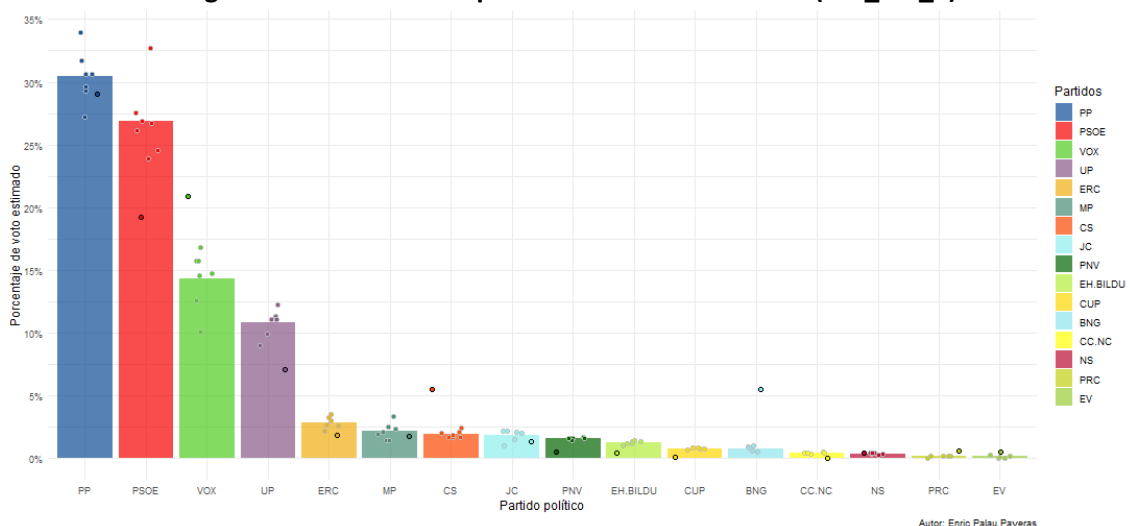


Fuente: Elaboración propia.

En términos de estimación del porcentaje de voto, se aprecia un claro empeoramiento general. Los errores tienden a ser superiores al 1% de votos y se acumula gran parte del error cometido en partidos como: UP, VOX y CS. Estos partidos son clave en nuestro nuevo contexto dónde el bipartidismo ha perdido fuerza. No es lo mismo equivocarse prediciendo hasta en un 3% el voto de CUP que un 3% de CS o el PNV. Los partidos grandes como PP y PSOE, en las últimas carreras, mantienen un comportamiento del voto bastante sistemático, por lo que hay más noción del resultado a esperar. Ahora bien, en el caso de partidos como VOX o UP, partidos relativamente jóvenes, la capacidad predictiva tiene que ser buena pues su influencia en términos de voto es clave para asegurar al PP y/o PSOE el gobierno. Es decir, un error del 1% en UP o VOX tiene que influenciar más el análisis de las estimaciones que un error del PP o PSOE del 1% o un error en CUP del 3%. Y en este caso, así sucede. Como comenta Endika: “las encuestas con sesgos continuados en el tiempo son más fáciles de predecir, por lo que también es más sencillo corregirlas” (Larrañaga, TheElectoralReport, 2022). En este caso hablamos de un error MAE del 0.72, dejando las redes como nuestro según peor modelo.

Para 2023 el modelo estima un 28.12% del porcentaje de voto al PSOE y un 19.53% al PP. Como comentábamos, el error cometido con estos partidos no es destacable y más o menos sigue el consenso de nuestros anteriores estimaciones para 2023. A VOX le asigna un 21.11% lo cual es un poco excesivo como para comprenderlo en la lectura de consenso de nuestros modelos. Pero, las elecciones de 2023 son un evento muy lejano y todo es posible. de consenso, pero el voto de CS y UP es inferior al 7%, dejándolos a la misma altura que le BNG. No es el primer modelo que pronostica un futuro muy pesimista para estos dos partidos, pero sí es el que más polarizado tiene ese “Wing Effect”. Aún así, hemos visto lo volátiles que fueron los valores de ajuste del modelo entre validaciones, por lo que no lo tomaríamos como referente pues puede estar incurriendo a un sobre ajuste.

Figura 45: Estimaciones para las elecciones de 2023 (red_BIC_4)



Fuente: Elaboración propia.

6.4.6. Máquinas de vector soporte (SVM)

A continuación vamos a indagar en el algoritmo Support Vector Machines. La idea de este modelo es buscar un hiperplano de separación para solucionar un problema de separación lineal de clases con métodos algebraicos. Este algoritmo se compone de tres conceptos básicos: “maximal margin”, “soft margin” y “kernel”. Los tres conceptos son muy extensos y requieren amplios conocimientos de algebra lineal, por lo que sólo aportaremos una breve introducción a ellos y a su parametrización. El concepto de “maximal margin” se relaciona con la separación de las observaciones por clases en un hiperplano y la intención es generar el máximo margen hallando el vector de parámetros adecuado. Luego, tenemos el “soft margin”, el permiso de fallo. Como la separación perfecta del hiperplano no suele existir, asignamos una constante de regularización C de margen o anchura de fallo (debemos asumir que habrá observaciones mal clasificadas por los separadores para no incurrir en sobreajuste). Finalmente, el “kernel”, el cual permite tratar la separación no lineal entre clases. Conseguimos este objetivo introduciendo nuevas funciones de las variables que sean no lineales, aumentando así la dimensión del vector de variables independientes, siendo así más fácil para el algoritmo encontrar separaciones lineales con buen tamaño de margen.

6.4.6.1. SVM con kernel lineal

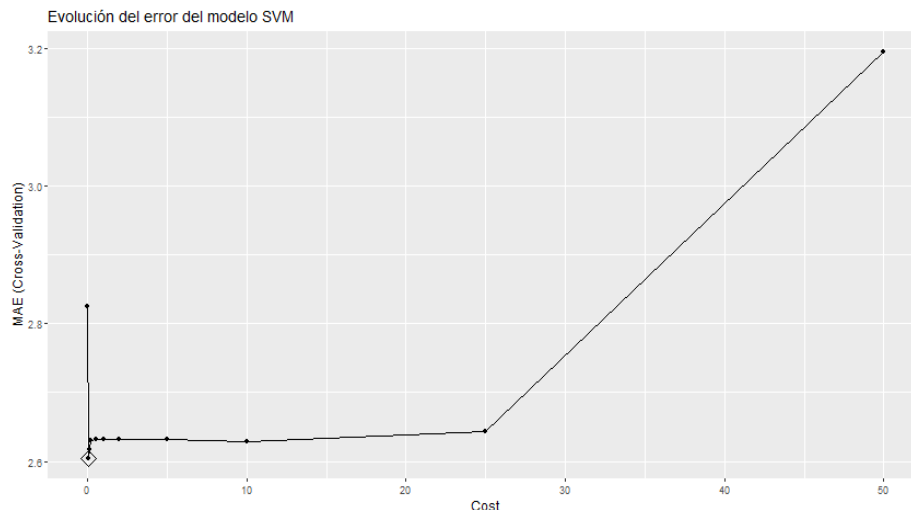
Siguiendo el estudio previo, continuaremos construyendo nuestros modelos de SVM basados en el conjunto de variables seleccionado con BIC. Aún así, debido a la utilización de “kernels”, en el ajuste de un SVM participa una matriz $n \times n$, donde n es el número de observaciones de entrenamiento. Lo que más influye en el tiempo de computación necesario para entrenar un SVM es el número de observaciones y no el de predictores. Por ello, estudiaremos este modelo con un único set de variables.

6.4.6.1.1. Parámetros y rango estudiado

En este caso, solamente se tunea el parámetro “C”. Recordemos que “C” es el parámetro inverso al permiso de fallo (+ C, + estricto, - permiso de fallo, -margen, -sesgo, + sobreajuste).

Nuestro análisis paramétrico pasará por dos fases en esta sección: 1 rango de costes laxos para cada set de variables. Conservaremos los estadísticos resultantes en “resultados” y todas las combinaciones de observaciones vs predicciones en “soluti”. El método de control será la validación cruzada (por coste computacional). El primer rango de costes será $c(0.01, 0.05, 0.1, 0.2, 0.5, 1, 2, 5, 10, 25, 50)$ y, a continuación, si se aprecia más de un valor óptimo, se pasarán a validación cruzada repetida. Ilustramos gráficamente los resultados de la exploración del parámetro de coste.

Figura 46: Estudio de parámetro C para SVM lineal, validación cruzada simple (4 grupos)



Fuente: Elaboración propia.

Vemos que el mínimo error se da con un coste=0.1, aunque con 0.05 y 10 también obtenemos buenos resultados. Recordemos que a mayor “C”, más estrictos son los modelos, menos permiso de fallo conceden en el hiperplano, menor margen de error, menor sesgo pero más sobreajuste. Aún así, el valor mínimo está en $c=0.1$ y si observamos los valores más cercanos de “C”, tienen un cambio porcentual de MAE, R^2 y RMSE ínfimo. Utilizaremos el “boxplot” para poder comparar los modelos resultantes en esta sección y la próxima.

6.4.6.2. SVM con kernel polinomial

Para posibilitar el aumento de dimensión, utilizamos el “kernel”, permitiendo hacer los cálculos que deberíamos desarrollar por el aumento de dimensión de las variables pero de una forma más sencilla y sin pasar realmente por la creación de nuevas variables. Con estos modelos seguiremos usando el mismo set de variables y probaremos usando dos “kernels” distintos.

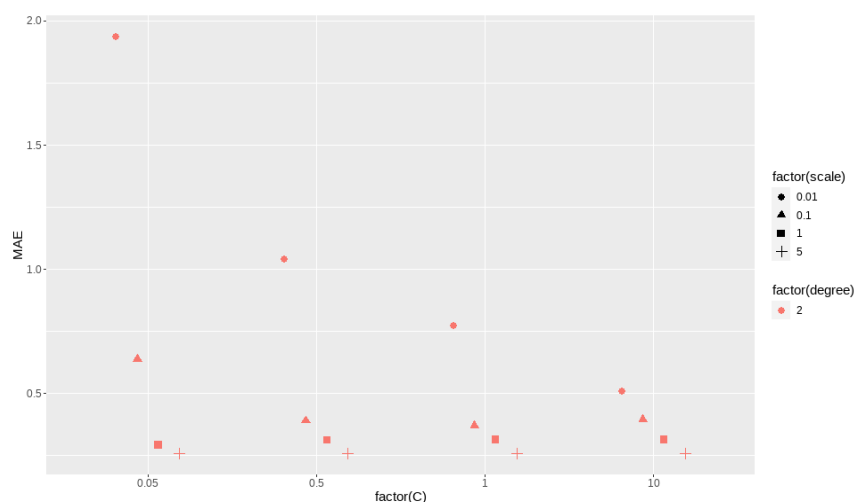
6.4.6.2.1. Parámetros y rango estudiado

En este caso, además de tunear al parámetro “C”, tenemos en cuenta el grado del polinomio, el cual es importante tunearlo para la relación sesgo-sobreajuste con el parámetro “scale”. Debido al gran coste computacional de este, como propuesta alternativa, usamos el siguiente estudio paramétrico. Tras tratar de realizar el tuneado y obtener la rejilla (según la metodología convencional), hemos detectado un gran requerimiento computacional e imposibilidad de finalizar el proceso. Es por ello, que nos hemos visto obligados a optar por la creatividad con un nuevo orden del mismo proceso. Primero, estudiar una ancha rejilla de “C” entre 0.001 y 10 y “scale” de 0.001 a 5 en segundo grado. Esto permite ver si en grado dos hay una clara tendencia entre los valores de la rejilla y descartar valores demasiado grandes o pequeños. Segundo, acertamos la rejilla en los rangos óptimos y enfrentamos en grado 2 y 3. De este modo, vemos cuál de los dos grados es mejor en un corto rango de parámetros buscando el menor coste computacional posible.

SVM con grado polinomial 2:

Nos fijaremos en las formas de las figuras (“scale”), su ubicación en el eje horizontal (“C”) y la altura de la observación (MAE) para determinar los parámetros óptimos en SVM saturado polinomial de grado 2.

Figura 47: Estudio de parámetros para SVM (kernel 2), validación cruzada simple (4 grupos)



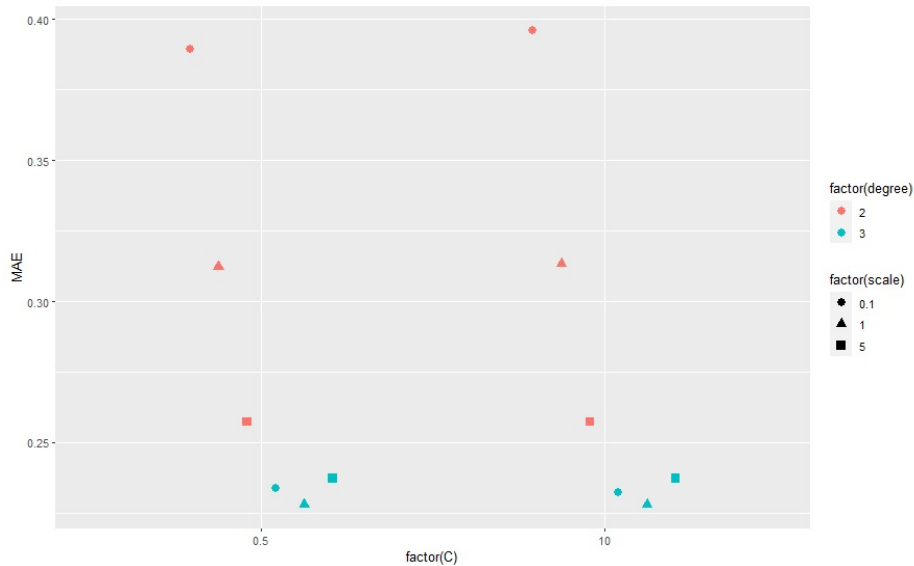
Fuente: Elaboración propia.

Con esto determinamos el próximo análisis en: la escala de 0.1, 5 y 1 con costes de 0.5 y 10.

SVM con grado polinomial 3:

Nos fijaremos en las formas de las figuras (“scale”), su ubicación en el eje horizontal (C) y la altura de la observación (MAE) para determinar los parámetros óptimos en SVM saturado polinomial de grado 2.

Figura 48: Estudio de parámetros para SVM (kernel 93), validación cruzada simple (4 grupos)



Fuente: Elaboración propia.

Vemos que el grado 3 supera en términos de MAE al grado 2 en todos los rangos de valores estudiados. Aún así conservaremos un ejemplo de “kernel” polinomial en grado 2. Los parámetros óptimos en SVM polinomial de grado 3 son: “C”=0.5 y “scale”= 1.

6.4.6.3. Comparativa y evaluación de modelos entrenados

A continuación vemos los diferentes ejemplos de parametrización óptima en validación cruzada repetida:

Tabla 11: SVM, parámetros y resultados en validación cruzada repetida

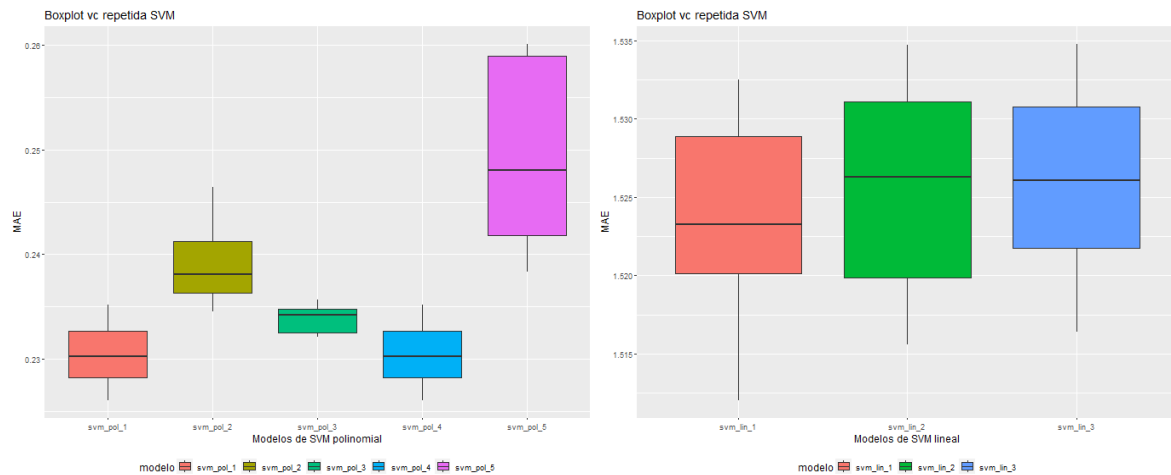
Model name	C	scale	degree	RMSE	MAE	R^2
svm_lin_1	0.05	NA	NA	2.60	1.523	0.628
svm_lin_2	0.1	NA	NA	2.632	1.525	0.619
svm_lin_3	10	NA	NA	2.632	1.525	0.619
svm_pol_1	0.5	1	3	0.356	0.230	0.992
svm_pol_2	0.5	5	3	0.4	0.239	0.99
svm_pol_3	10	0.1	3	0.302	0.233	0.995
svm_pol_4	10	1	3	0.356	0.230	0.992
svm_pol_5	0.5	5	2	0.649	0.249	0.966

Fuente: Elaboración propia.

El uso de “kernels” marca en gran parte la diferencia entre los modelos planteados. En nuestro caso, tenemos un problema dónde se acumulan las dimensiones entre partidos,

encuestadoras, carreras, días para las elecciones etc. Por lo que el uso de un modelo basado únicamente en la separación de hiperplanos no es muy conveniente. Al transformar esa separación en una forma irregular como pasa con el “kernel” = 3, podemos llegar a saltar esas dimensiones dando posiblemente con la separación oportuna.

Figura 49: Validación cruzada repetida (4 grupos y 10 iteraciones) en SVM



Fuente: Elaboración propia.

Si el uso de “kernel” ya ha sido un factor determinante, el grado del “kernel” polinomial ha marcado una diferencia abismal entre los modelos. En términos de sesgo, los modelos SVM_pol_4 y SVM_pol_1 serían nuestros mejores modelos, con un MAE de 0.23. La diferencia entre estos dos modelos es el coste asignado, pues el grado del “kernel” y la escala son las mismas. Entendemos que el valor del coste, entonces, converge. Aún así, el balance con la variabilidad no es tan buena como la del SVM_pol_3, por lo que nos centraremos en ese cambio de escala que nos ha permitido reducir la variabilidad. Por la escasa diferencia de sesgo entre estos modelos optaremos por el SVM_pol_3 como ejemplo de ilustración en test.

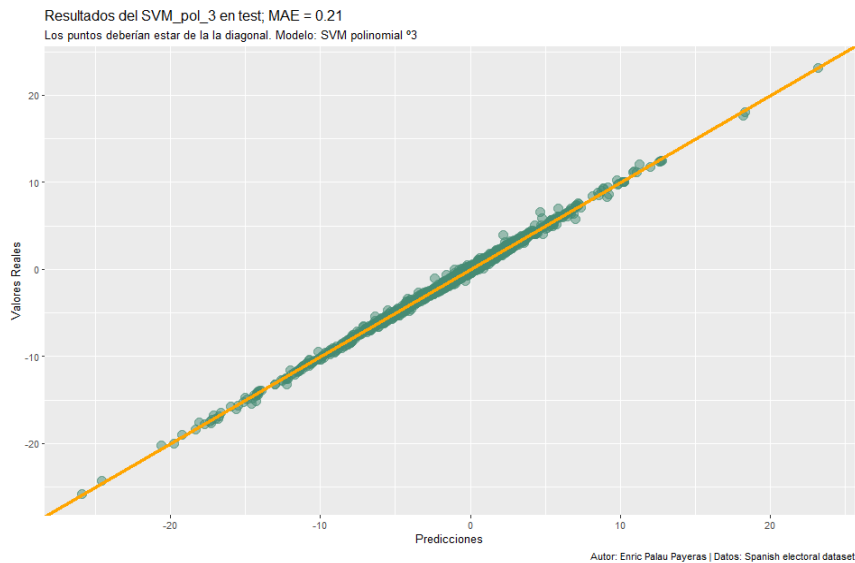
6.4.6.4. Evaluación en test y 2023

Los resultados del SVM_pol_3 en test son muy similares a los resultados obtenidos en “train”, por lo que el modelo es estable. En términos de ajuste, probablemente estamos hablando del mejor modelo en todo el proyecto. Con un MAE de 0.21 y un R^2 del 99%, tenemos un modelo que es capaz de predecir y explicar casi a la perfección el error cometido por cada encuesta.

Probablemente, debido a la naturaleza de la base de datos, deberíamos explorar varios métodos de partición “train”/“test” para asegurar al 100% estos resultados casi inverosímiles. De todos modos, debemos considerar que hacemos pasar los nuestros por 4 tipos de validación distintas, por lo que dejaremos umbral de mejor en este aspecto para próximas investigaciones. En comparativa con otros modelos en test, aquí ya hemos conseguido corregir el problema con las encuestas “outlier”. Nuestro modelo, independientemente de lo “outlier” que sea el error cometido por la encuesta, es capaz

de identificarlo. En otras palabras, este modelo ha superado en términos absolutos a los previos.

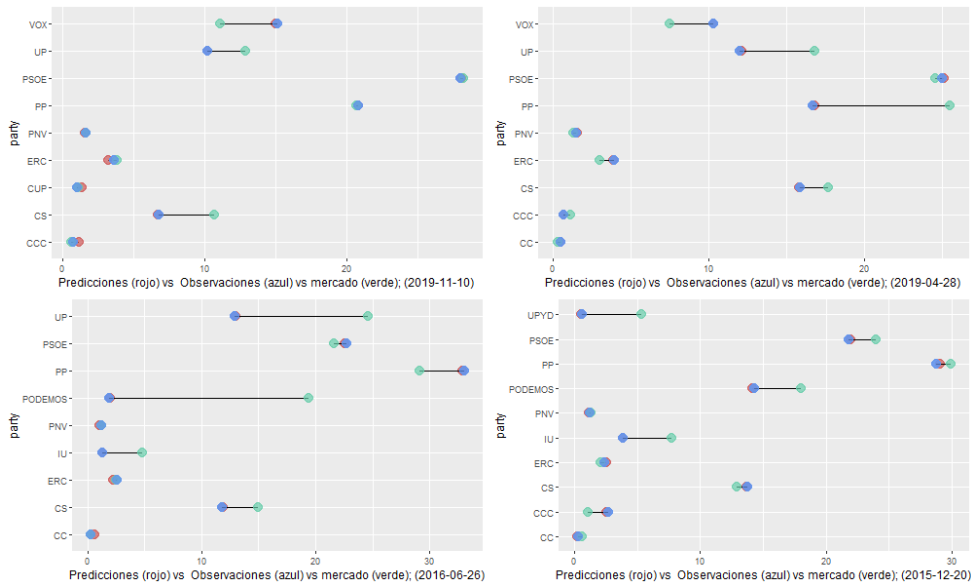
Figura 50: Errores estimados en test (SVM_pol_3)



Fuente: Elaboración propia.

En términos de estimación de voto, nuestro modelo parece ser también el mejor hasta la fecha. Con un MAE del 0.22, supera todos los promedios, casas y modelos evaluados hasta el momento. Es más, el máximo error cometido es una infraestimación de 1.5 puntos porcentuales a AP para la carrera del 1993. Si evaluamos el MAE en estimación de voto, vemos que el valor se mantiene estable con un 0,21. Esa estabilidad en el criterio de bondad de ajuste, entre pruebas de validación, nos da confianza para pensar que no estamos tratando con un modelo sobre ajustado. Es más, lo que considerábamos dificultades para el resto de los modelos como UP y Podemos en 2016, en este caso, no han sido un problema.

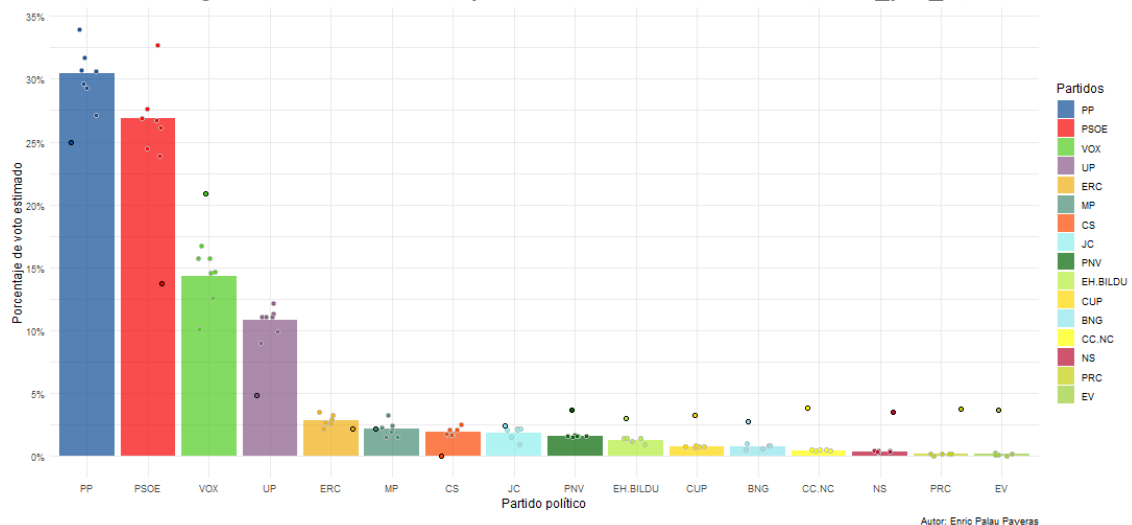
Figura 51: Predicción en test del % de voto por partido (SVM_pol_3), últimas 4 carreras



Fuente: Elaboración propia.

Viendo los buenos resultados en test, será interesante evaluar las estimaciones del modelo para 2023. Debemos recordar que estos algoritmos están limitados en muchos aspectos y no deben ser interpretados como una predicción exacta del “outcome” electoral. Más bien, debemos considerarlos como una herramienta de análisis para auto proyectar eventos futuros. Por ello es importante hacer un buen modelo y evaluarlo correctamente. En este caso, la predicción para 2023 difiere bastante de los anteriores modelos pues propone a VOX con un 20% de la estimación de voto por delante del PSOE con un 14% de la estimación de voto. También ha asumido estimaciones de voto muy optimistas para partidos que no suelen superar el 2% y ha dejado a UP en un 5%. Por otro lado, ha penalizado CS sin acercarse apenas al 1%. No podemos valorar la verosimilitud de estas estimaciones, más alimentando el modelo con encuestas tan prematuras. De todos los eventos estimados entre nuestros algoritmos, este es el más disruptor y el que menos sigue la lectura consenso. Recordamos que el conjunto de test_2023 carece de observaciones heterogéneas. Con el tiempo y el aumento de este recurso, vamos a poder extraer más conclusiones.

Figura 52: Estimaciones para las elecciones de 2023 (SVM_pol_3)



Fuente: Elaboración propia.

7. Evaluación comparativa entre encuestas, promedios y modelos

Tras haber testado diferentes modelos hasta identificar uno aparentemente bueno para nuestra casuística en el apartado anterior, a continuación, realizamos un último ejercicio pendiente.

Entre los rankings de encuestadoras tenemos un consenso bastante claro de cuáles son las mejores en España, por lo que vamos a revisar los resultados pronosticados por dos de nuestros modelos contra los que estiman las mejores encuestadoras (así como también sus promedios). De esta manera, pondremos en práctica y funcionamiento todo el trabajo de investigación desarrollado en este trabajo.

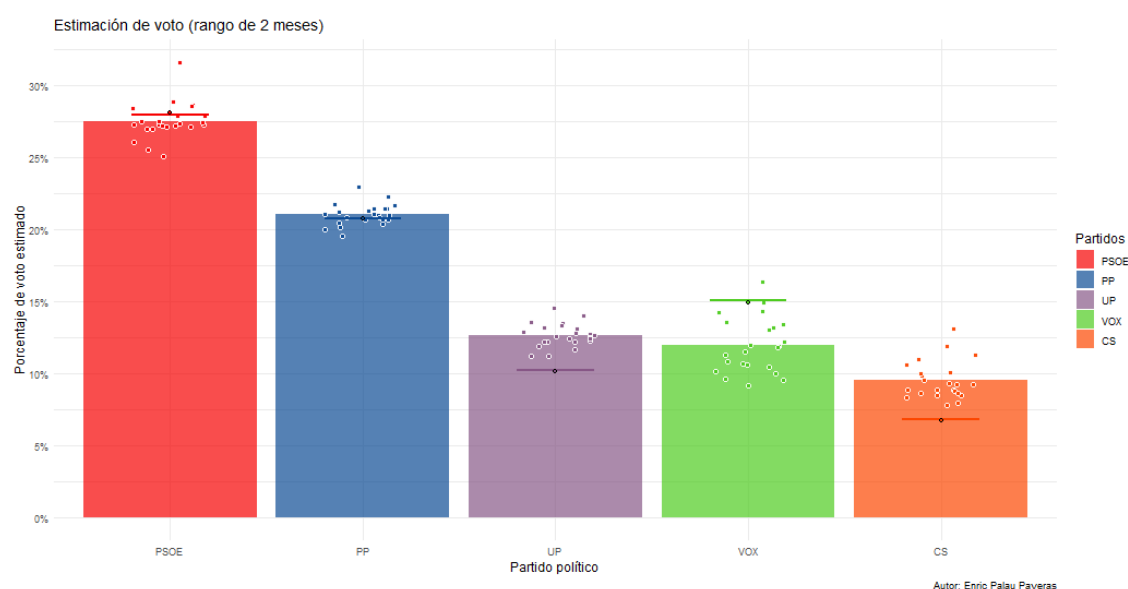
Figura 53: Ranking de encuestadoras de TheElectoralReport

TheElectoralReport										
Interactivos Artículos Autores Newsletter Sobre el blog										
ENCUESTADORA	CALIFICACIÓN γ	ENCUESTAS ANALIZADAS	COBERTURA MULTIPARTI	TAMAÑO MUESTRA	ERROR MUESTRAL	MAE TOTAL	MAE PONDERADO	MAE EVOLUCIÓN	+/- ESPERADO	
Sigma Dos	5	123	✓	1,800	2.4	1.8	1.8	2.1 \rightarrow 1.0	-0.5	
GAD3	5	44	✓	2,188	2.1	1.7	2.0	2.5 \rightarrow 1.0	-0.4	
SocioMétrica	5	24	✓	1,478	2.6	1.2	1.5	2.2 \rightarrow 1.4	-0.8	
Ipsos	5	25	✓	18,535	0.7	1.6	1.6	3.7 \rightarrow 1.5	-0.5	
CIES	5	10	✓	1,501	2.6	1.1	0.1	0.5 \rightarrow 0.0	-1.3	
Opina	4 5	84	✗	1,305	2.8	1.7	1.5	—	-0.6	
Celeste-Tel	4 5	45	✓	1,100	3.0	1.7	2.1	1.9 \rightarrow 1.3	-0.5	
IBES	4 5	5	✓	1,800	2.4	1.6	1.6	3.0 \rightarrow 0.8	-0.8	
GfK	4 5	3	✓	15,864	0.8	1.5	1.8	2.5 \rightarrow 1.4	-1.1	

Fuente: (Nuñez, 2019).

Sabiendo que Sigma Dos, GAD3 y SocioMétrica están en el top 3 entre las encuestadoras, las tomaremos de ejemplo para comparar, en términos de MAE, la capacidad predictiva de nuestros modelos. En el siguiente gráfico, tenemos, las encuestas de estas tres casas en los dos meses previos a las elecciones (puntos con perfilado blanco) con el promedio de mercado (barras) y la estimación de nuestro modelo (punto con perfil negro). Las líneas verticales representan los resultados reales.

Figura 54: Casas, encuestas, promedios y modelos; Evaluación en test



Fuente: Elaboración propia.

Comparamos el MAE del 0.1 (0.22 sobre la predicción de errores) logrado por nuestro mejor modelo (SVM_pol_3) con el performance de las tres mejores casas, entre los dos meses previos a las elecciones. En conjunto, el MAE de las tres casas es de 2.07 y GAD3 se coronaría con un mae del 1.9, seguida de SOCIOMÉTRICA y SIGMA_DOS con un MAE de 2.1 ambas. Hemos superado con creces las principales encuestadoras en términos de estimación de voto. Es más, el error máximo cometido por los promedios de mercado es de 4 puntos, mientras el de nuestro modelo es del 0.1. En el caso de GAD3, SIGMA_DOS y SOCIOMÉTRICA su MAE conjunto (o de submercado) sería de 2. Como indicábamos al principio, el sesgo de las encuestas es bajo y no lo hacen nada mal si

consideramos que las encuestas más sesgadas se desviaban en 5 puntos porcentuales. También sabemos que algunas encuestas han llegado a alcanzar errores de tan solo 0.2 o incluso 0.01. Es cierto, que por lo general nuestro modelo suele “corregir” el error de la encuesta pues es su variable objetivo, generando un mejor ajuste ajustándose mejor. Pero también hemos podido detectar casos en el que estas encuestadoras en su promedio lo han hecho mejor que nuestro modelo. En el caso de la última carrera tuvimos muy buen ajuste y un claro efecto corrector del modelo. Luego, los promedios, en muchos casos logran mejorar el sesgo y suavizan el efecto de cambios bruscos, pero si todas las encuestas del promedio muestran la misma tendencia o dirección, el sesgo final será el mismo (caso de las elecciones del 2015). Además, predecir mediante predictores que ya tienen un sesgo bajo es muy ventajoso, pero puede implicar a fallidas de sistema.

Del mismo modo, comparamos que se estima para 2023 en medios como ElPlural, y lo que estiman nuestros modelos. Los resultados descritos a continuación, se encuentran en una tabla resumen dentro de nuestros anexos (punto E). ElPlural tiene una sección llamada “La madre de las encuestas” en la que introducen su estimación como la definitiva pues se basan en las que consideran las mejores encuestadoras. Incluyen las encuestas de Electomania, Celeste Tel, DYM, IMOP, Simple Lógica, Invymark y ElPlural.com. Para el PP con Alberto Núñez, estiman que podría llegar liderar la campaña con más de un 30%. Se repiten estos resultados entre otros medios como TheElectoralReport, GAD3... Entre nuestros modelos el máximo ha sido de 28% y las estimaciones más comunes entre el 26 y 27%. Hablamos de diferencias bastante consistentes, pero ninguno de los resultados parece incongruente. El PSEOE y UP parecen tener un futuro electoral poco óptimo tanto por parte de nuestros modelos como de las encuestadoras en general. El PSOE tantea valores estimados del 26,6% y el 24%. Según nuestros modelos suele pasar por un valores cercanos al 20%. Aún así, el que considerábamos nuestro mejor modelo llegó a estimar cerca de un 14% lo cual sería un hito histórico, bastante inverosímil. Del mismo modo, las estimaciones de UP suelen rondar el 5% cuando las encuestadoras y promedios estiman cerca del 10%. Finalmente, para VOX, las estimaciones de nuestros modelos rondan el 15% y 17%, mientras según las encuestas será un 15%. Para CS las estimaciones son muy volátiles entre medios, e igual entre nuestros modelos. Es cierto que la estimación máxima está en el 6% pero muchos modelos asumen que el partido pueda pasar a un 1% o 0%. Por lo tanto, vemos una clara tendencia y armonía entre las estimaciones, pero al tratar un evento futuro a casi un año de plazo, es difícil poder valorar las estimaciones actuales.

En definitiva, nuestros modelos han demostrado ser aplicables en un trabajo de campo real como son las elecciones de 2023. Y vemos que hay una lectura de consenso entre modelos, promedios y encuestadoras. Aún así, como investigadores y/o divulgadores, debemos entender que las estimaciones de nuestros algoritmos se basan en correlaciones, funciones y otros aspectos meramente matemáticos. No podemos determinar causalidad en base a nuestros modelos y por ello, al hablar de nuestros resultados, debemos aclarar que son meras estimaciones. Nuestros modelos son una herramienta de lectura y son útiles para guiar y ofrecer nuestra percepción del futuro electoral en España.

8. Conclusiones

Haciendo retrospección, podemos concluir que hemos alcanzado los tres objetivos inicialmente propuestos y que, además, hemos logrado aportar a nuestro estudio una aplicación funcional en el sistema electoral de nuestro país.

Primero, conseguimos crear una ETL para el histórico electoral en España. Con este proyecto, podremos tratar de nuevo con bastante facilidad problemas y estudios relacionados con datos y elecciones. Segundo, logramos crear una metodología propia para predecir la intención de voto. A pesar de no haber podido comparar los procedimientos de otros autores (pues son proyectos lucrativos), hemos podido coger pinceladas y conceptos de varios profesionales en el área para culminar con nuestro propio proceso. Finalmente, en tercer y último lugar, hemos estudiado y aplicado una gran variedad de algoritmos de Machine Learning, entendiendo así la naturaleza de los propios algoritmos y de los datos estudiados.

Entre nuestros modelos, los basados en árboles funcionan relativamente bien con este problema, tal como vimos entre modelos de Bagging, Random Forest y GBM. A pesar de ello, el modelo que mejor ha funcionado fue la máquina de vector soporte con kernel polinomial de grado 3, obteniendo un MAE de 0,21 en test. Hemos observado que los modelos con mejor resultado son relativamente complejos y que por lo tanto, las relaciones entre nuestras variables predictoras y objetivo no son del todo lineales. También hemos logrado un muy buen entendimiento de las variables predictoras, teniendo algunos de los mejores modelos con tan sólo 41 variables de 133 “dummies”. Vimos que los datos de encuestas y promedios son los bloques de variables que más información aportan y los más seleccionados entre nuestros modelos.

Cumplir nuestros objetivos particulares nos ha permitido también alcanzar la meta general de este campo de estudio: entender ese MAE del 2,23 de las encuestas y, a su misma vez, optimizar las estimaciones en intención de voto, llegando hasta un MAE del 0,21. Hemos aplicado nuestros mejores modelos a los datos de 2023 y hemos realizado estimaciones para las próximas elecciones. De este ejercicio hemos podido concluir la aplicabilidad funcional de nuestro modelo y la ampliación de conocimiento en este campo. Finalmente, hemos podido evaluar como nuestros modelos han superado las estimaciones y promedios de las mejores casas en el histórico. Como es el caso de GAD3 con un MAE de 1.9 entre las encuestas de los últimos dos meses de carrera. Vista la eficiencia general de los modelos planteados. Concluimos también, que el proceso previo de ETL y la metodología planteada, podrían ser la clave para obtener modelos que con poca complejidad, resulten ágiles en términos de sesgo.

De cara a futuras investigaciones en este campo, proponemos tratar una serie de ideas a partir de este mismo estudio. En el estado del arte ya comentamos lo mucho que se suele recomendar incluir datos por comunidades. También recomendamos aplicar estudios mediante técnicas de aprendizaje no supervisado (por ejemplo, análisis de componentes principales o análisis clúster) para reducir la dimensionalidad de la base de datos. Con una infinidad de posibilidades, dejamos las puertas abiertas para extender este proyecto a nuevos enfoques y utilidades.

Bibliografía

- SAS Institute Inc. (2017, 30 agosto). *Introduction to SEMMA*. SAS. Recuperado 23 diciembre 2022, de <https://documentation.sas.com/doc/en/emref/14.3/n061bzurmej4j3n1jnj8bbjm1a2.htm>
- Appvizer. (2021, 16 febrero). *Procesa exitosamente tus datos gracias al ETL*. Appvizer. Recuperado 23 diciembre 2022, de <https://www.appvizer.es/revista/it/etl/etl>
- Beal, D. J. (2007). Information Criteria Methods in SAS® for Multiple Linear Regression Models . *Science Applications International Corporation*, 1-8. <https://analytics.ncsu.edu/sesug/2007/SA05.pdf>
- Datosmacro (s.f.). España: *Economía y demografía*. Datosmacro. Recuperado 10 agosto 2022, de <https://datosmacro.expansion.com/paises/espana>
- Grupo Crecimiento Verde. (2019, 6 mayo). *El medio ambiente pesa en la decisión de voto*. Recuperado 10 agosto 2022, de <https://grupocrecimientoverde.org/el-medio-ambiente-pesa-en-la-decision-de-voto/>
- Guasch, A. (Ed.). (2019, 25 abril). *Bipartidismo: un sistema en horas bajas*. LaVanguardia. Recuperado 7 junio 2022, de <https://www.lavanguardia.com/vida/junior-report/20190423/461818508054/bipartidismo-elecciones-generales-espana.html>
- IPSOS, I. S. (2019, abril). 2019 *European Parliament Elections Study of Potential Voters*. IPSOS. https://europeanclimate.org/wp-content/uploads/2019/04/European-Parliament-Study_Media_EU.pdf
- Larrañaga, E. N. (2020, 23 septiembre). *Cómo funciona el promedio de encuestas*. TheElectoralReport. Recuperado 7 junio 2022, de <https://electoralreport.com/articulos/metodologia-promedio-de-encuestas/>
- Liébana, J. A. (2021). Apuntes Máster en Minería de datos e Inteligencia de negocios: SEMMA.
- Llaneras, K. (2018, 26 marzo). *Election forecast everywhere*. Medium.com. Recuperado 7 junio 2022, de <https://medium.com/@kikollan/election-forecast-elpais-6cc1e2f9d384>
- Llaneras, K. (2021, 21 abril). *Así están las encuestas en Madrid: las opciones de ganar para izquierda y derecha*. EL PAÍS. Recuperado 7 junio 2022, de <https://elpais.com/espana/elecciones-madrid/2021-04-21/asi-estan-las-encuestas-en-madrid-las-opciones-de-ganar-para-izquierda-y-derecha.html>

- Sáez L., J.L., y Jaime C., A. M. (2014). Atribución de la responsabilidad y voto económico El caso de España. *El trimestre económico*. 81(324), 71-81.
<https://www.eltrimestreeconomico.com.mx/index.php/te/article/view/369/702>
- Masters, A. B. (2020, 5 septiembre). *Estimating House Effects*. Medium.com. Recuperado 7 junio 2022, de <https://medium.com/swlh/estimating-house-effects-5c465f2aca87>
- Ministerio del Interior, Gobierno de España. (s.f.). *INFOELECTORAL*. Recuperado 7 julio 2022, de <https://infoelectoral.interior.gob.es/opencms/es/elecciones-celebradas/resultados-electorales/>
- Núñez L., E. (2019). *Ranking de encuestadoras*. TheElectoralReport. Recuperado 7 marzo 2022, de <https://electoralreport.com/interactivos/ranking-encuestadoras/>
- Núñez L., E. (2022, 28 julio). *Claves para entender el resultado andaluz: sexo, edad, estudios y transferencias de voto*. TheElectoralReport. Recuperado julio 2022, de <https://electoralreport.com/articulos/claves-resultados-andalucia-transferencias-demografia/>
- Ortega, A. L. (s.f.). *¿Cómo funciona el mercado de predicción?*. PREDI 10N. Recuperado 10 noviembre 2022, de <https://aloport.github.io/predi/projects.html>
- Peinado, M. L. (2014, 5 noviembre). *No te líes: esta es la diferencia entre intención de voto directo y estimación de voto*. EL PAÍS. Recuperado 7 julio 2022, de https://verne.elpais.com/verne/2014/11/05/articulo/1415189081_000068.html
- Perdomo, C. J. (2008). ¿Se pueden predecir geográficamente los resultados electorales? Una aplicación del análisis de clusters y outliers espaciales. *Centro de Investigación y Docencia Económicas (CIDE), Estudios demográficos y urbanos. Ciudad de México: SciELO*, 23, 3(69), 571-613
<https://www.scielo.org.mx/pdf/educm/v23n3/2448-6515-educm-23-03-571.pdf>
- Portela, J. (2022). *Técnicas de machine learning*. UCM, Estudios Estadísticos. Máster en minería de datos e inteligencia de negocio.
- Rodrigo, J. A. (2020, 1 noviembre). *Machine Learning con R y caret*. Cienciadedatos.net. Recuperado 7 junio 2022, de https://www.cienciadedatos.net/documentos/41_machine_learning_con_r_y_caret#Anexo5:_M%C3%A9tricas

- Silver, N. (2012, 22 junio). *Calculating 'House Effects' of Polling Firms*. FiveThirtyEight. Recuperado 5 mayo 2022, de <https://fivethirtyeight.com/features/calculating-house-effects-of-polling-firms/>
- Silver, N. (2020, 12 agosto). *How FiveThirtyEight's 2020 Presidential Forecast Works — And What's Different Because Of COVID-19*. Fivethirtyeight. Recuperado 5 mayo 2022, de <https://fivethirtyeight.com/features/how-fivethirtyeights-2020-presidential-forecast-works-and-whats-different-because-of-covid-19/>
- Wikipedia (2000). Opinion polling for the 2000 Spanish general election. Recuperado 5 mayo 2022, de https://en.wikipedia.org/wiki/Opinion_polling_for_the_2000_Spanish_general_election
- Wikipedia (s.f.). Anexo:Presidentes del Gobierno de España. Recuperado 10 de agosto 2022, de https://es.wikipedia.org/wiki/Anexo:Presidentes_del_Gobierno_de_Espa%C3%B1a

Anexos

A. Código R y SAS:

Disponibles ambos, código R y SAS en el siguiente enlace:
https://github.com/3nriket/TFM_encuestas_y_elecciones_EPP_UCM_2023.git

B. Listado de variables por bloques:

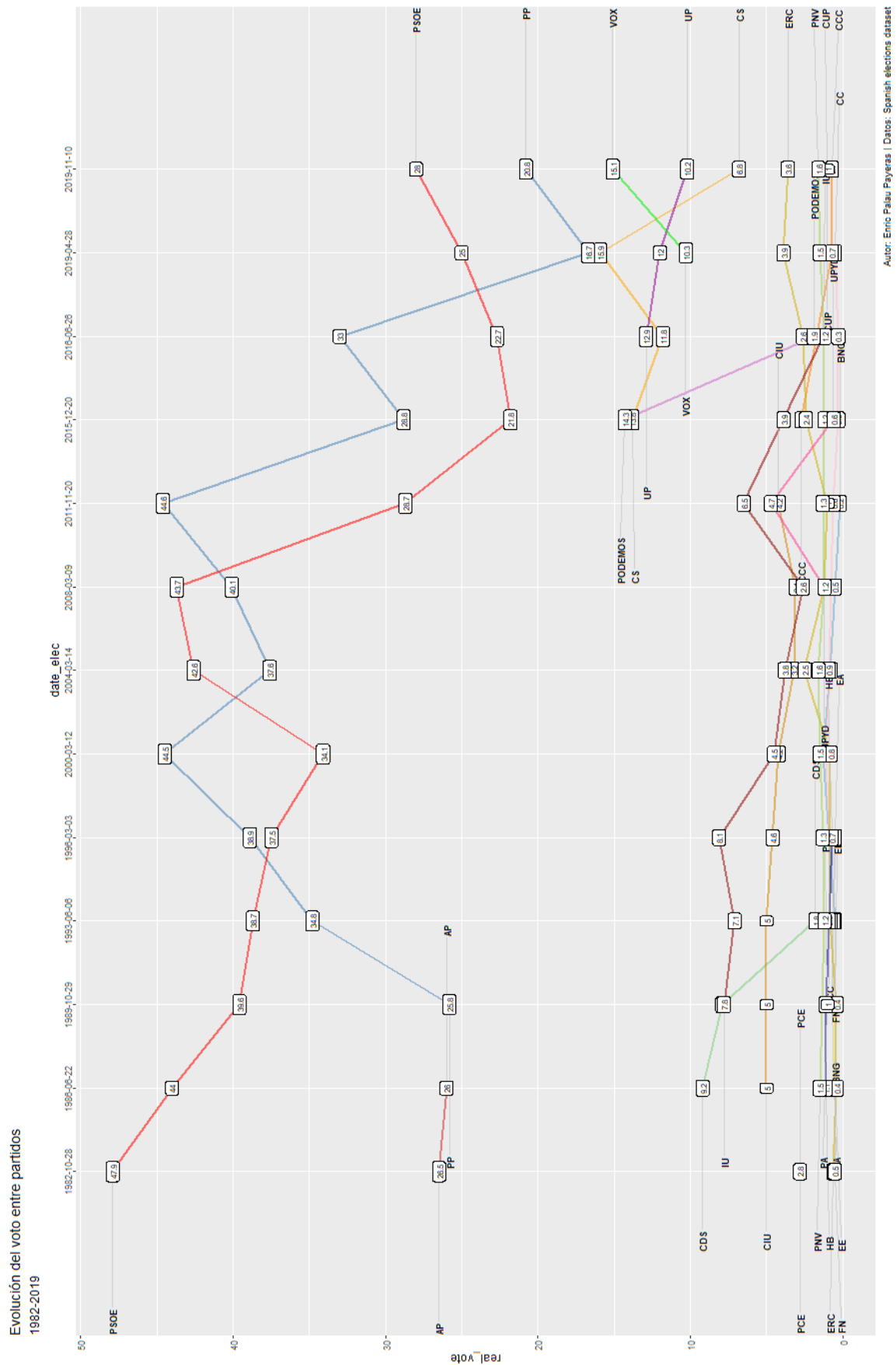
Grupo	Subgrupo	Variables	Descripción
wikipedia	n	n	Tamaño muestral de la encuesta
	porc_surveys_firm	porc_surveys_firm	Porcentaje de encuestas realizadas por la casa sobre el total
	est_surv_vote	est_surv_vote	Estimación de la intención de voto
	poll_firm	poll_firm_CIS	poll_firm: casa encuestadora
		poll_firm_GALLUP	
		poll_firm_IMOP	
		poll_firm_NC_REPORT	
		poll_firm_OPINA	
		poll_firm_SOCIOMÉTRICA	
		poll_firm_DYM	
		poll_firm_GESOP	
		poll_firm_METROSCOPIA	
		poll_firm_NOXA	
		poll_firm_SIGMA_DOS	
		poll_firm_TNS_DEMOSCOPIA	
		poll_firm_ASEP	
		poll_firm_ELECTOPANEL	
		poll_firm_HAMALGAMA_MÉTRICA	
		poll_firm_MYWORD	
		poll_firm_OBRADOIRO_SOCIO	
		poll_firm_SIMPLE_LÓGICA	
		poll_firm_VOX PÚBLICA	
		poll_firm_CELESTE.TEL	
		poll_firm_GAD3	
	party	party_AP	party: partido
		party_CDS	
		party_EE	
		party_HB	
		party_PCE	

		party_PSOE	
		party_CC.NC	
		party_CUP	
		party_EV	
		party_MP	
		party_PP	
		party_UPYD	
		party_BNG	
		party_CIU	
		party_EH.BILDU	
		party_IU	
		party_PNV	
		party_UCD	
		party_CC	
		party_CS	
		party_ERC	
		party_JC	
		party_PODEMOS	
		party_UP	
		party_CCC	
		party_EA	
		party_FN	
		party_NS	
		party_PRC	
		party_VOX	
		party_PA	
	wing	wing_RIGHT	wing: ala ideológica del partido
		wing_LEFT	
	lead_party	lead_party_CS	Partido en cabeza de la campaña
		lead_party_UCD	
		lead_party_PODEMOS	
		lead_party_PP	
		lead_party_PSOE	
	lead2_party	lead2_party_ARM	Segundo partido en cabeza
		lead2_party_PP	
		lead2_party_VOX	
		lead2_party_AP	
		lead2_party_PODEMOS	
		lead2_party_UP	
		lead2_party_EA	
		lead2_party_UCD	
		lead2_party_CS	
		lead2_party_PSOE	
infoelectoral	date_elec	year_elec	Año de la carrera
	days_to_elec	days_to_elec	Días para las elecciones

		urna_0	Rangos de evaluación según los días para las elecciones
		urna_7	
		urna_15	
		urna_60	
		urna_365	
		exit_poll	Encuesta a pie de urna "true" o "false"
	promedios	prom_casa_partido	Promedios y estadísticos de encuestas
		prom_carrera_wing	
		house_effect_e	
		prom_casa_wing	
		prom_carrera_casa_partido	
		prom_carrera_partido	
		prom_carrera_casa_wing	
		wing_effect_e	
		prom_general_partido	
		prom_general_wing	
datos fundamentales	economia	eco_pib_percap	PIB per capita
		eco_pib_var	Variación del PIB
		eco_fisc_ing_percap	Ingreso fiscal per cápita
		eco_fisc_ing	Ingreso fiscal per cápita
		eco_deficit	Deficit económico
		eco_debt_percap	Deuda per cápita
		eco_unemployment	Desempleo
		eco_smi	Salario Mínimo Interprofesional
		eco_rate_avg	Salario Medio
	población	pobl	Población
		pobl_densidad	Densidad poblacional
		pobl_fem_porc	Porcentaje de mujeres en la población
		pobl_em_rate	Ratio de emigrantes
		pobl_im_rate	Ratio de inmigrantes
		pobl_suicide	Suicidios
		pobl_suicide_percienmil	Suicidios por cien mil
		pobl_kill	Homicidios
		pobl_kill_percienmil	Homicidios por cien mil
		pobl_life_expectancy	Esperanza de vida
		pobl_pobreza_rate	Población (porcentual) en riesgo de pobreza

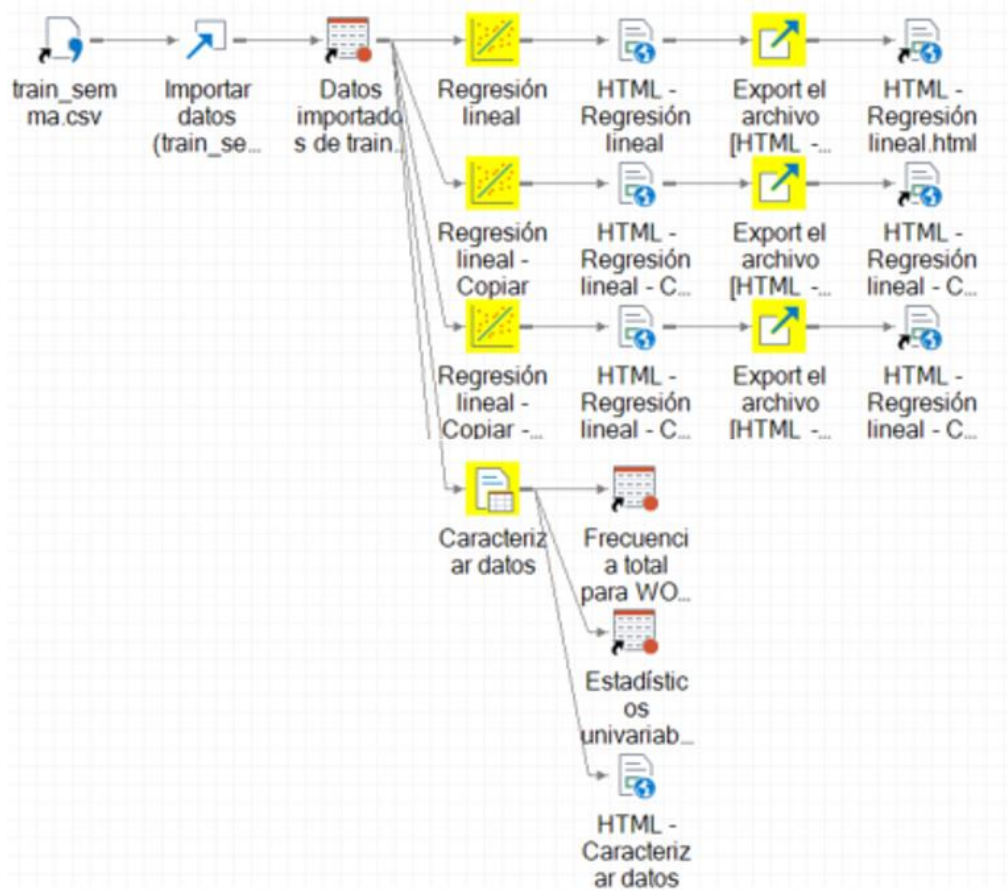
		pobl_idh	Índice de desarrollo humano
	medioambiente	env_gwh_consum	Consumo en gwh
		env_kwh_consum_percap	Consumo eléctrico per cápita
		env_gwh_prod	Producción energética en gwh
		env_gwh_prod_renovable	Producción energética en gwh de energías renovables
		env_co2_percap	Emisiones co2 per cápita
		env_co2	Emisiones co2
	gobierno	gov_cor_rate	Ratio de corrupción en la gestión de gobierno
		gov_exp_pib	Inversión del PIB
		gov_exp_san	Inversión en sanidad sobre el PIB
		gov_exp_san_percap	Inversión en sanidad sobre el PIB, per cápita
		gov_exp_war	Inversión en fuerzas militares sobre el PIB
		gov_exp_war_percap	Inversión en fuerzas militares sobre el PIB, per cápita
		gov_exp_edu	Inversión en educación sobre el PIB, per cápita
		gov_exp_edu_percap	Inversión en educación sobre el PIB, per cápita
		gov_pre_PSOE	Fuerza de Gobierno en legislatura previa
		gov_pre_PP	
		gov_pre_UCD	

C. Histórico del voto por partido ampliado (figura 16):



D. Proceso SAS mediante nodos:

Proceso de selección de exploración de estadísticos, selección de variables e informes en SAS, mediante nodos:



E. Resultados numéricos de las predicciones para 2023:

Tabla con los resultados de ElPlural, nuestros modelos y el promedio de mercado para la próxima carrera de 2023. El promedio de mercado que hemos definido en este caso, es el promedio de las encuestas con mínimo una población de 30 encuestados y en un rango de como máximo 365 días previos al evento electoral.

party	ElPlural	prom_mercado	prediccion_de_partido_arbol	prediccion_de_partido_rf	prediccion_de_partido_gbm	prediccion_de_partido_red	prediccion_de_partido_svm
BNG	NA	0,78	0,78	2,86	0,36	2,35	2,73
CC.NC	NA	0,46	0,29	0,63	0,13	1,96	3,87
CS	1,8	1,94	7,46	1,18	7,17	5,58	0,05
CUP	NA	0,80	0,84	3,13	0,08	1,53	3,24
EH.BILDJ	NA	1,27	1,15	3,42	0,74	1,75	3,01
ERC	NA	2,88	3,00	5,20	2,83	2,96	2,19
EV	NA	0,16	0,38	2,59	0,30	0,78	3,65
JC	NA	1,85	1,80	4,30	1,77	2,57	2,44
MP	0,2	2,17	2,40	4,57	2,28	2,99	2,19
NS	NA	0,40	0,50	2,71	0,40	0,90	3,56
PNV	NA	1,60	1,59	3,75	1,30	0,25	3,65
PP	30,3	30,44	28,79	26,64	26,95	24,88	25,00
PRC	NA	0,18	0,33	2,53	0,43	0,64	3,74
PSOE	26,6	26,91	27,39	24,78	20,25	19,530	13,74
UP	10,7	10,83	8,55	7,61	10,16	6,10	4,88
VOX	15,3	14,31	18,46	19,35	15,75	21,11	20,90

**los valores de promedio de mercado se calculan con aquellas encuestas que tengan un espacio muestral de más de 30 personas y a menos de 365 días de las elecciones*