Universidad Católica Argentina –
Facultad de Ingeniería y Ciencias Agrarias

551 – TRABAJO FINAL

Título del Trabajo: Predicción de la confianza del productor agropecuario mediante text analytics

Alumno

Nombre: María Vera Rueda **Nro. Registro:** 151520194 **Carrera:** Ingeniería Industrial

Director

Nombre: Dr. Ing. Emilio Picasso

Fecha de entrega: 05/02/2021

Agradecimientos

Agradezco al Ing. Xavier Ignacio Gonzalez e Ing. Mariano Bonoli Escobar por su contribución con las bases de este trabajo, y al equipo de CREA por la facilitación de los datos y sus comentarios.

ÍNDICE

1. Resumen

El sector agropecuario y agroindustrial es el principal ganador de divisas de la Argentina. Este sector ha sido afectado por la inestabilidad política y económica hace décadas atrás, variando entre políticas de libre comercio y políticas proteccionistas. La agricultura es mayoritariamente un negocio privado, su desarrollo y crecimiento dependen de las estrategias individuales elegidas por los propios productores agropecuarios. Estas decisiones son determinadas en gran parte por la confianza que los productores tienen en relación con el contexto económico, financiero y sectorial.

Para cuantificar la confianza del productor, el movimiento CREA comenzó a medir el Índice de Confianza del Empresario Agricultor (ICEA) desde el 2012 a través de encuestas cuatrimestrales, que requieren grandes costos económicos y de tiempo. Este índice refleja los movimientos de la actividad económica y evalúa la disposición de crecimiento del sector. Sin embargo, sería beneficioso idear un mecanismo que permita obtener este índice de una manera más rápida y eficiente. Es por eso que proponemos una forma alternativa de medir la confianza del agricultor con herramientas de *Data Mining* y *Text Analytics* para extraer información proveniente de Twitter y desarrollando un modelo estadístico basado en esta información, que sea capaz de predecir el ICEA de forma inmediata y menos costosa.

El objetivo del presente trabajo es investigar la relación entre el ICEA y la opinión del público agricultor expresada en *tweets,* y a partir de eso crear un modelo estadístico que prediga los valores futuros del ICEA.

Las preguntas a responder son las siguientes:

¿Es posible estimar la confianza del productor agropecuario a través de lo que *twittean* usuarios interesados en el tema? ¿Qué información contenida en los *tweets* influencian significativamente en la confianza del productor agropecuario argentino? ¿A través de qué modelo estadístico se podría conseguir un buen ajuste de los datos?

La metodología que utilizamos para el desarrollo del trabajo se dividió en cuatro partes:

1 - Recolección de datos: Consistió en la identificación de usuarios relevantes a analizar, seleccionando aquellos usuarios que representen la opinión del sector agropecuario, y la extracción de los *tweets* publicados – entre 05-2014 y 07-2020 – por los usuarios seleccionados a través de interfaces de aplicaciones de programación con Python. Obtuvimos un total de 734.258 *tweets.*

2 - Procesamiento de datos: Los datos fueron procesados mediante la segmentación de *tweets,* normalización del texto, filtración por palabras clave de los *tweets* recolectados y *sentiment analysis,* análisis que permite cuantificar la actitud del autor expresada en un texto devolviendo un *score* (puntaje) mediante Google Cloud Natural Language.

3 - Construcción del índice de *sentiment*: Mediante el *score* obtenido en el paso anterior creamos un nuevo índice que replicara el mecanismo del índice tradicional. Así, obtuvimos el *Índice de Sentiment* cuatrimestral por usuario.

4 - Análisis para vincular el índice de *sentiment* con el índice de confianza del empresario agropecuario (ICEA) mediante un modelo predictivo: Para vincular ambos índices, creamos un modelo lineal múltiple de 12 variables, donde cada una representara uno de los subtemas[1] – resultantes de la clasificación de palabras clave – relevantes a la agricultura. Mediante herramientas estadísticas, eliminamos del modelo las variables que no representaran significativamente al estimador de la confianza y obtuvimos un modelo final de 3 variables explicativas. Al mismo, le agregamos componentes dinámicas[2] con el objetivo de que la variable explicada sea aún más precisa.

El modelo estadístico para predecir valores futuros del Índice de Confianza del Empresario Agricultor en base a los *sentiments* obtenidos fue:

$$\tilde{Y}_c = 53{,}18 - 322{,}72X_1 + 118{,}64X_3 + 261{,}78X_7 + 0{,}603Y_{c-1} + \eta_c \quad (1)$$

$$\eta_c = -0{,}46\eta_{c-1} - 0{,}677\eta_{c-2} \quad (2)$$

Donde,

$\tilde{Y}_c$**:** Predicción del ICEA del cuatrimestre c

X_1: Índice de *Sentiment* cuatrimestral en *tweets* referidos al clima

X_3: Índice de *Sentiment* cuatrimestral en *tweets* referidos a temas impositivos

X_7: Índice de *Sentiment* cuatrimestral en *tweets* referidos a la macroeconomía nacional

[1] De las 47 palabras clave hicimos una clasificación de estas dentro de los siguientes subtemas: clima, sanitario, impositivo, comercio internacional, regulaciones, agro, macroeconomía nacional, situación, política nacional, política provincial, economía, economía sectorial.

[2] Una componente dinámica es aquel coeficiente que vincula la predicción del ICEA actual con valores del ICEA de uno o más períodos pasados.

Y_{c-1}: ICEA del cuatrimestre c-1
η_c: Valor residual del cuatrimestre c
η_{c-1}: Valor residual del cuatrimestre c-1
η_{c-2}: Valor residual del cuatrimestre c-2

En el presente trabajo se logró encontrar satisfactoriamente la relación entre el ICEA y el *sentiment* representado en *tweets* relevantes a la agricultura. Esto se alcanzó a través de un modelo múltiple lineal combinado con un modelo dinámico. El modelo resultante consiguió un coeficiente de determinación de 0,89 (siendo 1 el valor de mejor ajuste), con lo cual se concluye que se logró un muy buen ajuste. El estimador de la confianza de aquellas personas relevantes en el sector agropecuario argentino se vio afectado significativamente por tres de las doce temáticas: clima, impositivo y macroeconomía nacional. Es importante destacar que el *sentiment* con respecto a temas sanitarios, comercio internacional, regulaciones, política nacional y provincial, economía y economía sectorial resultó no del todo relevante a la hora de estimar la actitud de las personas.

Como conclusión, la predicción de la confianza del productor agropecuario a través de *tweets* es posible y tiene una precisión aceptable. Comprobamos que ciertos *tweets* con ciertas palabras logran representar al ICEA más que otras. Como próximos pasos proponemos una refinación en la selección de datos, especialmente la realización de una previa investigación para la selección de palabras clave, y una ampliación del tamaño de muestra, es decir la ampliación del período de estudio, para poder conseguir un resultado aún más preciso.

2. Introducción

2.1 Sector Agropecuario en Argentina

El sector agropecuario y agroindustrial es la principal fuente de divisas de la Argentina, un importante generador de puestos de trabajo, y tiene una significativa participación en el valor agregado de la economía. Se entiende por agropecuario a toda actividad primaria que tenga vínculo con la agricultura y la ganadería. La agroindustria es la actividad económica por la cual se industrializan y comercializan los productos agropecuarios, forestales y biológicos. Es decir, transforma lo producido en el sector primario agregándoles valor. Esta industria es donde desemboca la actividad primaria agropecuaria y en conjunto forman unos de los principales pilares en la economía argentina.

Debido a la riqueza y extensión de los suelos argentinos, la actividad agropecuaria es tradicionalmente una de las principales actividades económicas en el país. En 2017, la Argentina produjo el 5% del total de granos del mundo (Calzada, J. et al., 2017). A nivel mundial, el país encabeza los primeros puestos en principales exportadores mundiales de soja, maíz, trigo, aceites (de oliva, maní, girasol y maíz), entre otros. En 2018 se registró que el 60% de las exportaciones de la Argentina las generó el sector agropecuario y agroindustrial, siendo el 40% del total parte del sector oleaginoso y cerealero (Calzada, J. et al., 2019).

Este sector ha sido afectado por la inestabilidad política y económica desde hace décadas atrás. Desde principios del siglo XX, las políticas agrícolas argentinas no han seguido una tendencia clara a lo largo de los años. Bajo diferentes marcos de política económica, el sector se ha enfrentado a los vaivenes en materia de política exterior, variando entre políticas de libre comercio y políticas proteccionista (sustitución de importaciones). De acuerdo con la OECD, la Argentina experimentó un auge en la agricultura desde mediados del siglo XIX hasta los años 30. Esta fue promovida por los distintos gobiernos de ese periodo que implementaron una economía abierta y estimularon la producción de materias primas, estas medidas impactaron positivamente al sector clasificando al país como “el granero del mundo”. Posteriormente, según el autor, desde los años 40 hasta los 70, el país adoptó una posición proteccionista para con la economía donde el sector agropecuario sufrió intervenciones en los precios, impuestos a la exportación y poca inversión por parte del gobierno. En 1970 hubo un intento de reabrir la economía, sin embargo, las políticas agrícolas seguían la misma línea de intervención de los años anteriores. En la última década del siglo XX, el país tuvo un giro económico donde la agricultura se vio favorecida, fue eliminada la fijación de precios, promovida nuevamente la exportación e importación que, junto con el avance tecnológico del momento, la agricultura se volvió a destacar. Este periodo fue seguido por un aislamiento económico

con aranceles más altos e impuestos a la exportación hasta el 2015, cuando asumió un nuevo gobierno y revirtió las políticas agrícolas anteriores al eliminar los impuestos a la exportación para la mayoría de los productos agrícolas y ganaderos. Esto dio lugar a cambios favorables en los precios relacionados y mayores ingresos para los productores de granos y carne (OECD, 2019).

Dado que la agricultura es mayoritariamente un negocio privado, su desarrollo y crecimiento depende de las estrategias individuales elegidas por los propios productores agropecuarios. Estas decisiones son determinadas en gran parte por la confianza que los productores tienen en relación con el contexto económico, financiero y sectorial. La volatilidad de políticas que tuvo el país a lo largo de su historia influyó al sector en gran medida y eso trajo como resultado una inestabilidad en la confianza del productor agropecuario. El presente trabajo intenta predecir esta confianza, lo que podría ser un punto de partida para medir la influencia y decisiones económicas de los productores.

2.2 Índice de Confianza

El índice de confianza del consumidor (ICC) es un indicador económico que mide el grado de optimismo o pesimismo que los consumidores sienten sobre la evolución del estado general de la economía y su situación financiera personal, haciendo referencia al comportamiento de los individuos frente a la economía. Qué tan segura está la persona sobre sus ingresos determinará su grado de consumo, por lo tanto, es un indicador clave para la economía.

El ICC se basa en la premisa de que, si el consumidor es optimista, consumirá y estimulará la economía, pero si es pesimista, la falta de confianza generará incertidumbre y sus patrones de consumo podrían desembocar en una recesión.

Las primerias teorías sobre confianza del consumidor se realizaron la década del 1940 como una idea contraria a la racionalidad de la teoría económica, se creía que el consumo estaba vinculado a los ingresos. Katona, G. (1951, 1953), establece que los agentes económicos se ven afectados por más aspectos que lo influencian a tomar decisiones. En sus primeros estudios define el *consumer sentiment* (sentimiento del consumidor) como una dimensión definida por dos fuerzas: la capacidad y la disposición a comprar. Por capacidad se refería a la cantidad de ingresos que tiene disponible el consumidor y por disposición a comprar a la percepción positiva o negativa frente sus futuros ingresos. Katona creía que el consumo no solo estaba ligado a los ingresos, sino también, a la certeza o incerteza vinculada a sus expectativas, el *consumer sentiment* lo definió en un rango de optimismo y confianza a pesimismo e incertidumbre.

Posteriormente, numerosos economistas siguieron la misma línea de estudio, surgieron teorías aún vigentes - como la teoría del comportamiento económico (Behavioral Economics Theory). Estas integran una visión psicológica y económica para estudiar los aspectos sociales, psicológicos y emocionales que influyen el comportamiento de los agentes económicos para predecir la toma de decisiones.

Actualmente existe una forma específica de medir el grado de optimismo del consumidor a través del Índice de Confianza del Consumidor (ICC). Este indicador económico contempla el sentimiento o grado de confianza de la persona frente a la economía general y a su situación financiera personal, el grado de este determina su comportamiento de consumo. En 2007, Curtin utiliza el test de causalidad de Granger en 37 países donde muestra en qué grado el ICC predicen y pueden ser predichos por una gran variedad de variables económicas, aunque estas varían según el país. Tanto el PBI, la tasa de desempleo, los ingresos personales y los bienes durables son parámetros que impactan fuertemente en la confianza del consumidor, por ejemplo, una disminución en el PBI o aumento de la tasa de desempleo disminuyen significativamente la confianza. El autor también muestra que el ICC afecta de forma análoga a estos parámetros. (Curtin, R., 2007).

El sistema más utilizado en la actualidad para obtener el ICC es a través de encuestas, el cual cuenta con algunas carencias. Primero que nada, demanda mucho tiempo y costos para conducir encuestas de muestras considerables. Segundo, las encuestas no se pueden realizar de manera continua, se hace una reducida cantidad de veces por año en lo posible, lo que no permite capturar la variación de confianza dentro de esos meses. Tercero, la encuesta se reduce a solo algunas preguntas, sin tener más información sobre otros aspectos que también podrían afectar al ICC. Por lo tanto, se asume que el ICC solo refleja lo que se pregunta en las encuestas.

2.2.1 Índice de Confianza del Empresario Agricultor

El Movimiento CREA (Asociación Argentina de Consorcios Regionales de Experimentación) está compuesto por aproximadamente 2000 miembros, comprendiendo todo tipo de actividad agropecuaria: agricultura, ganadería, lechería y cultivos perennes, distribuidos en todo el país; que representan la mayor parte de los productores agropecuarios del país. Desde 2012, comenzaron a medir el Índice de Confianza del Empresario Agricultor (ICEA) a través de encuestas. Las encuestas se realizan a sus miembros cuatrimestralmente en marzo, julio y noviembre. A partir de las mismas se obtienen datos de la sección “percepción y expectativas” dentro del Sistema de Encuestas Agropecuarias (SEA) realizada por CREA. La evolución de este índice permite reflejar movimientos cíclicos en la actividad económica a corto plazo.

El indicador ICEA cuantifica las opiniones de los empresarios agropecuarios sobre la economía nacional, sectorial y el contexto financiero y económico empresarial. Se obtiene a través de las encuestas mencionadas anteriormente, incluyen 6 preguntas, que se dividen en tres subíndices temáticos: situación económica general (pregunta 1 y 2), situación económica y financiera para la empresa (preguntas 3 y 4) y situación del sector agropecuario (preguntas 5 y 6):

Las preguntas referentes a la situación Económica General son:

1. ¿Cómo cree que se encuentra en relación con un año atrás?
2. ¿Cómo cree que se será dentro de un año?

Las preguntas referentes a la Situación Económica y Financiera de la Empresa:

3. ¿Cómo cree que se encuentra en relación con un año atrás?
4. ¿Cómo cree que será dentro de un año?

Las preguntas referentes a la Situación del Sector Agropecuario:

5. ¿Cómo cree que es el momento actual para realizar inversiones en su empresa? Por ejemplo: compra de maquinaria, vehículos, mejoras e instalaciones, compra de vientres, compra anticipada de insumos, entre otros.
6. ¿Cómo cree que será el nivel de precios de los productos agropecuarios dentro de un año en relación con el nivel actual? (Considere qué mayor incidencia tengan en el resultado de su empresa)

Se admiten tres respuestas: positiva, negativa y neutra, obteniendo así el índice parcial:

$$Indice\ Parcial = K \times (p - n + 1) \quad (1)$$

$K = 50$
$p = proporción\ de\ respuestas\ positivas\ sobre\ el\ total\ de\ encuestados$
$n = proporción\ de\ respuestas\ negativas\ sobre\ el\ total\ de\ encuestados$

El índice parcial se obtiene por cada pregunta, luego, se agrupan las preguntas según subíndice temático y se obtiene un índice promedio para cada uno, finalmente se hace un promedio general de las tres secciones temáticas, obteniendo así el Índice de Confianza del Empresario Agropecuario (ICEA) CREA:

$$ICEA\ CREA = \sum_{i=1}^{n} \frac{1}{n}\ Índices\ Parciales \tag{2}$$

$$ICEA\ CREA = \frac{1}{3}\ Índice\ del\ País + \frac{1}{3}\ Índice\ del\ Sector\ \frac{1}{3}\ Índice\ de\ la\ Empresa \tag{3}$$

Con respecto al tamaño muestral, el ICEA se calcula con una confianza del 95% y un margen de error del 5%. (Fusco, M., et al., 2017).

2.3 Big Data

Big data representa los activos de información caracterizados por un volumen, velocidad y variedad tan altos que requieren una tecnología específica y métodos analíticos para su transformación en valor.

El *Big Data* hace referencia a los conjuntos grandes y complejos de datos que, al ser tan voluminosos, los softwares de procesamiento de datos convencionales, simplemente no pueden administrarlos. Sin embargo, estos volúmenes masivos de datos pueden utilizarse para abordar problemas que antes no hubiera sido posible solucionar.

Big Data marcó el comienzo de una mayor transformación, a pesar de que desde los 1960s se comenzó a hablar de “revolución informática” o “era digital”, estos términos empezaron a volverse efectivamente realidad a finales del siglo XX e inicios del XXI. En el año 2000 se registró que un 25% de la información almacenada estaba en forma digital y en tan solo siete años, en 2007, se tenía 300 exabytes almacenados digitalmente, representando un 93% del total de los datos del mundo. Hoy en día, se estima que la información en formato digital duplica su cantidad a nivel mundial cada dos años, lo que nos obliga como sociedad a buscar las tecnologías necesarias para su mejor uso (Mayer-Schönberger, V. et al., 2013).

Uno de los principales desafíos de la sociedad es crear un marco normativo acerca de este nuevo concepto, donde los individuos sean conscientes de hacia dónde van los datos que nosotros mismos

generamos, dónde queremos que lleguen, cómo y con qué fines. La importancia de *Big Data* no radica en cuántos datos tenemos, sino en lo que podemos obtener de esos datos. El análisis de estos datos marca una gran diferencia en cuanto a reducción de costos, tiempo y calidad de las decisiones.

Se caracteriza por 3 dimensiones que convergen para definirlo:

1- **Volumen:** Hace referencia a la cantidad masiva de datos, ya que estos se están generando constantemente, y a las diversas fuentes provenientes. Esto es fundamental, debido a que aumenta considerablemente la precisión de los resultados.

2- **Velocidad:** No solo se generan muchos datos y desde muchas fuentes, sino que lo normal es que la velocidad en que se generan sea muy alta. Esto provoca un flujo de datos muy difícil de manejar con un software tradicional.

3- **Variedad:** Debido a las diversas fuentes de la que proviene la información, implica que el formato o estructura de los datos varíe mucho, por lo que se deben unificar los datos. Supone un nuevo desafío dependiendo de cada caso.

Normalmente, también se considera una cuarta dimensión:

4- **Veracidad:** Hace referencia al nivel de fiabilidad asociado a ciertos tipos de datos. La velocidad y el volumen de datos complican la verificación de su veracidad.

Como datos, entendemos a cualquier información relevante para el análisis que, por lo general, son datos digitales. Hay 2 tipos:

Estructurados: Forman parte de una estructura definida. Son datos fácilmente catalogables, por ejemplo: nombre, apellido, edad, etc.

No estructurados: Aquellos datos que no tienen ni forman parte de una estructura definida. Como, por ejemplo, datos escritos en un *email*, Microsoft Word, redes sociales, etc. Estos datos tienen mucha información valiosa, pero al no estar estructurada, se presenta una complicación a la hora de analizarlos. Sin embargo, hoy en día a través de algoritmos de *Machine Learning* se pueden analizar.

Los datos que manejamos en esta investigación son textos provenientes de *twitter*, por lo que se clasifican como datos no estructurados.

Dependiendo del tipo de dato en cada caso, se debe buscar la mejor herramienta para procesarlos. Teniendo los datos necesarios almacenados según diferentes tecnologías de almacenamiento, nos daremos cuenta de que necesitaremos diferentes técnicas de análisis de datos como las siguientes:

Analítica Descriptiva: Responde a la pregunta ¿Qué sucedió? Analizando el pasado con datos históricos. Simplifica y resume los datos para darnos una visión y contexto para entenderlos. Para ello, se utilizan herramientas como: *Business Intelligence*, Análisis Estadístico y *Data Mining*. Es uno de los más usados para una buena administración de una empresa u organización.

Analítica Predictiva: Responde a la pregunta ¿Qué podría pasar? Contrastando datos presentes e históricos, el objetivo es ofrecer escenarios y comportamientos futuros posibles, basados en probabilidades. La posibilidad que ocurra un evento en el futuro, la previsión o estimación de un parámetro, todo esto se realiza a través de modelos predictivos. Combina herramientas de: *Data Mining*, Modelos de *Machine Learning* y Estadística.

Analítica Prescriptiva: Responde a la pregunta ¿Qué deberíamos hacer?, además de prever futuros escenarios posibles, también sugiere decisiones a tomar frente a estos escenarios y evaluar el impacto de cada uno si es que se tomara. Se lleva a cabo con las siguientes herramientas: *Machine Learning*, Análisis de Decisión Multicriterio o Simulación.

En esta investigación utilizaremos la analítica predictiva. Con herramientas de *Data Mining* analizaremos los datos para que posteriormente se pueda realizar un modelo predictivo de la información que necesitamos.

Así como el *Big Data* se refiere a las grandes cantidades de información, para la recolección y el almacenamiento de esta, se necesita del *Data Mining.* Su función es identificar y extraer la información relevante de estos conjuntos de datos para luego aplicar diferentes técnicas y arrojar resultados específicos. El *Big Data* por sí solo no es suficiente, se necesita de estas herramientas especiales para gestionar esta información.

2.3.1 Data Mining

El *Data Mining* es el análisis de las grandes bases de datos (*Big Data)*. Es un conjunto de técnicas y tecnologías que permiten explorar grandes bases de datos provenientes de diferentes fuentes, de manera automática o semiautomática, con el objetivo de encontrar patrones repetitivos, modelos predictivos o estadísticas más precisas dentro de esa gran cantidad de datos. En pocas palabras, transforma la información en conocimiento. El *Data Mining* surge con el objetivo de otorgar una fácil interpretación de una enorme cantidad de datos y para que pudieran ser utilizados para contribuir en la toma de decisiones. Los datos son los medios para llegar a conclusiones mediante la elaboración de algoritmos capaces de procesar la información y cruzar los datos que interesan en cada momento.

SAS Institute (Suite of Analytics Software) define el concepto de *Data Mining* como el proceso de *Selecting* (Seleccionar), *Exploring* (Explorar), *Modifying* (Modificar), *Modeling* (Modelizar) y *Assessment* (Valorar) grandes cantidades de datos con el objetivo de descubrir patrones no conocidos anteriormente.

El *Data Mining* es utilizando en la materia de las industrial actuales, como por ejemplo:

Marketing: Sirve para mejorar la segmentación el mercado analizando las relaciones entre los parámetros característicos del cliente: género, gustos, edad, etc. Es posible predecir su comportamiento para así realizar campañas personalizadas de fidelización. También se puede predecir qué usuarios se pueden dar de baja a un servicio, que les interesa según sus búsquedas, etc.

Comercios: Para poder optimizar el trabajo de los empleados y los costos. A través de la identificación de horarios más concurridos en los locales optimizar tiempos, detectar ofertas valoradas por un cliente, estimar la demanda de productos, etc.

Banca: Se utiliza para entender los riesgos del mercado, detección de fraudes, movimientos de tarjetas, transacciones, patrones de compra y datos financieros de clientes.

Medicina: Favorece para otorgar diagnósticos más precisos. Al contar con toda la información del paciente se pueden prescribir tratamientos más efectivos. Posibilita una gestión más eficaz y eficiente para el uso de recursos sanitarios, predecir enfermedades en ciertos segmentos de la población, etc.

2.4 Text Analytics

Text Analytics hace referencia al descubrimiento de los conocimientos que se pueden encontrar en archivos de textos. Es una disciplina que tiene aplicaciones en muchas áreas, combinando técnicas del *Natural Language Processing* (NLP), *Data Mining* y *Machine Learning*. El *Text Analytics* está definido como el conjunto de técnicas lingüísticas, estadísticas, y de *Machine Learning* que modelan y estructuran la información contenida en un texto para análisis exploratorio de datos o investigación. Es el proceso por el que se traducen altos volúmenes de datos no estructurados, en forma de texto, en datos cuantitativos para desenmascarar percepciones, tendencias y patrones. Su objetivo es proveer un entendimiento completo de distintos tipos de textos como *emails*, textos de redes sociales, encuestas, reseñas, etc.

Dentro de esta disciplina, utilizaremos uno de sus métodos que es el *Sentiment Analysis.*

Sentiment Analysis

El *Sentiment Analyisis* en estudios o análisis de opinión pública se remonta a comienzos del siglo XX y el análisis de subjetividad de textos desarrollados por la comunidad lingüística computacional en los 1990s. Sin embargo, el estallido del uso de S*entiment Analysis* aparece en el siglo XXI con las publicaciones de Das, S. et al. (2001) y Tong, R. M. (2001), interesados en analizar el sentimiento con respecto al mercado. Subsecuentemente, Turney, P. D. (2002) y Pang, B., et al. (2002) aportan sobre el mecanismo del análisis en sí. Todas estas publicaciones desembocaron en la popularidad del *Sentiment Analysis* focalizados en el procesamiento del lenguaje natural. Entre el 2004 hasta el 2018 se registraron el 99% del total de las publicaciones referidas al *Sentiment Analysis* y sigue en continuo crecimiento (Mäntylä, M. V., et al., 2018).

En el *Sentiment Analysis* el objetivo es entender la opinión de la gente expresada en un idioma escrito (texto). Un aspecto importante de “entender” es reconocer el sentimiento o la emoción de lo que se dice. Como seres humanos, generalmente podemos notar si una oración es positiva o negativa, feliz o triste, pero preguntarle a una computadora que haga esto es una capacidad relativamente nueva. El *Sentiment Analysis* busca determinar la actitud que el autor del texto tiene hacia determinada persona u objeto, involucrando la experiencia, el conocimiento, la opinión y la emoción. Esta actitud ejerce una influencia clave en el comportamiento del individuo y por lo tanto de la sociedad.

Mediante el *Sentiment Analysis* buscamos entender cuál es la actitud exacta de una frase, posterior a eso queremos conocer qué valoración tiene dicha frase y para ello se le aplica la denominada polaridad, a

través de la cual se clasifica el mensaje en función de la actitud que tenga el autor al escribirlo, pudiendo ser éste positivo, neutro o negativo. Esto permite comprender la actitud de los usuarios hacia una temática, marca o producto, y traducirlo a una cantidad, es decir, cuantificar la actitud. Esto resulta sumamente útil para poder obtener datos concluyentes provenientes de textos y poder predecir el comportamiento.

El *Sentiment Analysis* procede automáticamente a partir de una serie de reglas lingüísticas que asignan una valencia positiva o negativa a ciertas palabras clave. Por ejemplo, dentro de las reseñas de un producto, mensajes que contengan “no me gusta”, “odio” o “no recomiendo” se clasificarán automáticamente cómo datos negativos. Mientras que, aquellos mensajes que incluyan un “me gusta”, “bueno” o “recomiendo”, quedarán clasificados cómo positivos. El análisis automático encuentra desafíos difíciles. Por ejemplo, cuando una palabra tiene diferente significado en diferentes ámbitos, como por ejemplo un nombre de marca que coincide con el de una ciudad, o también cuando las frases se escriben con sarcasmo. El análisis en este caso requiere la intervención del analista.

El proceso del análisis consiste en:

- Fraccionar el texto en distintas partes (oraciones, frases o tokens)
- Identificar el *sentiment* de cada token
- Asignar un score a cada token

2.5 Twitter

Twitter es el sistema de microblogging más popular entre otros servicios equivalentes como Tumblr, MySpace, Blog, etc., que permite enviar y recibir textos, llamados *tweets*. Los *tweets* son mensajes cortos, restringidos a 140 caracteres de longitud. Dada su naturaleza de servicio de microblogging (mensajes rápidos y cortos), la gente utiliza acrónimos, comete errores de ortografía, usa emoticones y otros caracteres que expresan distintos significados. Mencionan otros usuarios (con el símbolo @) o escriben hashtags para destacar temas (con el símbolo #), además un solo *tweet* contiene mucha información relacionada con el usuario: fecha de creación del *tweet*, ubicación, etc. Twitter permite seguir otros usuarios, lo que permite al seguidor ver los *tweets* del seguido en su página de inicio. Cada usuario elige seguir al usuario que le interesa o le parece relevante lo que publica. Twitter permite crear tus propios *tweets* o *retweetear* información publicada por otros, este último permite compartir información inmediata y efectivamente con los usuarios que lo siguen.

Esta red social se convirtió extremadamente popular entre académicos, estudiantes, políticos y el público en general. El acceso a la información en tiempo real y proveniente directamente de sus protagonistas, y el expresar la opinión sobre distintas temáticas sin ninguna restricción son una de las características que más valoran los usuarios de Twitter. Al ser una de las redes sociales de comunicación más populares mundialmente, es así una gran fuente de información que con el uso adecuado de herramientas de *Data Mining* se podría utilizar para diversas cosas. Dado que la mayor parte de sus usuarios la utilizan como medio para expresar su contento o descontento frente a situaciones económicas, políticas o sociales y dada su popularidad - utilizada por individuos influyentes, representantes de organizaciones y hasta incluso presidentes - esta fuente de datos será nuestra muestra representativa de la población para el estudio del presente trabajo.

2.6 Text Analytics y Twitter

Las redes sociales introducen una nueva forma para que el público general muestre su actitud frente a temas de interés general, de una forma distinta, que está abierta para todos. Estos últimos años, hemos presenciado el florecimiento de las redes sociales. Todos los días, enormes cantidades de contenido son generados en las redes sociales, especialmente en los servicios de microblogging. En Twitter, 500 millones de *tweets* son publicados por día (Sayce, D., 2019). Esto ha creado una revolución en la generación y distribución de contenido ya que se crea y comparte sin ningún tipo de control o limite. La era digital ha dado pasos agigantados en el crecimiento de las redes sociales, y al mismo tiempo ha creado una nueva oportunidad para que el público expresara su opinión sobre temas actuales y hacerlas de público conocimiento. Como resultado de esto, se creó una gran fuente de datos a través de los años y la forma de capitalizar estas fuentes ha sido un total desafío. El 80% de los datos digitales son no estructurados (Webster, J., 2012) y una gran parte incluyen datos provenientes de redes sociales. Esta información está dada en formato de texto escrito, donde la herramienta de *text analytics* resulta fundamental para su análisis.

El *sentiment analysis* es una herramienta para organizar textos no estructurados automáticamente, sus algoritmos permiten entender la polaridad de los *tweets* en tiempo real. Se analiza la opinión de las personas, evaluaciones, actitudes y emociones. Toda esa información recolectada a través del análisis de datos en redes sociales nos lleva al entendimiento de los sentimientos del individuo y su perspectiva frente a la situación económica de un país, política o incluso sobre un producto o servicio.

Las ventajas de realizar el análisis con esta herramienta son:

Análisis en tiempo real: Llevar un monitoreo constante y tener un análisis más detallado de los cambios de polaridad, detectando cambios repentinos en sentimientos, incremento de disgustos por parte de los individuos y para tomar acciones antes que los problemas avancen.

Escala: Permite analizar grandes cantidades de opiniones en tiempos reducidos, mientras que, si fuera manual, demandaría mucho más tiempo hacerlo.

Criterio Consistente: Se realiza un análisis de texto objetivo, utilizando un mismo criterio para todos. Al hacerlo manual se corre el riesgo de otorgarle subjetividad al análisis dependiendo de la persona que lo haga. Se le pude indicar al algoritmo qué criterio utilizar para el análisis y se realizara de la misma manera para todo texto.

2.7 Predicción

La predicción en estadística es el anuncio de lo que se espera que puede suceder. Es un elemento importante de las ciencias, en general, pues permiten hacer experimentos y contrastar el hecho esperado con la realidad.

El objetivo de las técnicas de predicción es obtener estimaciones o pronósticos de valores futuros a partir de datos empíricos. Estas técnicas no requieren la especificación de los factores que determinan el comportamiento de la variable, sino que se basan únicamente en la modelización del comportamiento sistemático de los datos.

2.8 Modelo Lineal

En muchos problemas existe una relación inherente entre dos o más variables, y resulta necesario explorar la naturaleza de esta relación. El modelo lineal permite representar la relación entre una variable continua y sus presuntos factores causales y predecir la variable dependiente.

Modelo lineal simple

El modelo lineal simple estudia el vínculo entre dos variables aleatorias que se denominan X = variable explicativa e Y = variable dependiente o de respuesta. En el modelo lineal simple se vincula una variable predictora con una de respuesta y propone que:

$$Y = \beta_0 + \beta_1 X + \varepsilon \quad (4)$$

Donde,

ε: Error
β_0: Ordenada al origen
β_1: Pendiente

La ecuación (4) indica que, para cada valor de X, la correspondiente observación Y consiste en el valor $\beta_0 + \beta_1 X$ sumándole una cantidad ε, que da lugar a relaciones no exactamente lineales. Los valores β_0 y β_1 son constantes que se denominan coeficientes de la ecuación. El modelo se denomina lineal porque propone que la Y dependa linealmente de X y se relacionen a través de los coeficientes de la ecuación. Otra forma de expresar la ecuación es para cada observación (X_i, Y_i):

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \quad (5)$$

Los parámetros de la ecuación se estiman a partir de los datos Xi y Yi mediante el **método de cuadrados mínimos**, que minimiza la suma de cuadrados de las diferencias en las ordenadas entre los puntos generados por la función elegida y los datos.

Las sumas de cuadrados para el análisis de varianza son:

- Suma de cuadrados total (SS_{Tot})

$$\sum_{i=1}^{n} (Y_i - \bar{Y})^2 \quad (6)$$

$$\sum_{i=1}^{n} Y_i^2 - n\bar{Y}^2 \quad (7)$$

- Suma de cuadrados de los residuos (SS_{Res})

$$\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{n} e_i^2 \tag{8}$$

- Suma de cuadrados de la regresión o del modelo (SS_{Reg})

$$\sum_{i=1}^{n}(\hat{Y}_i - \bar{Y})^2 \tag{9}$$

Siendo,

$\hat{Y}_i$: Valor predicho o ajustado i-ésimo
$\bar{Y}$: Promedio observado de la muestra
Y_i: Valor observado de la muestra

La SS_{Tot} compara cada valor observado con el promedio del modelo. La SS_{Res} compara cada valor observado con la predicción del modelo, que representa el residuo por observación – la cual se espera que sea mínima. Por último, la SS_{Reg} que compara la predicción de cada valor con el promedio del modelo, y nos muestra la reducción en la imprecisión que se obtiene por usar un modelo de regresión lineal.

Una vez estimada la función de regresión simple, ésta debe cumplir ciertos supuestos, debe ser evaluada la calidad del ajuste a través del coeficiente de determinación y debe ser verificada la significancia de los coeficientes para asegurarse de que se obtuvo así un modelo capaz de predecir futuras respuestas (con cierto error de precisión) mediante su función estimada.

Supuestos del modelo

Los supuestos bajo los cuales será válido el modelo (5) son:

1. Los ε, tienen media cero $E\ (\varepsilon_i) = 0$
2. Los ε_i tienen todos la misma varianza (desconocida) σ^2 , que es el otro parámetro del modelo, $Var\ (\varepsilon_i) = \sigma^2$. A este requisito se lo llama *homocedasticidad*

3. Los ε_i tienen distribución normal
4. Los ε_i son independientes entre sí, y no están correlacionados con las X_{i}

. Los cuatro supuestos mencionados anteriormente pueden resumirse en la siguiente expresión:

$$\varepsilon_{\mathrm{i}} \sim N(0, \sigma^2), \quad 1 \leq i \leq n, \qquad independientes\ entre\ si$$

Coeficiente de determinación

Una vez estimados los parámetros a partir de los valores observados, es importante evaluar la calidad del ajuste. El coeficiente de determinación (R^2) es un indicador de este:

$$R^2 = 1 - \frac{SS_{Res}}{SS_{Tot}} \tag{10}$$

Este coeficiente indica qué proporción de la variabilidad total de la variable Y puede ser explicada por la variable explicativa, en consecuencia, es una medida de la capacidad de predicción del modelo. Mide la fuerza de asociación lineal entre X e Y, y sus propiedades son las siguientes:

$$0 \leq R^2 \leq 1$$

- No depende de las unidades de medición
- Mientras mayor es R^2 , mayor es la fuerza de las variables explicativas para predecir la variable respuesta Y
- Mientras mayor sea R^2 menor es la SS_{Res} y, por lo tanto, más se ajusta el modelo a las observaciones obtenidas

Inferencia sobre β_1

Este coeficiente que representa la pendiente de la ecuación, indica en qué grado se relacionan las variables. Si β_1 es notablemente distinto de cero se concluye que hay un vínculo lineal entre X e Y. Para la inferencia se realiza un ensayo de hipótesis:

$$H_0)\ \beta_1 = 0 \qquad H_1)\ \beta_1 \neq 0$$

La distribución del estadístico es normal, pero al tener solamente sus parámetros muestrales, se infiere a través del estadístico t de student:

$$t_{obs} = \frac{\widehat{\beta_1}}{\sqrt{\widehat{Var}\ (\widehat{\beta_1})}} \tag{11}$$

El ensayo se rechaza si:

$$t_{obs} \leq t_{n-2;1-\frac{\alpha}{2}} \quad o \quad t_{obs} \geq t_{n-2;\frac{\alpha}{2}}$$

Siendo α el error de tipo I y n el tamaño de la muestra.

Modelo Lineal Múltiple

Es uno de los modelos estadísticos más utilizados. En muchos casos al intentar explicar una variable continua Y se dispone de muchas potenciales variables explicativas. Usualmente, una variable explicativa es insuficiente para explicar la variable respuesta. El modelo lineal múltiple es una extensión del modelo simple que incorpora varias variables explicativas simultáneamente.

La modelo lineal múltiple busca predecir la esperanza de una variable continua Y cuando se conocen las variables explicativas o predictoras que denotaremos $X_{1,}\ X_2\ , \ldots\ X_{p-1}$. El modelo se denomina lineal puesto que la esperanza de Y condicional a las $X's$ depende linealmente de las variables explicativas X_n. Los coeficientes de la ecuación se determinan a partir de n observaciones. El modelo es:

$$Y_i = \beta_0 + \beta_1 X_{1\mathrm{i}} + \beta_2 X_{2\mathrm{i}} + \cdots + \beta_{p-1} X_{ip-1} + \varepsilon_i \tag{12}$$

Donde $\beta_0, \beta_1, \beta_2, \ldots, \beta_{p-1}$ son parámetros desconocidos, $X_{i1,}\ X_{i2}\ , \ldots\ X_{ip-1}$ son los valores de las variables predictoras medidas en el i-ésimo individuo con $1 \leq i \leq n$, n siendo el tamaño de muestra, Y_i es la variable respuesta medida en el i-ésimo individuo observado y ε_i es el error para el individuo i-ésimo, que no es observable.

En el caso del modelo lineal múltiple, se deben cumplir los mismos supuestos mencionados para la regresión simple. Donde la perturbación ε_i es una variable totalmente aleatoria:

$$\varepsilon_{\mathrm{i}} \sim N(0, \sigma^2), \quad 1 \leq i \leq n, \qquad independientes\ entre\ sí$$

Coeficiente de Determinación Múltiple (R^2 y R^2_{adj})

El coeficiente de determinación se extiende naturalmente al caso del modelo lineal múltiple, representando la capacidad predictiva del conjunto de variables explicativas. Se demuestra que el coeficiente de determinación no puede disminuir cuando se agregan variables explicativas al modelo, independientemente de su capacidad predictiva. Se define entonces el coeficiente de determinación ajustado para resolver este inconveniente:

$$R^2_{adj} = 1 - \frac{(n-1)}{(n-p)} \frac{SS_{Res}}{SS_{Tot}} \tag{13}$$

Este coeficiente puede disminuir si se agrega una variable explicativa al modelo, ya que cualquier disminución de la SS_{Res} puede ser más que compensada por la pérdida de un grado de libertad en el denominador $n - p$. Si al agregar una variable el R^2_{adj} aumenta, quiere decir que esta nueva variable contribuye a la predicción de Y y si en cambio el coeficiente disminuye o no aumenta al incorporar esta nueva variable, es una señal de que no contribuye suficientemente para explicar Y.

Inferencias sobre los β_{k}

La inferencia para los coeficientes de la ecuación se realiza de la misma forma que en una regresión lineal simple. Se realizan ensayos de hipótesis para cada coeficiente y su rechazo implica una relación lineal entre Y y X_k.

Colinealidad de las variables explicativas

La colinealidad entre las variables explicativas es una propiedad indeseable que desestabiliza la estimación de los coeficientes y las predicciones del modelo. La presencia de combinaciones lineales de variables explicativas aproximadamente nulas genera este problema que tiene las siguientes consecuencias:

1- No se puede identificar de forma precisa el efecto individual que tiene cada una de las variables colineales X sobre la variable respuesta Y.
2- Los coeficientes de regresión estimados se modifican sustancialmente cuando se agregan o se quitan variables del modelo.
3- Los errores estándares de los estimadores de los coeficientes aumentan significantemente cuando se incluyen covariables muy correlacionadas en el modelo, es decir se infla la varianza estimada de los estimadores.
4- Los coeficientes pueden ser no significativos aun cuando exista una asociación verdadera entre la variable respuesta y el conjunto de variables explicativas.

El diagnóstico de la colinealidad se realiza de las siguientes maneras:

- Si el coeficiente de determinación R^2 es alto, pero ninguno de los predictores resulta significativo, hay indicios de colinealidad.

- La matriz de correlación en la que se estudia la relación lineal entre cada par de predictores se puede analizar, en el caso que el coeficiente de correlación entre dos variables se acerca a 1, se puede deducir que están correlacionadas.

- Generar un modelo de regresión lineal simple entre cada una de las variables explicativas frente al resto. Si en alguno de los modelos el coeficiente de determinación R^2 es alto, estaría señalando a una posible colinealidad.

- Factor de Inflación de la Varianza (VIF). Indica en qué grado la colinealidad afecta a la varianza de la variable.

Un método para lidiar con la colinealidad es la regularización, que consiste en controlar la magnitud de los coeficientes del modelo evitando que se desestabilicen. Las dos variantes principales de regularización del modelo lineal son: Ridge y LASSO.

LASSO (Least Absolute Shrinkage and Selection Operator)

Se complementa el objetivo de cuadrados mínimos con un término de penalidad para los coeficientes:

$$mín_{\alpha,\beta}\ \{\ \left|\left|Y - X\beta\right|\right|_2^2 +\ \lambda||\beta||_1\}$$

Siendo λ un parámetro de regulación o de penalidad que limita el tamaño de los coeficientes.

Dadas las características de la norma L1 que se utiliza para medir la longitud del vector de coeficientes, la optimización conduce a la anulación de algunos coeficientes a medida que la penalidad crece. De este modo el método conduce a una selección de variables de modo sistemático, apto para el aprendizaje automático. La determinación de la penalidad se realiza mediante cross-validation.

Modelo dinámico

Cuando los datos evolucionan a lo largo del tiempo es muy probable que falle el supuesto de independencia del modelo lineal, ya que datos próximos en el tiempo tienden a estar relacionados entre sí. El modelo dinámico introduce una representación de la dependencia temporal en el modelo lineal para dar cuenta de este fenómeno.

La formulación general del modelo dinámico es la siguiente:

$$\begin{cases}\tilde{y}_t = \beta_0 + \beta_1 x_{1t} + \beta_2 x_{2t} + \ldots + \beta_{p-1} x_{p-1,t} + \tilde{u}_t \\ u_t = \phi(u_{t-k}) + \theta(\varepsilon_{t-k}) + \varepsilon_t\end{cases} \quad (14)$$

Donde el residuo u_t tiene dependencia temporal representada por la segunda expresión en función de sus valores pasados (parte autorregresiva) o de las perturbaciones aleatorias pasadas (parte de media móvil).

Nosotros trabajaremos únicamente con el modelo autorregresivo, cuya expresión explícita es:

$$u_t = \varepsilon_t + \phi_1 u_{t-1} \ldots + \phi_p u_{t-p} \quad (15)$$

Se asume que el proceso u_t es estacionario, lo cual, bajo el supuesto de normalidad, equivale a establecer la constancia de la media, la varianza y la función de autocovarianza (covarianza de la variable consigo misma en el pasado).

La estimación de los parámetros del modelo se realiza mediante el **método de máxima verosimilitud** (Peña, 2010).

3. Objetivo

El objetivo del presente trabajo es investigar la relación entre el ICEA (Índice de Confianza del Empresario Agricultor) y la opinión del público agricultor expresada en *tweets,* para luego crear un modelo estadístico que prediga los valores futuros del ICEA. Esto permitirá anticipar el movimiento del ICEA entre mediciones cuatrimestrales, y eventualmente reemplazarlo.

Diversos aspectos del entorno influyen en la confianza del productor y esto repercute en las decisiones económicas del mismo. CREA (Asociación Argentina de Consorcios Regionales de Experimentación Agrícola), a través del ICEA, busca medir cuantitativamente esta confianza llevando a cabo encuestas cuatrimestrales a los miembros de la asociación. Esto demanda tiempo y dinero, y a su vez, depende de la disponibilidad de las personas encuestadas. Nuestro objetivo es desarrollar una herramienta que permita extraer la información que se obtendría de este índice de confianza en forma remota, en tiempo real y sin altos costos.

Hoy en día, las personas se están involucrando cada vez más al mundo de las redes sociales. Por ello, optamos por Twitter como nuestra fuente de datos para basar el modelo. Esta red social es una gran fuente de la opinión pública que posibilita capturar una opinión en forma remota, más inmediata y de manera constante. A diferencia de las encuestas ICEA, cuyos resultados se obtienen cada cuatro meses, nuestra metodología permite hacerlo con mayor periodicidad y a un menor costo.

Con herramientas de *Data Mining* rescataremos *tweets* pasados dentro del periodo de mayo del 2014 hasta julio del 2020 y mediante *Text Analytics* transformaremos estos datos no estructurados en un nuevo índice que, utilizando los valores de ese período del ICEA, creará un modelo estadístico que ajuste los valores del nuevo índice con los del índice original. Una vez encontrado este modelo, y sujeto a su bondad

de ajuste, podremos potencialmente desarrollar una herramienta que nos permita predecir valores futuros del ICEA de manera constante mediante este modelo predictivo.

Las preguntas a responder son las siguientes:

¿Es posible estimar la confianza del productor agropecuario a través de lo que *twittean* usuarios interesados en el tema? ¿A través de qué modelo estadístico se podría conseguir un buen ajuste de los datos? ¿Qué información contenida en los *tweets* influencian significativamente en la confianza del productor agropecuario argentino?

4. Metodología

La metodología que utilizamos para el desarrollo del trabajo se divide en cuatro partes: la recolección de datos, el procesamiento de los mismos, la construcción del índice de *sentiment* y el análisis para vincular el índice de *sentiment* con el Índice de Confianza del Empresario Agropecuario (ICEA) mediante un modelo predictivo.

4.1 Recolección de datos

Para armar nuestra base de datos recurrimos a Twitter seleccionando aquellos *tweets* que consideramos relevantes dentro del periodo de tiempo optado. Desarrollamos un método de identificación de usuarios relevantes en el rubro agropecuario y extrajimos sus respectivos *tweets*. Estos dos pasos los realizamos a través de la API (interfaz de programación de aplicaciones) oficial de Twitter y un complemento de esta.

4.1.1 Identificación de usuarios relevantes a analizar

El primer paso para identificar los distintos usuarios de Twitter que consideramos relevantes para el análisis fue partir de los seguidores de CREA (@crea_arg). De un total de 35 mil seguidores notamos que no todos formaban parte del rubro agropecuario. Es por eso que, para obtener una selección más precisa de los usuarios que nos interesan, elegimos otras cuatro entidades relevantes que tuvieran usuario en Twitter para analizar sus seguidores. Tuvimos en cuenta aquellas entidades que publicaran información sobre temáticas agropecuarias y que tuvieran una considerable cantidad de seguidores. Seleccionamos las siguientes:

Agro Sitio (@agrositio): Canal de información y negocios para el campo y la cadena agroindustrial. Cuenta con 33 mil seguidores.

Info Campo (@infocampoweb): Canal de información sobre el campo, con 38 mil seguidores.

Instituto Nacional de Tecnología Agropecuaria (@intaargentina): Organismo estatal descentralizado con autarquía operativa y financiera, desarrolla acciones de investigación e innovación tecnológica en las cadenas de valor, regiones y territorios para mejorar la competitividad y el desarrollo rural sustentable del país. 68 mil seguidores.

LA NACIÓN Campo (@LNcampo): Sección del diario argentino LA NACIÓN enfocada en el campo, con 28 mil seguidores.

A partir de esta selección de usuarios, extrajimos todos los seguidores de estos mediante Python, con la API pública de Twitter, Tweepy.

Tweepy es una librería de Python para acceder a la API de Twitter, la cual tiene distintas funcionalidades como leer y escribir información relacionada con entidades de Twitter como *tweets*, usuarios y tendencias. Provee una forma de invocar a estas entidades como *endpoints* HTTP sin tener que manejar demasiado detalle. La API de Twitter utiliza OAuth, un protocolo de autorización abierto, para autenticar todas las solicitudes. Antes de llamar a la API de Twitter, necesitamos crear y configurar las credenciales de autenticación. La autenticación es un proceso por el cual nos registramos a la API justificando la solicitud de acceso a los datos de Twitter, detallando la investigación y el motivo de acceso a la información contenida en esta red social. La solicitud debió ser analizada y aprobada para poder tener acceso a la aplicación.

El uso de esta API comprende varias funcionalidades, dentro de las cuales utilizamos la función de extraer los *ids* de los seguidores de las cuentas mencionadas anteriormente.

Una vez obtenidos los *ids* de los seguidores de las cinco cuentas, reducimos la selección a los seguidores comunes a todas estas. Es decir, aquellos usuarios que siguieran simultáneamente a las cinco cuentas relevantes serían los que analizaríamos. El área gris de intersección de la figura 1 representa la resultante de este filtro.

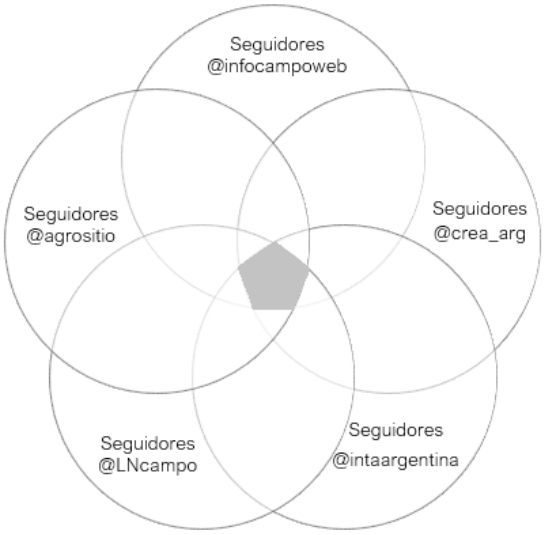

Figura 1 Área de selección de seguidores

4.1.2 Extracción de tweets

Luego de la selección descripta anteriormente, listamos los usuarios resultantes con su respectivo *id* y nombre de usuario para así comenzar la extracción de *tweets*. Aunque Tweepy tenga la funcionalidad de extraer *tweets*, no fue posible utilizarla en este paso dado que se limita a extraer aquellos cuya máxima antigüedad sea de 7 días con respecto a la fecha actual. El periodo que optamos para la extracción fue desde mayo de 2014 hasta julio del 2020, lo cual imposibilitó el uso de la API oficial de Twitter. Para realizar la extracción sin límite de antigüedad se encuentran numerosas herramientas, dentro de ellas, Get Old Tweets.

Get Old Tweets es el mejor método para extraer *tweets* pasados de forma gratuita. Consiste en un proyecto programado en Python para la obtención de *tweets* históricos. Fue creado por Jefferson Henrique y mejorado por Dmitry Mottl obteniendo como resultante GetOldTweets3, compatible con la última versión de Python. No ofrece ninguna otra funcionalidad que tenga Tweepy más que la obtención de *tweets,* a diferencia que Get Old Tweets permite hacerlo sin límite de antigüedad y cantidad. Esta librería devuelve la siguiente información:

Tweet: Clase modelo que permite obtener una descripción específica del tweet

- id (str)

- permalink (str)
- username (str)
- to (str)
- text (str)
- date (datetime) in UTC
- retweets (int)
- favorites (int)
- mentions (str)
- hashtags (str)
- geo (str)

TweetManager: Una clase que ayuda a obtener tweets del modelo Tweet:

- getTweets (TwitterCriteria): Devuelve la lista de los tweets extraídos, usando TwitterCriteria

TwitterCriteria: Colección de parámetros de búsqueda que se usan con TweetManager

- setUsername (str or iterable): Especifica un usuario propio de Twitter (con o sin @).
- setSince (str. "yyyy-mm-dd"): Una fecha mínima para restringir la búsqueda.
- setUntil (str. "yyyy-mm-dd"): Una fecha máxima para restringir la búsqueda.
- setQuerySearch (str): Filtración de tweets dado un texto de búsqueda.
- setTopTweets (bool): Si es True, solo los Top Tweets serán extraídos.
- setNear(str): Una referencia de una ubicación desde donde los tweets debieron ser generados.
- setWithin (str): Un radio de distancia que debe cumplir la ubicación.
- setMaxTweets (int): La máxima cantidad de tweets extraídos. Si no está especificado se extraerá el máximo posible.

Las anteriores funcionalidades que se enumeran son comandos que ofrece la librería. Estos devuelven datos relevantes de los *tweets* que se extraen, por cada comando tenemos una instrucción que se le indica al programa y devuelve información en forma de las variables que se indican entre paréntesis por cada comando. Los tipos de variables que devuelve son:

String (str): cadena de caracteres, devuelve un texto en formato *string*.

Dato lógico (bool): Representan valores de lógica binaria, devuelve 1 o 0 que normalmente hacen referencia a verdadero o falso.
Entero (int): Devuelve un subconjunto finito de números enteros.
Fecha (datetime): Devuelve un *string* en formato de fecha.

Extrajimos cada uno de los *tweets* publicados por usuario perteneciente a la lista de usuarios relevantes y que hayan sido publicados desde el 01/05/2014 hasta el 31/07/2020. Esto resulto en una cantidad de 734.257 *tweets* totales. Este proceso tomó aproximadamente 6 semanas, obteniendo un archivo en formato CSV por cada usuario con sus respectivos *tweets*. En el gráfico (1) se puede observar la cantidad publicada por mes:

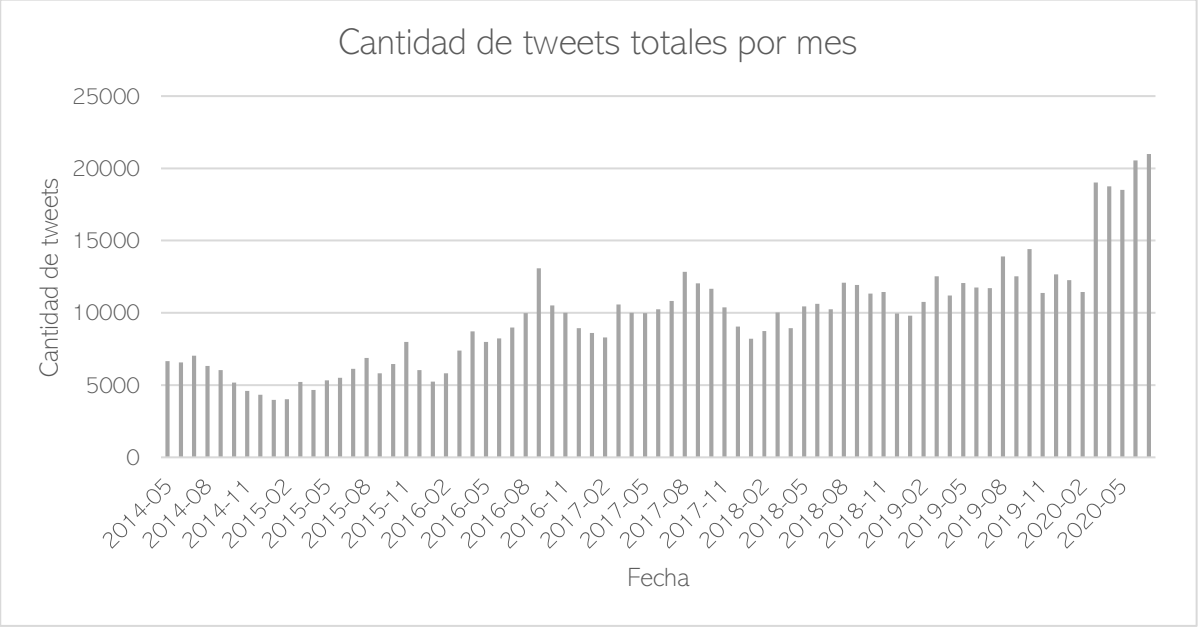

Gráfico 1 Cantidad de tweets totales por mes

4.2 Procesamiento de datos

Para procesar los datos, procedimos a crear un *data frame* (estructuras de datos de dos dimensiones que pueden contener datos de distintos tipos) que recopilaba todos los archivos individuales con los *tweets* de cada usuario, para organizar la información y que permita una fácil interpretación en los pasos procedentes: segmentación de *tweets*, que lista los textos separándolos palabra por palabra para un mejor análisis de estos, filtración de tweets, que filtra aquellos *tweets* que contuvieran palabras clave y fueran

relevantes para analizar y por último *sentiment analysis*, que analiza la actitud de aquellos *tweets* filtrados e indica el grado de optimismo del autor con respecto a lo que expresa.

4.2.1 Segmentación de tweets

La segmentación de *tweets* permite el análisis de texto para facilitar la manipulación de las palabras que contiene. Para segmentar esta información, creamos una función utilizando una herramienta de la librería NLTK.

NLTK (Natural Language Tool Kit) es una librería de Python creada en 2001, que tiene como objetivos poner en práctica los conocimientos de procesamiento de *Natural Language* sin tener que entrar en trabajos tediosos, proveer un contexto y estructurar la información para que sea fácil de interpretar. Ofrece componentes que puedan ser utilizados independientemente sin necesidad de entender el resto del *tool kit*. Es una fuente abierta y gratuita y contiene las siguientes funcionalidades:

Tarea a realizar	Módulos	Funcionalidad
Acceso a corpora	corpus	Interfaces estandarizadas a corpora y lexicons
Procesamiento de strings	tokenize, stem	tokenizadores, tokenizadores de oraciones, stemmers
Descubrimiento de colocaciones	collocations	t-test, chi-cuadrado, point-wise mutual information
Etiquetado de partes del texto	tag	n-gram, backoff, Brill, HMM, TnT
Machine Learning	classify, cluster, tbl	Arbol de decisiones, entropía Máxima, Näive Bayes, EM, k-media
Fragmentar	chunk	expresión regular, n-gram, named-entity
Analizado	parse, ccg	gráfico, feature-based, unificacion, probabilistica, dependencia
Interpretación semántica	sem, inference	cálculo lambda, lógica de primer orden, chequeo de modelo

Tarea a realizar	Módulos	Funcionalidad
Evaluación métrica	metrics	precisión, recall, coeficientes de convenio
Probabilidad y estimación	probability	distribución de frecuencia, distribución de probabilidades suavizadas
Aplicaciones	app, chat	concordancia de gráficos, analizadores, navegador WordNet, chatbots
Trabajo de campo lingüístico	toolbox	manipular información en formato SIL toolbox

El módulo que utilizamos en el trabajo fue el de *tokenize*. La función de *tokenizar* es uno de los pasos más básicos, y uno de los más importantes en el análisis de texto. Su objetivo es separar un texto entre unidades más pequeñas llamadas *tokens*, usualmente palabras o frases. Así fue como creamos una función que desglosa el *tweet* en una lista de palabras para que luego al analizarlo, se pueda acceder a cada una de las palabras individualmente.

4.2.2 Normalización del texto

La normalización de texto es el proceso de transformación de texto para la obtención de una forma canónica. Usualmente incluye la uniformización en el uso de mayúsculas, acentos, signos de puntuación, acrónimos y abreviaciones como también la corrección de errores ortográficos o de tipeo. Este es un paso especialmente importante a la hora de procesar el texto proveniente de usuarios en redes sociales debido al alto nivel de ruido. La reducción de este ruido permite mejorar la calidad de la información para su posterior procesamiento. En este trabajo nos enfocamos en el proceso de convertir caracteres de mayúscula a minúscula y quitar las tildes de las palabras. La necesidad de esto se debe a que en programación para cada variedad de carácter se le asigna un único código, por ejemplo, las letras a – A – á tienen cada una un código asignado distinto, a pesar de que para nosotros los tres caracteres son una misma letra. El objetivo de normalizar es indicarle al programa que esas tres variantes las interprete como una misma.

Para esto utilizamos dos herramientas:

Eliminación de tildes

Unicode es un estándar de codificación de caracteres diseñado para la transmisión y visualización de texto de numerosos idiomas y disciplinas técnicas. En el idioma español manipulamos caracteres específicos de la lengua que son las letras con tildes (á,é,í,ó.ú) y usualmente en la escritura de textos informales como los *tweets* suelen omitirse los tildes. Para poder tener un análisis del texto uniforme, utilizamos el módulo de Python *Unidecode* para canonizar todas las palabras quitándole las tildes. Este módulo lo que hace es representar la codificación *Unicode* en caracteres *ASCII* (corresponde a la expresión inglesa American Standard Code for Information Interchange), es decir, traduce el carácter original al más cercano del teclado de Estados Unidos. Nosotros lo utilizamos para eliminar las tildes. La función **unidecode()** nos devuelve:

Unicode	ASCII
á	a
é	e
í	i
ó	o
ú	u
Á	A
É	E
Í	I
Ó	O
Ú	U

Tabla 1. Equivalente de Unicode a ASCII

Con esto conseguimos tener un criterio uniforme para el análisis de palabras. Nos fue sumamente útil para la identificación de palabras claves dentro de los *tweets*, ya que independientemente de si el autor escribió la palabra con tilde o no, el programa la interpretará como una misma palabra.

Conversión de mayúscula a minúscula

Otra operación fundamental para unificar el criterio de análisis de palabras fue convertir las letras mayúsculas a minúsculas. Esto nos permitió que, a la hora de analizar una palabra, independiente de si el autor utilizo mayúsculas o minúsculas para escribirla, el programa la interpreta como una misma. Para esto

utilizamos la función **lower()** que devuelve una copia del *string* con todas sus letras en minúscula. Es una función propia de Python que se le asigna a los *strings*.

4.2.3 Filtración de tweets

Una vez obtenida la base madre conteniendo todos los *tweets* publicados por los usuarios relevantes, realizamos un segundo filtrado por palabras clave. El objetivo de este paso es que si el *tweet* fue publicado por un usuario dentro de la selección de cuentas relevantes y además contenía al menos una de las palabras clave, consideramos el *tweet* efectivamente relevante para nuestro trabajo. Las *keywords* (palabras clave) fueron las siguientes:

gobierno	retenciones	precio
ministerio	impuesto	precios
presidente	impuestos	commodity
presidencia	regulacion	comodity
ministro	regulaciones	commodities
ministra	clima	comodities
inflacion	inundacion	costos
política	inundaciones	dolar
economia	lluvia	dolares
economica	lluvias	tasa
economicas	sequia	tasas
gobernador	sequias	tributario
gobernadora	sanitario	tributarios
gobernacion	plagas	tributaria
situacion	agroquimico	trubitarias
retencion	agroquimicos	

Invocamos la función mencionada anteriormente que *tokeniza* los *tweets,* devolviendo una variable lógica (1 o 0) que hace referencia a TRUE o FALSE. Siendo verdadero cuando el *tweet* contenía alguna de estas palabras y falso cuando no las contenía. Creamos una nueva columna en el *data frame* que indicaba esta información.

4.2.4 Sentiment Analysis

El último paso del procesamiento de *tweets* es el paso troncal de la investigación. El objetivo fue traducir la opinión expresada en los *tweets,* anteriormente seleccionados, a un número. Fue necesario cuantificar la actitud del púbico agropecuario para ver en qué posición se encuentran con respecto a la situación económica y agropecuaria del país y personal. Esta información cuantificada nos sirvió para construir el índice de *sentiment*.

El algoritmo que determina el *sentiment* de cada texto es producto de un desarrollo complejo con alta involucración de expertos en Lingüística y Aprendizaje Automático. Varias compañías tienen equipos de desarrollo trabajando en este tema y ofrecen sus algoritmos:

- Google Cloud NLP

- IBM Watson

- Lexalytics

- MeaningCloud

- Amazon Comprehend

- Aylien

De estas plataformas elegimos Google Cloud que ofrece una variedad de APIs. La de nuestro interés fue la API de Natural Language para obtener el *sentiment* de los *tweets.*

Google Cloud

El término *cloud* hace referencia a un conjunto de servicios que se proveen a través de una red. Permite al desarrollador enforcarse en el proyecto en sí, en vez de la infraestructura necesaria tanto de hardware como software. Hace que el foco deje de estar en CPUs y RAM y se dirija a APIs, para operaciones de mayor nivel como almacenar y gestionar datos. La plataforma Google Cloud es una colección de productos que permite que el mundo use parte de la infraestructura interna de Google, incluye varios

servicios tales como APIs y tecnologías avanzadas propias de Google como Bigtable, Cloud Datastore o Kubernetes. Esta plataforma se destaca en cuanto a que opera a tal escala que le permite tener gran ventaja económica y en velocidad.

Google Cloud Natural Language es una de las API's dentro de Google Cloud. El proceso de *Natural Language* consiste en enviar un *input* (en este caso un texto) y la aplicación devuelve una serie de anotaciones sobre dicho texto.

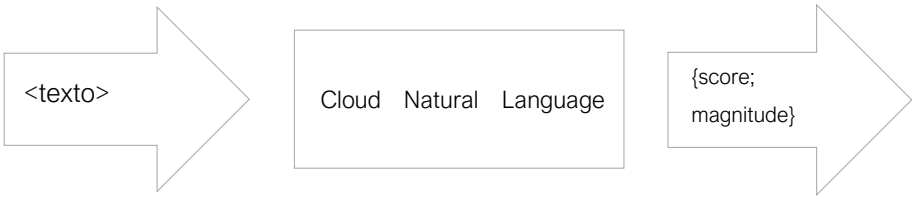

Figura 2 Flujo de funcionamiento de la API Natural Language para el sentimiento

La API NL tiene distintos métodos de analizar y obtener anotaciones de un texto, cada nivel de análisis proporciona información útil para la comprensión del texto, como los que se mencionan a continuación:

Análisis sintáctico: Extrae *tokens* y oraciones, identifica categorías gramaticales y crea árboles de análisis de dependencias de cada frase.

Análisis de *sentiment* de entidades: Busca entidades conocidas en el texto dado, muestra información sobre esas entidades e identifica el *sentiment* predominante hacia la entidad en el texto (positiva, negativa o neutra).

Análisis de entidades: Identifica entidades del dominio público en los documentos y los etiqueta por tipos como fecha, persona, organización, ubicación, evento, producto o medio.

Análisis de *sentiment*: Uno de los aspectos más interesantes del API es su habilidad para entender contenidos emocionales en oraciones y reconocer si esta expresa positividad, negatividad o neutralidad y en qué cantidad.

Clasificación de contenido: Analiza el contenido del texto y muestra una categoría para el mismo

El método que utilizamos para analizar el texto fue el de análisis de *sentiment*. El algoritmo de Google informa no solo la valencia del *sentiment* o score (positiva o negativa) sino también su magnitud. Combinando esas dos variables se alcanza una mejor comprensión del *sentiment:*

Oración	Score	Magnitud
"La casa es soñada"	Positivo	Alta
"La casa es fea"	Negativo	Alta
"La casa es linda. Queda en un barrio inseguro"	Neutro	Alta
"Esto es una casa"	Neutro	Baja

Tabla 2. Interpretación de sentiment

La *magnitude* indica la intensidad general de una emoción (tanto positiva como negativa). Para interpretarla de una forma más gráfica, en la figura 3 se pone como ejemplo un texto compuesto por dos oraciones, cada una con un *sentiment* distinto (uno positivo y otro negativo), representadas por vectores. Para obtener la *magnitude* del texto completo se suman los vectores, el módulo del vector resultante a la suma indica la *magnitude* general del texto, mientras que la dirección y el sentido lo indica el *sentiment*:

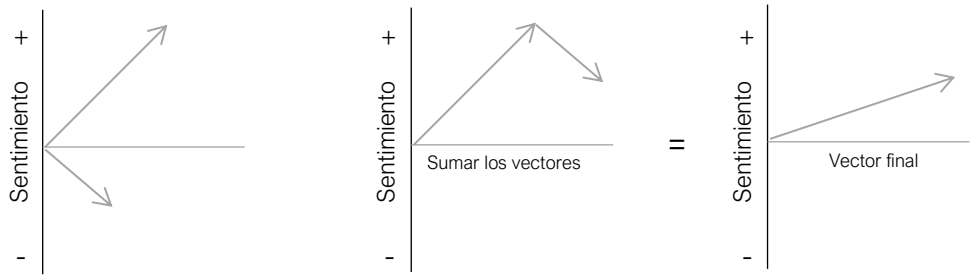

Figura 3 Combinación de múltiples vectores de sentimiento

La API de *Natural Language* devuelve valores denominados *score* y *magnitude*. Siendo el ***score*** un numero entre -1 y 1 que van de muy negativo a muy positivo, como se indica en la figura 4. La ***magnitude*** va desde 0 a $+\infty$, donde cero indica que la oración fue realmente neutra y un valor mayor indica proporcionalmente con cuanta intensidad expresó el sentimiento (independiente de su negatividad o

positividad). La oración es la mínima unidad que se puede analizar, por lo que cuando se presenta una sola oración, el *score* y *magnitude* van a ser equivalentes. Cuando se presenta un texto con *score* resultante neutro, la *magnitude* ayuda a indicar si es una única oración efectivamente neutra o es parte de una composición de múltiples oraciones positivas y negativas que se contrarrestaron entre ellas y resultó siendo neutra.

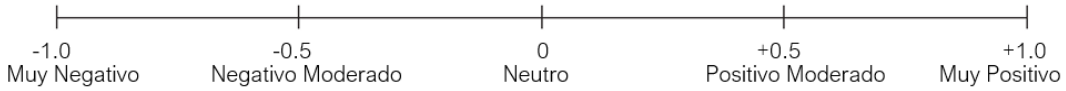

Figura 4 Escala del score

4.3 Construcción del índice de *sentiment*

Una vez obtenido el *sentiment* para cada *tweet* que consideramos relevante, denominado Índice de *Sentiment* (IS), buscamos la forma en que estos valores predigan lo más precisamente posible a los valores del Índice de Confianza del Empresario Agricultor (ICEA). Para esto, intentamos adaptar los datos de forma tal que el nuevo índice replique el mecanismo del índice tradicional, a través de los siguientes pasos:

1- Promediamos el IS de cada usuario por mes para tener un valor representativo del *sentiment* medio de la persona en ese periodo. Por ejemplo, si una persona tiene *sentiments* negativos y positivos a lo largo del mes su promedio dará neutro. Consideramos que es el mismo razonamiento de la persona real cuando debe inferir sobre sus sentimientos a lo largo de un mes:

$$\bar{IS}_i = \frac{\sum_{j=1}^{n} IS_j}{n} \tag{16}$$

$i: Usuario$

$n: Cantidad\ de\ tweets\ relevantes\ del\ usuario\ por\ mes$

2- Luego promediamos las medias de los IS de cada usuario por mes, obteniendo la opinión media del mes de las personas:

$$\bar{IS}_t = \bar{\bar{IS}}_i = \frac{\sum_{i=1}^{m} \bar{IS}_i}{m} \tag{17}$$

t : $Número\ de\ mes\ del\ año$
i: $Usuario$
m: $Cantidad\ de\ usuarios\ por\ mes$

3- Por último, dado que las encuestas para obtener el ICEA se realizan cuatrimestralmente y captan la confianza cuatrimestral de las personas, creímos conveniente obtener la media cuatrimestral del *sentiment*. Esto representa que la persona infiere en cómo se sintió mes a mes a lo largo de los meses pasados para concluir en su confianza a la hora de completar la encuesta:

$$\overline{IS}_c = \frac{\overline{IS}_t + \overline{IS}_{t-1} + \overline{IS}_{t-2} + \overline{IS}_{t-3}}{4} \quad (18)$$

t: $Número\ de\ mes\ del\ año$
c: $Cuatrimestre\ del\ año$

Para calcular estos promedios, recurrimos al uso de tablas dinámicas (pivot tables) en Python, *pivot tables* es una operación comúnmente usada en hojas de cálculos y otros programas que operan con datos tabulados. *Pivot table* toma una columna simple de datos como *input* y agrupa las entradas en una tabla de dos dimensiones que brinda operaciones multidimensionales entre los datos. Es una función contenida dentro de la librería pandas y es una herramienta sumamente útil para la manipulación de grandes datos.

4.4 Análisis para vincular el índice de *sentiment* con el ICEA mediante un modelo predictivo

Obtenido el IS por cuatrimestre, debimos buscar un modelo que ajuste estos valores, los cuales son nuestra muestra de variables independientes (X), a los valores dependientes ICEA (Y):

X: Variable explicada
Y: Variable explicativa

Dada la cantidad de palabras tenidas en cuenta para hacer el filtro de *tweets,* y siendo estas representativas de distintos subtemas como, por ejemplo, la macroeconomía nacional, la política, medidas impositivas, etc., pueden afectar de diferente manera a la confianza del sector agropecuario. Optamos por dividir la variable explicativa en distintos subgrupos ya que cada uno puede afectar de manera más o menos significativamente a la predicción del ICEA. Es una forma de poder tener un mejor ajuste del modelo

eliminando aquellos subgrupos de palabras que no resultan significativos para explicar el Índice tradicional y dejando aquellos subgrupos que si explican en gran medida a este índice.

Las palabras clave (*keywords)* fueron clasificadas de la siguiente manera:

Subgrupo (j)	Variable (X_j)	Keywords
1	X_1	clima, inundacion, inundaciones, lluvia, lluvias, sequia, sequias, helada, heladas
2	X_2	sanitario, plagas, agroquímica, agroquímicos
3	X_3	impuesto, impuestos, tributario, tributarios, tributaria, tributarias, retención, retenciones, derecho de exportación, derechos de exportación
4	X_4	commodity, comodity, commodities, comodities
5	X_5	regulación, regulaciones
6	X_6	tambo, tambos
7	X_7	inflación, dolar, dolares, tasa, tasas
8	X_8	situacion
9	X_9	gobierno, presidente, presidencia, ministerio, ministro, ministra, política
10	X_{10}	gobernacion, gobernador, gobernadora
11	X_{11}	economia, económica, economicas, hacienda
12	X_{12}	costos, precio, precios

Tabla 3. Segmentación de palabras claves en subgrupos

Para crear este modelo, partimos de un **modelo lineal multiple**.

Modelo Lineal Múltiple

El modelo lineal múltiple relaciona cada una de las 12 variables a través de un coeficiente que representa el grado en que esa variable es capaz de explicar a la variable respuesta (el ICEA). El modelo es el siguiente:

$$Y_c = \beta_0 + \beta_1 X_{1c} + \beta_2 X_{2c} + \beta_3 X_{3c} + \beta_4 X_{4c} + \beta_5 X_{5c} + \beta_6 X_{6c} + \beta_7 X_{7c} + \beta_8 X_{8c} + \beta_9 X_{9c} + \beta_{10} X_{10c} + \beta_{11} X_{11c} + \beta_{12} X_{12c} + \varepsilon_c$$

Siendo,

Y_c: Valor del ICEA del cuatrimestre c
X_{jc}: $Sentiment$ medio de los $tweets$ que contienen las palabras del subgrupo i dentro del cuatrimestre c
ε_c: Error de la predicción del cuatrimestre c

Para estimar los coeficientes del modelo recurrimos a **R**. Con la función lm() de R, logramos estimar un modelo lineal múltiple a través del el método de **cuadrados mínimos.**

Dada la cantidad de variables X existía un alto grado de correlación entre las variables. El método utilizado para eliminar variables y dejar en el modelo solo las variables que realmente explican el ICEA y son independientes entre sí, fue el **LASSO**. Luego de este procesamiento, obtuvimos el modelo lineal múltiple final. Como resultado, X_1, X_3 y X_5 fueron las que representaban significativamente al modelo, es decir, aquellos *tweets* que hicieran referencia al clima, a temas impositivos o a temas sobre la macroeconomía argentina representaban el *sentiment* de los usuarios. De lo contrario, temas sanitarios, commodities, regulaciones, tambos, situación, gobierno, economía y precios no representaban de la misma manera al índice.

Modelo Dinámico

El ICEA es una serie de tiempo que representa la confianza de los productores agropecuarios a intervalos regulares, y muestra estructura dinámica. El modelo que contempla las variables explicativas pasadas para predecir la variable explicativa actual es el **modelo dinámico**. A través de una estimación del

modelo dinámico sobre el modelo múltiple lineal estimado previamente obtuvimos un modelo lo más representativo del ICEA posible.

Así como en el modelo de regresión múltiple y es una función de los predictores j , y ε_c es usualmente considerado un término del error no correlacionado (de ruido blanco), en modelo dinámico el error de una regresión contiene autocorrelación. El error ε_c es reemplazado por η_c en la ecuación y se considera que η_c sigue un modelo AR(p):

$$Y_c = \beta_0 + \beta_1 X_{1c} + \beta_3 X_{3c} + \beta_7 X_{7c} + \eta_c$$

$$\eta_c = \phi_1 \eta_{c-1} + \phi_2 \eta_{c-2} + \varepsilon_c$$

Donde ε_c es un ruido blanco.

ϕ_k: $Coeficiente\ dinámico$
η_{c-k}: $Error\ del\ k-$ésimo $cuatrimetre\ pasado$
k: $Decalage\ cuatrimestral$

Para la estimación del modelo dinámico utilizamos **E-Views** que emplea el método de **máxima verosimilitud**.

Una vez creado el modelo múltiple lineal, eliminada su correlación y agregadas las componentes dinámicas obtuvimos finalmente el modelo definitivo para poder predecir de la manera más semejante posible al valor del ICEA.

5. Resultados

Mediante la metodología descripta anteriormente obtuvimos la base de datos madre, conteniendo aquellos *tweets* publicados por los usuarios relevantes dentro del ámbito agropecuario en la Argentina desde 05-2014 hasta 07-2020. Dentro de los *tweets* publicados por dichos usuarios, filtramos aquellos que contuvieran alguna de las 47 palabras clave. De los 734.257 *tweets* extraídos en el primer filtro, 45.061 *tweets* resultaron contener al menos una de las palabras clave. Determinamos el *sentiment* de cada *tweet* mediante Google Cloud NL, obteniendo *score* y *magnitude*.

De la selección resultante, clasificamos los *tweets* según el subgrupo *j* (*j*= 1,2,3..12) dependiendo de la palabra clave que contuviera y según el mes t en que fue publicado. El tamaño de muestra por bloque se detalla en el apéndice 7.1.1. A partir de esta clasificación calculamos con la fórmula (17) el Índice de *Sentiment* mensual para cada grupo de palabras, y luego mediante la fórmula (18) el IS cuatrimestral para cada grupo de palabras: X_1 a X_{12}.

En el siguiente gráfico se puede observar la cantidad de *tweets* publicados por cuatrimestre que contenía al menos una palabra de cada subgrupo j:

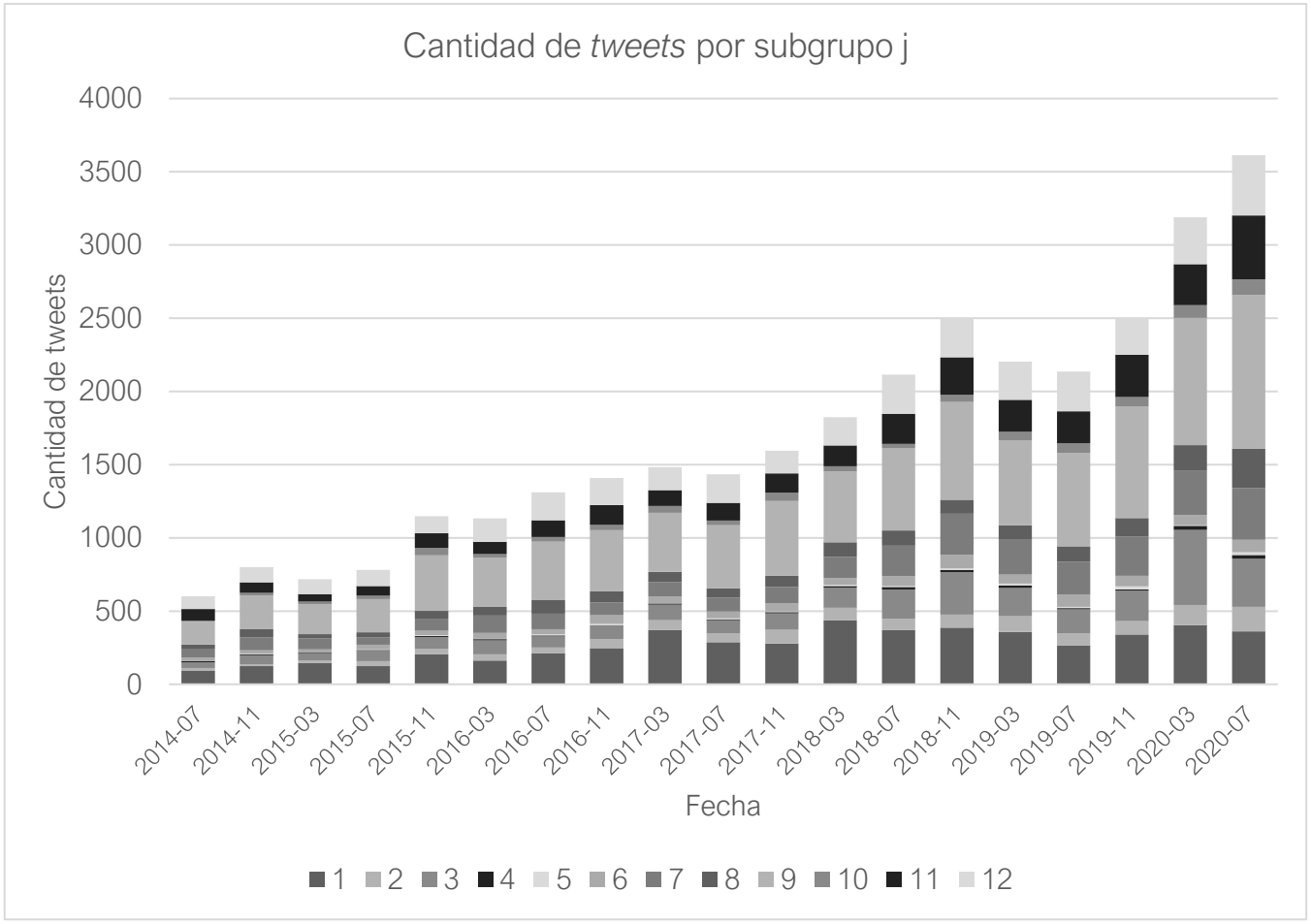

Gráfico 2 Cantidad de tweets por subgrupo j

Luego creamos un modelo lineal para explicar el Índice de Confianza del Empresario Agricultor medido por CREA (ICEA) mediante estos doce índices de *Sentiment*, resultando:

Estimadores	Subgrupo	Coeficiente	Error estándar	Estadístico t	p-value
b_0	intersección	83,42	42,02	1,985	0,0944
b_1	clima	-108,93	292,28	-0,373	0,7222
b_2	sanitario	80,30	216,25	0,371	0,7231
b_3	impositivo	32,49	132,98	0,244	0,8151
b_4	comercio internacional	55,71	146,94	0,379	0,7177
b_5	regulaciones	-50,01	56,37	-0,887	0,4092
b_6	agro	-135,21	190,99	-0,708	0,5055
b_7	macroeconomía nacional	396,15	338,26	1,171	0,2860
b_8	situación	-190,44	241,81	-0,788	0,4609
b_9	política nacional	179,79	182,31	0,986	0,3621
b_{10}	política provincial	-14,08	158,90	-0,089	0,9323
b_{11}	economía	184,07	272,08	0,677	0,5239
b_{12}	economía sectorial	-310,97	205,39	-1,514	0,1808

Tabla 4 Estimadores resultantes de modelo de regresión – primera iteración

Error estándar residual	12,02
R^2	0,8245
R^2_{adj}	0,4735
Estadístico F	2,349
p-value	0,1519

Tabla 5 Estadísticas de modelo – primera iteración

El modelo alcanza un coeficiente de determinación de R^2_{adj} 0,47, que es insuficiente para lograr predicciones precisas.

Además, el modelo presenta un elevado nivel de colinealidad, como muestran los indicadores VIF:

b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	b_9	b_{10}	b_{11}	b_{12}
14,254	22,42	1,518	13,185	4,217	16,02	13,18	6,70	4,18	2,85	17,423	7,65

Tabla 6 Indicadores VIF primera iteración de regresión

Aplicamos LASSO para tratar la colinealidad, mediante la biblioteca glmnet de R. El algoritmo procede estableciendo una serie de valores de la penalidad y calculando el error cuadrático medio para cada uno mediante cross-validation. Es decir, recorre toda la muestra omitiendo cada observación, estimando el modelo sobre las observaciones restantes, y calculando el error de predicción para el valor omitido. La cross-validation asegura que el error sea obtenido en condiciones realistas: las predicciones se realizan sobre valores que el modelo no conoce.

El siguiente gráfico muestra el error cuadrático medio calculado de esa manera en función de la penalidad (log(λ)). De este modo se obtiene el valor de la penalidad que minimiza el error de predicción.

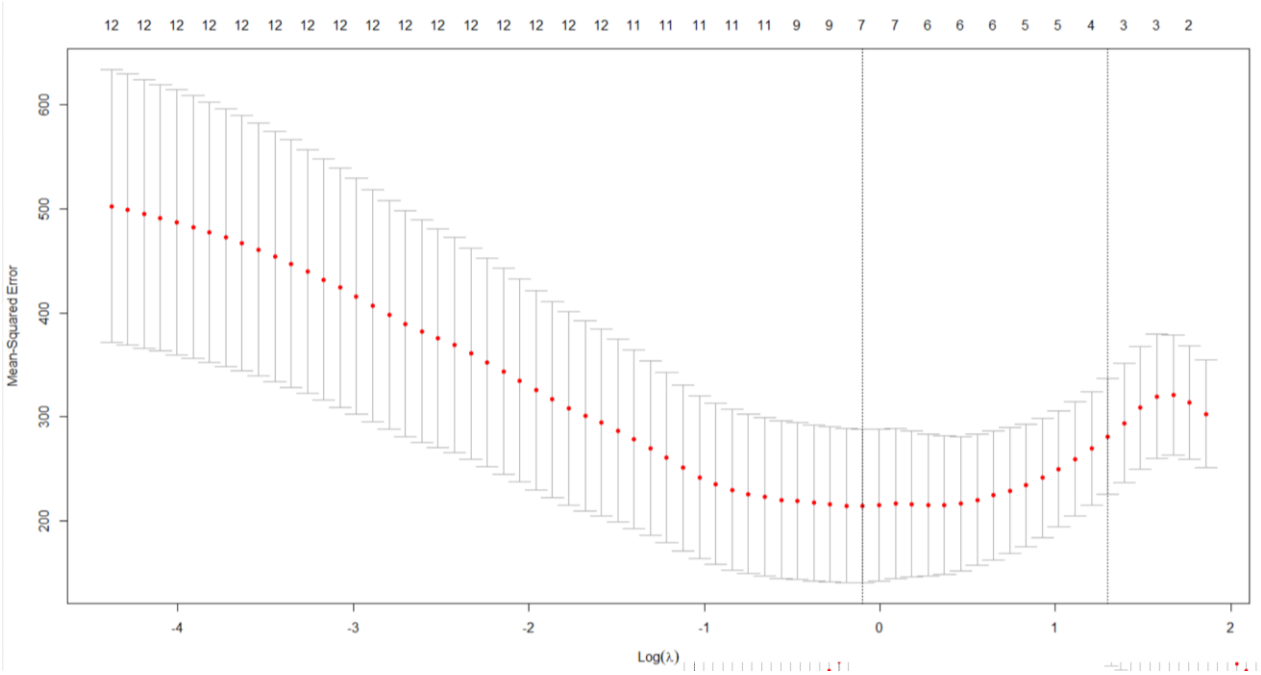

Gráfico 3 Error cuadrático medio en función de log(λ)

El modelo sugerido por el LASSO tiene 5 variables explicativas:

Estimador	Subgrupo	Coeficiente	Error estándar	Estadístico t	p-value
b_0	intersección	68,97	17,58	3,923	0,001748
b_1	clima	-412,22	95,22	-4,329	0,000818
b_3	impositivo	97,16	99,21	0,979	0,345292
b_4	comercio internacional	29,21	41,16	0,710	0,490461
b_7	macroeconomía nacional	288,55	145,77	1,980	0,069333
b_9	política nacional	218,47	132,70	1,646	0,123632

Tabla 7 Estimadores resultantes modelo regresión – segunda iteración

Error estándar residual	10,9
R^2	0,6877
R^2_{adj}	0,5676
Estadístico F	5,726
p-value	0,005238

Tabla 8 Estadísticas del modelo – segunda iteración

Y no tiene colinealidad grave como muestran los siguientes VIF, todos inferiores a 4:

b_1	b_3	b_4	b_7	b_9
1,8422	1,03	1,26	2,98	2,7012

Tabla 9 Valores VIF segunda iteración de regresión

Tomando como punto de partida este modelo estático, formulamos diversos modelos dinámicos introduciendo la variable respuesta rezagada y aplicando modelos ARMA a la perturbación aleatoria. El mejor modelo (criterio de Schwarz) es el de la tabla siguiente, donde se retuvieron las variables explicativas X_1, X_3 y X_7, referidas a temas climáticos, impositivos y de macroeconomía nacional.

Estimador	Coeficiente	Error estándar	Estadístico t	p-valor
b_0	53,18	20,76	2,56	0,0283
b_1	-322,72	106,61	-3,027	0,0127
b_3	118,64	45,53	2,606	0,0262
b_7	261,78	128,38	2,039	0,0688
b_{c-1}	0,603	0,197	3,062	0,012
φ_1	-0,46	0,488	-0,94	0,37
φ_2	-0,677	0,268	-2,53	0,03

Tabla 10 Estimadores de modelo de regresión y ARMA

Los indicadores de desempeño general del modelo son:

R^2	0,895
R^2_{adj}	0,821
SBC	7,455
SE_{Reg}	6,85
SS_{Res}	469

Tabla 11 Estadísticas del modelo

El coeficiente de determinación alcanzado permite hacer predicciones razonablemente precisas.

La cantidad de *tweets* que contenían al menos una de estas palabras, contenidas en los subgrupos 1, 3 y 7, se pueden observar en el siguiente gráfico:

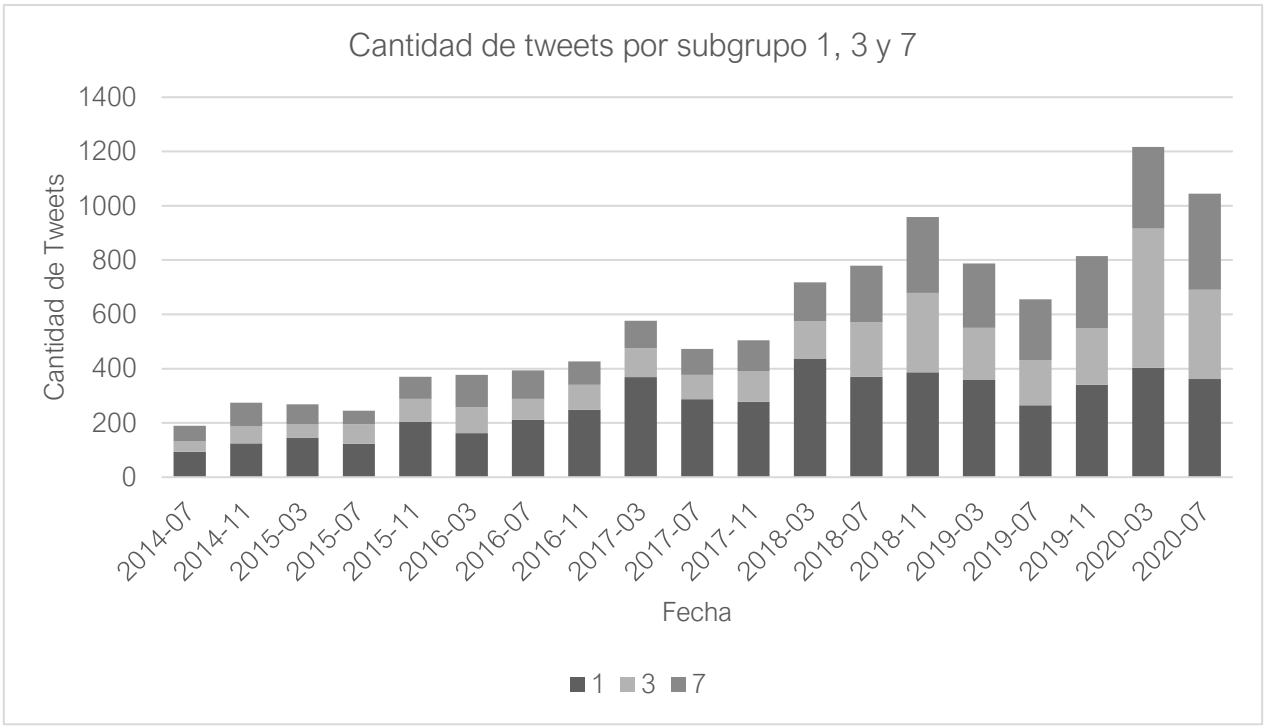

Gráfico 4 Cantidad de tweets por subgrupo 1,3 y 7

La evolución del *sentiment* por subgrupo fue:

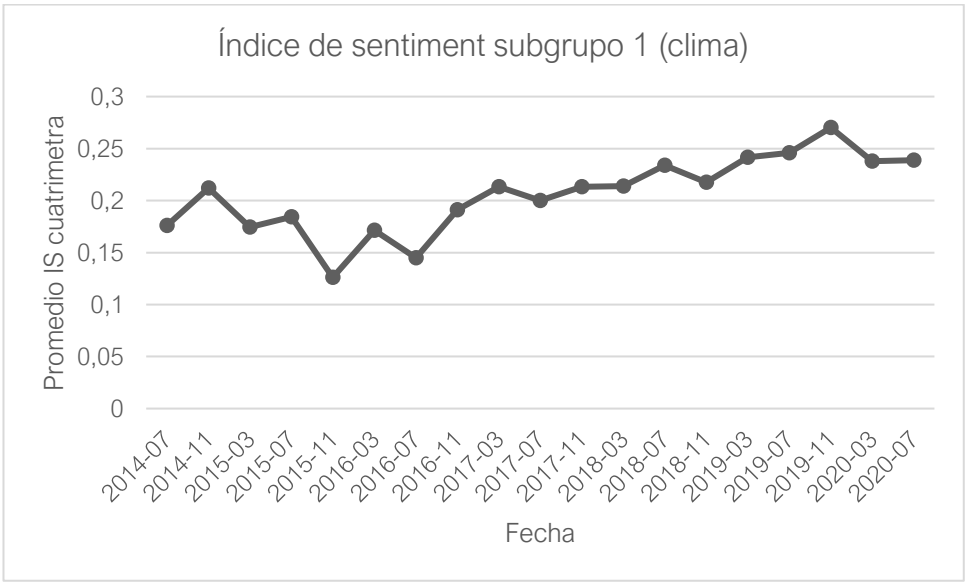

Gráfico 5 Índice de sentiment subgrupo 1

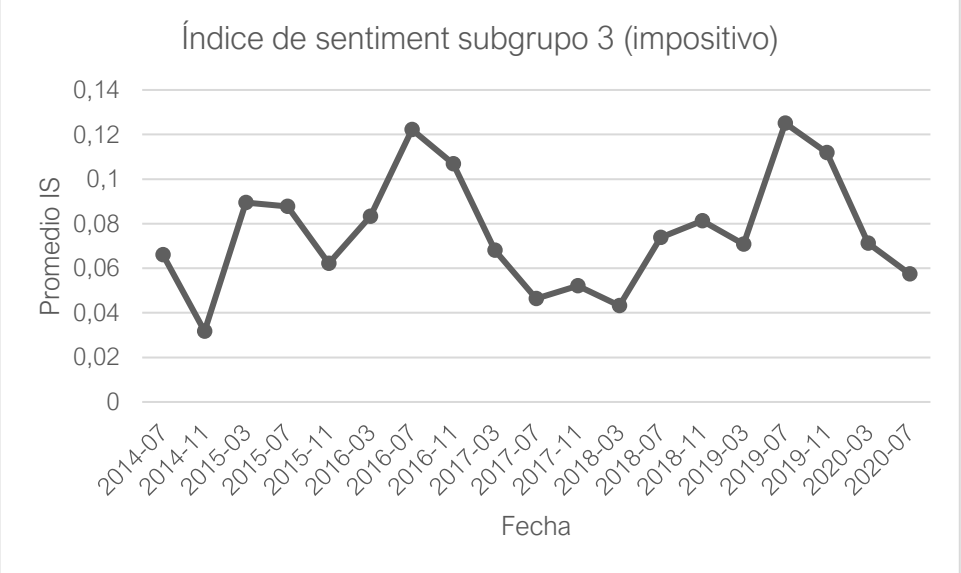

Gráfico 6 Índice de sentiment subgrupo 3

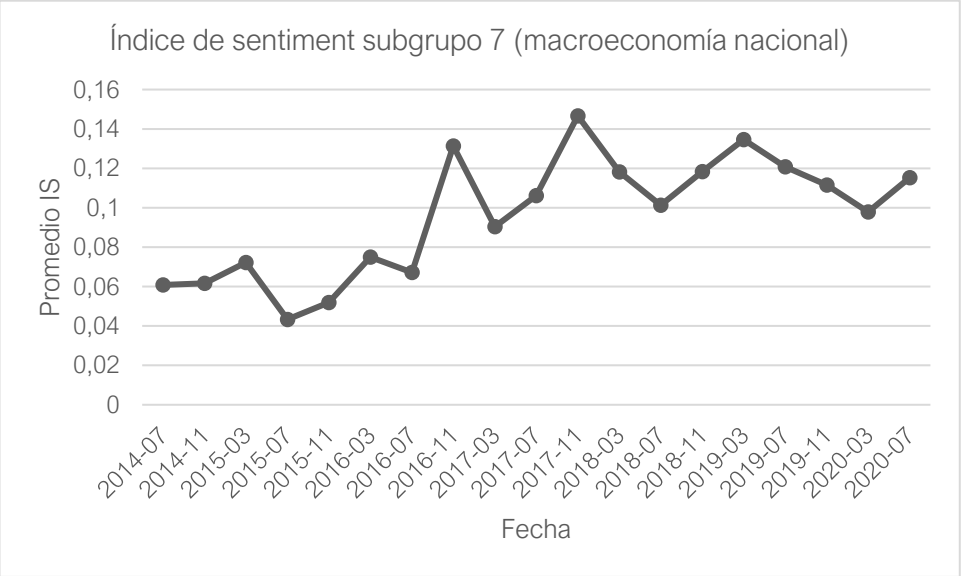

Gráfico 7 Índice de sentiment subgrupo 7

El modelo estadístico para predecir valores futuros del Índice de Confianza del Empresario Agricultor en base a los *sentiment* obtenidos es:

$$\tilde{Y}_c = 53,18 - 322,72X_1 + 118,64X_3 + 261,78X_7 + 0,603Y_{c-1} + \eta_c \quad (19)$$

$$\eta_c = -0,46\eta_{c-1} - 0,677\eta_{c-2} \quad (20)$$

La predicción para cada valor de $\tilde{Y}_c$ se realiza sumando la del modelo estático (19) con la del modelo dinámico (20) siendo η_{c-1} el valor residual entre la predicción de la componente estática de $\tilde{Y}_c$ con el valor observado Y_c.

A partir del modelo realizamos la predicción esperada para 11-2020 y la comparamos con el valor real medido por la encuesta cuatrimestral:

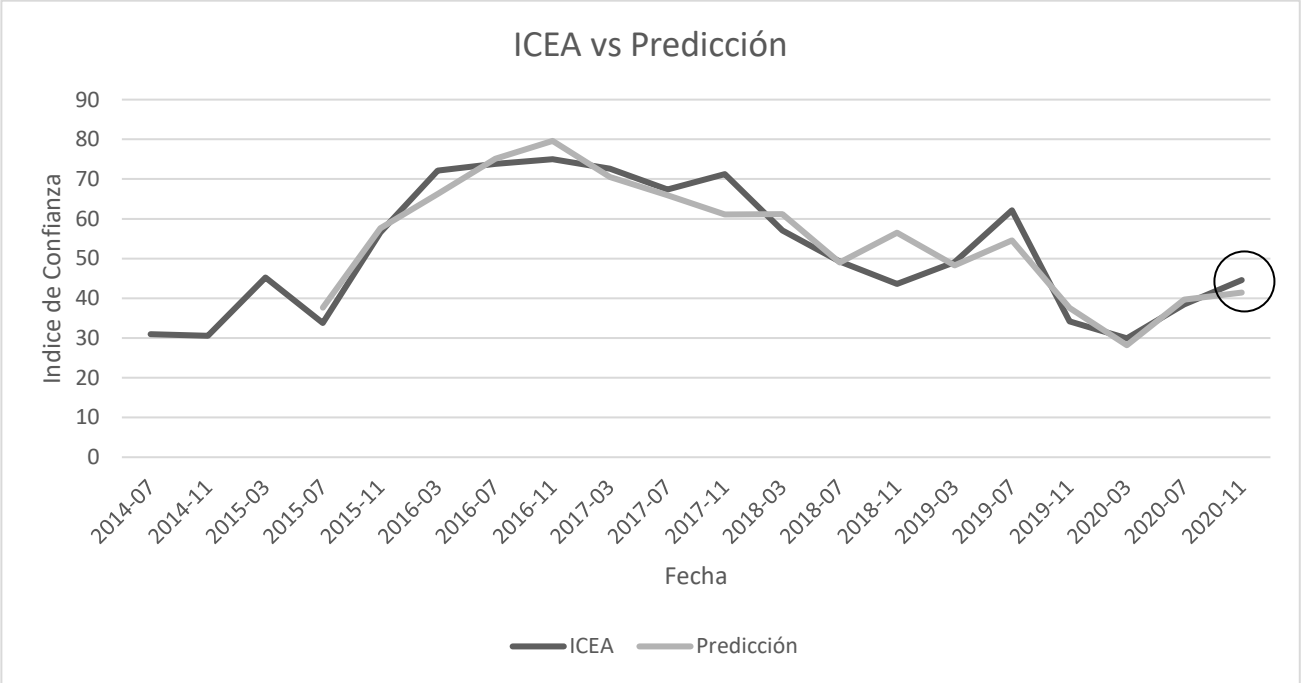

Gráfico 8 ICEA vs Predicción

$\tilde{Y}_{11-20} = 41,1$
$ICEA_{11-20} = 44,6$

Como se observa en el gráfico, el valor de ICEA de noviembre se asemeja a la predicción del modelo de forma satisfactoria.

6. Conclusión

La medición de la confianza del productor agropecuario es un parámetro de gran importancia en economías donde parte de su actividad depende de la agricultura. El sector agropecuario es un gran aportador de divisas para el país debido a sus volúmenes de exportación, una fuente de trabajo y un productor de materia prima para otras industrias. El nivel de relevancia en la economía nos lleva a su estudio. La Argentina es un país que sufrió cambios de gobiernos que diferían mucho entre sí, y no siguieron una misma línea de políticas referidas a la agricultura. La inestabilidad de políticas para el sector fomentó y fomenta la variabilidad en la confianza del productor agropecuario, que se ve reflejado en su comportamiento frente a la producción agropecuaria, lo que afecta a la actividad económica del país. Es por eso que la evolución de este índice es un parámetro que sirve potencialmente para inferir el desempeño económico general.

Actualmente, el índice de confianza del productor agropecuario (ICEA) es medido por la Asociación Argentina de Consorcios Regionales de Experimentación Agrícola (CREA). Dado el alto costo del actual mecanismo mediante encuestas, la alternativa de predecir este índice mediante un modelo estadístico aportaría grandes beneficios. Por un lado, permitiría saber con antelación los resultados del índice y eventualmente reemplazar las encuestas. Por otro, significaría un gran ahorro de tiempo y costos ya que se haría de forma remota e inmediata.

Para analizar la plausibilidad de la herramienta propuesta, al principio del trabajo se plantearon tres preguntas a responder. La primera fue si es posible estimar la confianza del productor agropecuario a través de lo que *twittean* usuarios interesados en el tema. En el presente trabajo se logró encontrar satisfactoriamente la relación entre el ICEA y el *sentiment* representado en *tweets* sobre distintos subtemas dentro de la agricultura. Esto se alcanzó, contestando a la segunda pregunta, a través de un modelo de regresión múltiple combinado con un modelo dinámico. El modelo resultante consiguió un coeficiente de determinación de 0,89 (siendo 1 el valor de mejor ajuste), lo cual se concluye que se logró un muy buen ajuste. La tercera pregunta cuestiona qué información contenida en los *tweets* influencian significativamente en la confianza del productor agropecuario argentino. El estimador del sentimiento general de aquellas personas relevantes en el sector agropecuario argentino se vio afectado significativamente por tres temáticas distintas. Por un lado, es afectado por la variable del modelo que representa el *sentiment* de aquellos *tweets* que contuvieran palabras referidas al clima. Cuando los usuarios se refieren a heladas, sequias, lluvias e inundaciones, el *sentiment* de los *tweets* publicados influye negativamente en el índice de sentimiento general. Es decir, a medida que el índice de sentimiento con respecto al clima aumenta, el

general disminuye. Dado que la variable del clima fue la que tuvo la muestra más grande, comparada con las otras dos variables relevantes, se deduce que los datos conteniendo palabras referidas al clima contuvieron un mayor ruido. Esto propone una mayor refinación de los datos a futuro, para poder obtener aquellos *tweets* referidos al clima que expliquen exclusivamente el sentimiento del usuario con respecto al clima en la agricultura. Por otro lado, otra variable que influye en el *sentiment* general es el *sentiment* con respecto a temas impositivos, es decir el reflejado en *tweets* que se refieran a impuestos, tributarios, retenciones o derechos a la exportación afectan de manera directa al *sentiment* general. Por último, la tercera variable que afecta al estimador es el *sentiment* de las personas para con la macroeconómica argentina. *Tweets* conteniendo palabras como dólar, inflación y tasa se las considera parte de esta temática. El modelo refleja que a medida que la población presenta un contento o descontento sobre retenciones, impuestos, inflación o dólar, el *sentiment* general será de contento o descontento también respectivamente. La influencia de estas dos variables en el *sentiment* general es un comportamiento lógico y esperable, ya que estos temas representan claros obstáculos para los productores agropecuario y afecta inmediatamente en el *sentiment* resultante del usuario.

Es importante destacar que el *sentiment* con respecto a temas sanitarios, comercio internacional, regulaciones, política nacional y provincial, economía y economía sectorial resultaron no del todo relevante a la hora de estimar la actitud de las personas. Se concluye que la confianza del productor se ve mayoritariamente afectada por la variabilidad de decisiones impositivas tomadas por el gobierno, el clima y la inestabilidad monetaria del país (inflación, dólar y tasas).

La incorporación de componentes dinámicas en el modelo indica un grado de vinculación de la confianza actual del productor con la de los dos cuatrimestres pasados.

Como conclusión, la predicción de la confianza del productor agropecuario a través de *tweets* es posible y con una precisión aceptable. Comprobamos que ciertos *tweets* con ciertas palabras logran representar lo que el ICEA refleja. Como próximos pasos proponemos una refinación en la recolección de datos, especialmente aquellos datos sobre el clima, y un aumento del tamaño de muestra – actualmente n= 19 – para poder conseguir un resultado aún más preciso.

7. Apéndice

7.1 Tamaño de muestra

$Var/_{Mes}$	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	$\sum X_t$
2014-05	32	7	17	4	0	4	18	7	62	4	30	32	217
2014-06	36	8	12	2	1	9	14	14	48	1	28	31	204
2014-07	26	2	9	2	0	10	25	13	46	3	20	23	179
2014-08	26	7	19	0	2	4	28	15	58	5	19	31	214
2014-09	27	1	11	4	1	9	27	20	59	6	18	26	209
2014-10	36	0	16	0	1	6	19	8	61	5	18	21	191
2014-11	36	4	16	2	0	5	14	12	52	3	17	25	186
2014-12	34	6	9	2	0	4	24	6	41	4	10	24	164
2015-01	36	2	12	2	0	4	22	8	55	5	11	30	187
2015-02	37	3	7	1	0	7	12	5	49	3	13	25	162
2015-03	38	7	21	3	0	4	17	9	58	7	17	23	204
2015-04	35	7	17	0	1	7	9	8	45	6	8	24	167
2015-05	32	10	18	1	0	7	12	3	49	6	19	25	182
2015-06	36	11	17	4	1	11	13	8	65	4	17	26	213
2015-07	21	7	19	0	1	8	16	14	68	8	21	37	220
2015-08	100	6	15	2	0	9	13	20	75	12	21	27	300
2015-09	32	10	13	2	4	6	15	10	55	6	15	24	192
2015-10	29	11	21	1	2	10	20	9	87	18	21	28	257
2015-11	44	8	34	1	0	6	34	15	162	14	43	38	399
2015-12	40	8	47	1	0	6	45	11	122	5	24	35	344
2016-01	32	11	17	5	0	10	18	16	65	1	14	42	231
2016-02	59	11	13	3	0	11	28	16	55	6	23	42	267
2016-03	31	12	19	0	2	13	28	19	91	12	21	43	291
2016-04	82	4	23	1	5	9	27	32	101	5	29	44	362
2016-05	48	9	19	2	1	4	25	31	105	7	24	52	327
2016-06	36	12	13	1	2	10	27	18	87	9	26	44	285
2016-07	46	14	22	1	1	12	26	14	103	12	36	50	337
2016-08	24	9	16	1	3	19	26	20	87	7	24	51	287
2016-09	57	19	26	1	1	16	28	18	119	12	38	55	390
2016-10	102	18	33	0	3	11	16	18	115	14	36	36	402
2016-11	65	15	18	0	2	13	16	21	96	3	38	42	329
2016-12	89	13	29	3	0	6	29	18	108	7	26	30	358
2017-01	122	20	18	1	0	18	22	25	86	10	23	42	387
2017-02	82	14	30	4	0	9	19	15	85	15	30	42	345
2017-03	76	21	29	1	2	11	31	12	124	13	30	42	392
2017-04	100	13	15	0	1	11	32	18	105	12	23	36	366

2017-05	69	16	15	2	2	11	19	15	113	8	29	54	353
2017-06	58	20	23	2	3	6	17	14	94	7	34	57	335
2017-07	60	11	37	4	0	18	27	17	118	6	34	47	379
2017-08	69	28	34	2	2	13	34	16	145	18	31	43	435
2017-09	96	25	21	1	1	17	29	25	121	13	34	36	419
2017-10	58	24	18	2	2	13	22	12	130	12	24	39	356
2017-11	55	16	40	3	0	12	28	23	115	13	44	35	384
2017-12	77	22	39	4	0	7	33	18	126	13	28	32	399
2018-01	79	21	36	0	3	8	35	21	98	8	24	46	379
2018-02	135	21	27	2	6	20	29	31	132	7	41	48	499
2018-03	146	22	36	3	1	12	46	30	126	9	50	65	546
2018-04	133	16	45	5	3	13	31	20	113	10	34	70	493
2018-05	118	17	66	3	1	16	76	24	163	3	62	65	614
2018-06	62	22	37	4	1	18	49	28	133	5	49	65	473
2018-07	57	21	53	5	1	21	52	33	155	9	61	67	535
2018-08	59	22	46	6	0	22	80	26	160	7	56	65	549
2018-09	90	17	127	5	2	23	97	25	188	10	82	77	743
2018-10	113	23	64	2	5	31	63	24	162	16	70	61	634
2018-11	125	26	55	1	3	17	40	21	157	16	50	58	569
2018-12	79	25	32	7	3	13	37	21	140	8	39	53	457
2019-01	120	28	57	5	2	13	46	30	118	16	48	56	539
2019-02	77	30	58	3	3	14	71	22	149	21	54	62	564
2019-03	82	25	45	2	4	22	83	27	174	13	76	90	643
2019-04	78	20	42	1	2	26	68	22	150	17	51	74	551
2019-05	71	28	42	3	3	20	51	31	178	11	57	74	569
2019-06	67	21	46	3	3	19	50	30	156	22	51	57	525
2019-07	49	12	36	1	1	17	56	23	155	14	60	66	490
2019-08	54	15	55	4	5	16	81	36	224	16	80	65	651
2019-09	93	26	36	1	6	12	77	32	166	14	82	58	603
2019-10	109	25	50	1	5	19	71	29	200	29	75	65	678
2019-11	84	25	67	2	6	24	38	30	171	8	51	58	564
2019-12	106	15	178	5	0	14	92	25	255	29	70	77	866
2020-01	87	34	99	6	1	21	75	15	154	14	52	76	634
2020-02	105	45	79	5	4	14	59	26	159	19	49	70	634
2020-03	105	45	157	10	1	20	75	111	300	24	110	97	1055
2020-04	100	32	90	8	1	17	91	91	281	20	109	121	961
2020-05	67	51	81	5	3	17	106	54	239	26	98	106	853
2020-06	88	42	84	7	4	20	86	60	287	33	99	95	905
2020-07	107	42	74	4	11	31	71	63	243	29	130	88	893
$\sum X_i$	5171	1316	2917	197	141	986	2986	1727	9297	843	3109	3786	**32476**

7.2 Códigos de programación

Referencias

Códigos realizados en Python: '*'

Códigos realizados en R: '**'

7.2.1 Recolección de datos

Identificación de usuarios relevantes a analizar*

```
import time
import tweepy
from tweepy import Stream
from tweepy import OAuthHandler
from tweepy.streaming import StreamListener
import config


consumer_key = "Uhg6QRH4QMQF1KUlb********"
consumer_secret ="ORSlO96x45L26PGUc3BSmZopBSgPs2PndahPyy4a**********"
access_token ="220010262-dD350JbeK7LqMh1xQOzyQRWgXoeNxk**********"
access_secret = "todkMVCXmzivMJ3LEdHyC8Q3698TfiKFBzE**********"

auth = OAuthHandler(config.consumer_key, config.consumer_secret)
auth.set_access_token(config.access_token, config.access_secret)
api = tweepy.API(auth, wait_on_rate_limit=True)

ids_crea = []
for page in tweepy.Cursor(api.followers_ids, screen_name="crea_arg").pages():
    ids_crea.extend(page)

ids_inta = []
for page in tweepy.Cursor(api.followers_ids,
screen_name="intaargentina").pages():
    ids_inta.extend(page)

ids_agrositio =[]
for page in tweepy.Cursor(api.followers_ids, screen_name="agrositio").pages():
    ids_agrositio.extend(page)

ids_infocampoweb =[]
for page in tweepy.Cursor(api.followers_ids, screen_name="infocampoweb").pages():
    ids_infocampoweb.extend(page)

ids_LNcampo =[]
for page in tweepy.Cursor(api.followers_ids, screen_name="LNcampo").pages():
    ids_LNcampo.extend(page)
```

```
ids_tot =
list(set(ids_crea).intersection(ids_inta).intersection(ids_agrositio).intesection
(ids_infocampoweb))
ids_tot3 = list(set(ids_tot).intersection(ids_LNcampo))


with open('outTotal.txt', 'a') as myfile:
    counter = 0
    screenNames = []
    ids_ = []

    for id_ in ids_tot3:
        counter = counter +1
        u = api.get_user(id_)

        screenNames.append(u.screen_name)
        ids_.append(id_)
        if counter >50:

            print("counter mark!")

            #reset counter
            counter = 0
            #export file
            for id2_, name_ in zip(ids_, screenNames):
                myfile.write(str(id2_) +","+str(name_)+"\n" )

            #reset screenNames
            screenNames = []
            ids_=[]

myfile.close()
```

In [ ]:

Extracción de *tweets**

```
import sys,getopt,datetime,codecs
if sys.version_info[0] < 3:
    import got
else:
    import GetOldTweets3 as got

def getTweetsByUserId (userId):

    with open('log.txt', 'a') as myfile:

        f1 = "2014-05-01"
        f2 = "2020-07-31"

        tweetCriteria = got.manager.TweetCriteria().setUsername(userId)

        tweetCriteria.since = f1
```

```
        tweetCriteria.until = f2

        outError = -1

        tweets = got.manager.TweetManager.getTweets(tweetCriteria)

        if len(tweets)==0:
            outError = -2


        if len(tweets) < 500 and len(tweets)>0:

            outError = 1

            try:

                outputFileName = "tweets/" + userId + ".csv"

                outputFile = codecs.open(outputFileName, "w+", "utf-8")


outputFile.write('username;;date;;retweets;;favorites;;text;;geo;;mentions;;hasht
ags;;id;;permalink')

                print('Searching...\n')

                def receiveBuffer(tweets):
                    if len(tweets)==0:
                        outError = -2
                    else:
                        for t in tweets:

outputFile.write(('\n%s;;%s;;%d;;%d;;%s;;%s;;%s;;%s;;%s;;%s' % (t.username,
t.date.strftime("%Y-%m-%d %H:%M"), t.retweets, t.favorites, t.text, t.geo,
t.mentions, t.hashtags, t.id, t.permalink)))
                        outputFile.flush()
                        print('More %d saved on file...\n' % len(tweets))

                if len(tweets)<500:
                    receiveBuffer(tweets)

            except arg:
                print('Arguments parser error, try -h' + arg)
                outError = 0
            finally:
                outputFile.close()
                print('Done. Output file generated "%s".' % outputFileName)


        myfile.write("\n" + userId + "," + str(outError ))


import pandas as pd
```

```
idsTot = pd.read_csv('outTotal.txt', header=None)

idsList = list(idsTot[1])
i=0
for user in idsList:
    print(user)
    getTweetsByUserId(user)
```

7.2.2 Procesamiento de datos

Segmentación de *tweets**

```
from os import listdir
import pandas as pd
fileList = listdir("tweets")
fileList = [f for f in fileList if f[0]!="."]
print(str(len(fileList)) + ' users analyzed.')

totalTweets = 0
fileListWContent = []
dfTweets = []
for fileName in fileList:
    print(fileName)
    df = pd.read_csv("tweets/" + fileName, sep=";", engine='python', encoding='utf-8')
    if df.shape[0]>1:
        fileListWContent.append(fileName)
        totalTweets = totalTweets + len(df['text'])
        dfTweetsTemp = df[['date','text']]
        dfTweetsTemp['user'] = fileName[:(len(fileName)-4)]
        dfTweets.append(dfTweetsTemp)


dfTweets = pd.concat(dfTweets, axis=0)

dfTweets["date"] = pd.to_datetime(dfTweets["date"])

dfTweets = dfTweets.loc[dfTweets['date'] < '2020-07-31']
dfTweets = dfTweets.loc[dfTweets['date'] >= '2014-05-01']
print(dfTweets)
dfTweets['month'] = dfTweets.date.map(lambda x: x.strftime('%Y-%m'))

monthlyTable = dfTweets["date"].groupby(dfTweets.date.dt.to_period("M")).agg('count')
months = list(monthlyTable.index)

months = [str(month) for month in months]
dfTweets = dfTweets.reset_index()
```

Normalización de texto*

```
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from unidecode import unidecode

def tokenize(text):
    tokens = word_tokenize(text.lower())

    return tokens

def noaccent(List1):
    List2=[]
    for t in List1:
        List2.append(unidecode(t))
    return list(List2)

def intersection(wordList1, wordList2):
    wordList1=noaccent(wordList1)

    temp = list(set(wordList1).intersection(set(wordList2)) )
    out = 0
    if len(temp)>0:
        out=1
    return out
```

Filtración de *tweets**

```
relCorpus = list(pd.read_csv('relevante.txt', header=None)[0])

relVector = []

for t in dfTweets['text']:
    relVector.append(intersection(tokenize(str(t)), relCorpus))


dfTweets['relevant'] = relVector
```

Análisis de *sentiment**

```
import os
os.environ["GOOGLE_APPLICATION_CREDENTIALS"] = 'TweetsSentiment-e32a4d583ad1.json'
```

In [ ]:

```
from google.cloud import language
from google.cloud.language import enums
from google.cloud.language import types

client = language.LanguageServiceClient()
```

```
sentimentDict = {}
for i, row in dfTweets.iterrows():
    if dfTweets.iloc[i, dfTweets.columns.get_loc('relevant')]==1 :

        #sentimentDict[i]= dfTweets.iloc[i, dfTweets.columns.get_loc('text')]
        textToAnalize = dfTweets.iloc[i, dfTweets.columns.get_loc('text')]
        document = types.Document(
            content=textToAnalize,
            type=enums.Document.Type.PLAIN_TEXT,
            language='es')

        annotations = client.analyze_sentiment(document=document,
encoding_type='UTF8')
        sentimentDict[i] = annotations
        print(i, sentimentDict[i].document_sentiment.score,
sentimentDict[i].document_sentiment.magnitude)

import numpy as np

dfTweets['score'] = np.nan
dfTweets['magnitude'] = np.nan


for j in sentimentDict:
    print(j,
sentimentDict[j].document_sentiment.score,sentimentDict[j].document_sentiment.magnitud
e)
    dfTweets.iloc[j, dfTweets.columns.get_loc('score')] =
sentimentDict[j].document_sentiment.score
    dfTweets.iloc[j, dfTweets.columns.get_loc('magnitude')] =
sentimentDict[j].document_sentiment.magnitude
```

7.2.3 Modelo Lineal

Separación de variables*

```
import nltk

from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from unidecode import unidecode

def tokenize(text):
    tokens = word_tokenize(text.lower())

    return tokens

def noaccent(List1):
    List2=[]
    for t in List1:
```

```
        List2.append(unidecode(t))
    return list(List2)


def intersection(wordList1, wordList2):
    wordList1=noaccent(wordList1)


    temp = list(set(wordList1).intersection(set(wordList2)) )
    out = 0

    if len(temp)>0:
        out=1
    return out
```

x1: clima

```
x1 = list(pd.read_csv('x1.txt', header=None)[0])

varx1 = []

for t in dfTweets['text']:

    varx1.append(intersection(tokenize(str(t)), x1))

dfTweets['x1']=varx1
```

x2: sanitario

```
x2 = list(pd.read_csv('x2.txt', header=None)[0])

varx2 = []

for t in dfTweets['text']:

    varx2.append(intersection(tokenize(str(t)), x2))

dfTweets['x2']=varx2
```

x3: impositivo

```
x3 = list(pd.read_csv('x3.txt', header=None)[0])

varx3 = []

for t in dfTweets['text']:

    varx3.append(intersection(tokenize(str(t)), x3))

dfTweets['x3']=varx3
```

x4: internacional

```
x4 = list(pd.read_csv('x4.txt', header=None)[0])
varx4 = []
for t in dfTweets['text']:
    varx4.append(intersection(tokenize(str(t)), x4))
dfTweets['x4']=varx4
```

x5: regulaciones

```
x5 = list(pd.read_csv('x5.txt', header=None)[0])
varx5 = []
for t in dfTweets['text']:
    varx5.append(intersection(tokenize(str(t)), x5))
dfTweets['x5']=varx5
```

x6: agro

```
x6 = list(pd.read_csv('x6.txt', header=None)[0])
varx6 = []
for t in dfTweets['text']:
    varx6.append(intersection(tokenize(str(t)), x6))
dfTweets['x6']=varx6
```

x7: macro nacional

```
x7 = list(pd.read_csv('x7.txt', header=None)[0])
varx7 = []
for t in dfTweets['text']:
    varx7.append(intersection(tokenize(str(t)), x7))
dfTweets['x7']=varx7
```

x8: situacion

```
x8 = list(pd.read_csv('x8.txt', header=None)[0])
varx8 = []
for t in dfTweets['text']:
    varx8.append(intersection(tokenize(str(t)), x8))
dfTweets['x8']=varx8
```

x9: política nacional

```
x9 = list(pd.read_csv('x9.txt', header=None)[0])
varx9 = []
for t in dfTweets['text']:
    varx9.append(intersection(tokenize(str(t)), x9))
dfTweets['x9']=varx9
```

x10: política provincial

```
x10 = list(pd.read_csv('x10.txt', header=None)[0])
varx10 = []
for t in dfTweets['text']:
    varx10.append(intersection(tokenize(str(t)), x10))
dfTweets['x10']=varx10
```

x11: economia

```
x11 = list(pd.read_csv('x11.txt', header=None)[0])
varx11 = []
for t in dfTweets['text']:
    varx11.append(intersection(tokenize(str(t)), x11))
dfTweets['x11']=varx11
```

x12: economia sectorial

```
x12 = list(pd.read_csv('x12.txt', header=None)[0])

varx12 = []

for t in dfTweets['text']:

    varx12.append(intersection(tokenize(str(t)), x12))

dfTweets['x12']=varx12
```

Tablas *pivot**

```
import numpy as np

tab1=pd.pivot_table(dfTweets[dfTweets.x1==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab2=pd.pivot_table(dfTweets[dfTweets.x2==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab3=pd.pivot_table(dfTweets[dfTweets.x3==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab4=pd.pivot_table(dfTweets[dfTweets.x4==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab5=pd.pivot_table(dfTweets[dfTweets.x5==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab6=pd.pivot_table(dfTweets[dfTweets.x6==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab7=pd.pivot_table(dfTweets[dfTweets.x7==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab8=pd.pivot_table(dfTweets[dfTweets.x8==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab9=pd.pivot_table(dfTweets[dfTweets.x9==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab10=pd.pivot_table(dfTweets[dfTweets.x10==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab11=pd.pivot_table(dfTweets[dfTweets.x11==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()
tab12=pd.pivot_table(dfTweets[dfTweets.x12==1],values='score', index='user',
columns='month', aggfunc=np.mean).mean()


tab=pd.concat([tab1,tab2,tab3,tab4,tab5,tab6,tab7,tab8,tab9,tab10,tab11,tab12], axis=1)
tab
tab.to_csv('variables1.csv', sep=';')
```

Modelo de regresión múltiple**

```
# Agro - Indice de Confianza por Sentiment

setwd("C:/Python/Agro/Text analytics")
library(readxl)   # read Excel files
library(car)

rm(list=ls())   #clean memory

# read data
#dat <- read_excel("agro icea sent.xlsx",sheet="mes")
dat <- read_excel("agro icea sent.xlsx",sheet="cuatrim")

# data preparation
dat <- dat[,-1]

X <- model.matrix(lm(ICEA~.-1,data=dat))
y <- dat$ICEA

n=nrow(X);p=ncol(X)


# Linear Simple
mod01 <- lm(y~. ,data=as.data.frame(cbind(y,X)))
mod <- mod01
summary(mod)
vif(mod)


# Shrink: LASSO, Ridge, Elastic Net
library(glmnet)

#Select lambda via Cross Validation
modcv <- cv.glmnet(X, y, family="gaussian"
                   ,alpha=1,nlambda=100,lambda.min.ratio=ifelse(n<p,0.01,0.0001)
                   ,standardize=T,intercept=T
                   ,type.gaussian="covariance"
                   ,standardize.response=F
                   ,type.measure="mse"
                   ,nfolds=n
                   ,parallel=F)
modcv$lambda.min;log(modcv$lambda.min)
#print(cbind(modcv$lambda,modcv$nzero))
plot(modcv)

#Shrinked model with selected lambda
mods <- glmnet(X, y, family="gaussian"
               ,alpha=1,nlambda=100,lambda.min.ratio=ifelse(n<p,0.01,0.0001)
               ,standardize=T,intercept=T
               ,type.gaussian="covariance"
               ,standardize.response=F)
plot(mods,xvar="lambda")  #|"norm"|"dev"
print(mods)
betanorm = colSums(abs(mods$beta))
```

```
print(cbind(mods$lambda,mods$df,mods$dev.ratio,betanorm))
#lambda = modcv$lambda.min
#lambda = exp(0.3)   #mes
lambda = exp(1.0)   #cuatrim
beta <- predict(mods,X,s=lambda,type="coefficients")
beta <- as.numeric(beta)
names(beta) <- c("beta0",colnames(X))
write.table(beta,file="beta.csv",sep=",")

#Unrestricted model with selected variables
beta <- beta[abs(beta[])>0.000001]
Xr <- X[,colnames(X) %in% names(beta)]
mod <- lm(y~.,data=as.data.frame(cbind(y,Xr)))
summary(mod)
#cor(Xr)
vif(mod)
```

8. Bibliografía

Calzada, J., & Corina, S. (2017). Argentina en el mercado mundial de granos y subproductos. *Bols. Comer. Rosario. Inf. Sem. Año*, 5-7.

Calzada, J., & Treboux, J. (2019). Importancia económica del sector agropecuario y agroindustrial en la República Argentina. *Bols. de Comer. Rosario Inf. Sem.*

Curtin, R. (2007). Consumer sentiment surveys: worldwide review and assessment. Journal of business cycle measurement and analysis, 2007(1), 7-42.

Das, S., & Chen, M. (2001, July). Yahoo! for Amazon: Extracting market sentiment from stock message boards. In *Proceedings of the Asia Pacific finance association annual conference (APFA)* (Vol. 35, p. 43).

Fusco, M., Pederiva, L., & Barelli, E. (2017). Índice de confianza de los empresarios agropecuarios en Argentina. *Revista de Investigación en Modelos Financieros*, *1*, 1-16.

Katona, G. (1951). Psychological Analysis of Economic Behavior.

Katona, G. (1953). Rational behavior and economic behavior. *Psychological review, 60(5), 307.*

Mäntylä, M. V., Graziotin, D., & Kuutila, M. (2018). The evolution of sentiment analysis—A review of research topics, venues, and top cited papers. *Computer Science Review*, *27*, 16-32.

Mayer-Schönberger, V., & Cukier, K. (2013). *Big data: la revolución de los datos masivos*. Turner.

OECD (2019), Políticas Agrícolas en Argentina, *OECD Publishing, Paris*, https://doi.org/10.1787/9789264311879-es.

Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. *arXiv preprint cs/0205070*.

Peña, Daniel (2010). Análisis de series temporales. *Alianza Editorial*.

Sayce, D. (2019). Number of tweets per day 2019. https://www.dsayce.com/social-media/tweets-day/

Tong, R. M. (2001, September). An operational system for detecting and tracking opinions in on-line discussion. In *Working Notes of the ACM SIGIR 2001 Workshop on Operational Text Classification* (Vol. 1, No. 6).

Turney, P. D. (2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. *arXiv preprint cs/0212032*.

Webster, J. (2012). Big Data Deserves IT's Attention. Computersworld. https://www.computerworld.com/article/2550148/big-data-deserves-it-s-attention.html