Sistemas de Recomendación

Alejandra Gpe. Esquivel Guillén
alejandraeg9899@gmail.com

7 de noviembre de 2015

1. Objetivos

Este trabajo busca mostrar un sistema de recomendación basado en clasificación, y como su implementación puede mejorar la cantidad de información que es mostrada en los portales web de los negocios inmobiliarios. Presentamos un caso de estudio donde usamos el algoritmo de vecinos cercanos (KNN) para un conjunto de datos inmobiliarios, donde se busca obtener una matriz que sintetize las caracteristicas de las propiedades inmoviliarias.

Los sistemas de recomendación(SR) son componentes cruciales para los negocios en linea, con ellos se busca incrementar el número de ventas diarias. Mostraremos que el conocimiento de estos sistemas, son una de las tendencias más importantes de la computación de nuestro tiempo.

2. Introducción

Los SR aparecen a mediados los 80 de manera natural, por los vendedores que buscaban entender lo que sus clientes deseaban comprar, se generaban catálogos impresos y se entregaban con cupones dentro, en los hogares de los clientes. Con la entrada del internet y el auge del correo electrónico se inició el acercamiento usando listas de correo, muy pronto se vio la necesidad de sumar clientes o visitas de ellos al sitio web corporativo del negocio para dar variedad al catálogo. Gracias a los avances en los algoritmos de minería de datos se observó que algunas herramientas estadísticas podían usarse para anticiparse a las preferencias si conocíamos el perfil de otros usuarios con características similares. Fue así que iniciaron como simple observación de las preferencias de los clientes y en un inicio sólo se ofrecían listas de los artículos más comprados por otros usuarios de temporada. Con frecuencia los individuos confiaban en las recomendaciones ofrecidas por otros; de la misma manera que alguien te recomienda un libro o una película. La escala de evaluación comenzó en la mayoría de los productos con una escala fija (A = bueno, B = regular, C = malo) en donde se establece el valor de cierta variable (calidad, precio, disponibilidad). Los sistemas de recomendación hoy en día juegan un rol importante en todos los sitios web. La meta de ellos es incrementar las ventas, visitas, variedad de contenidos y presentar experiencias de usuario personalizadas ofreciendo sugerencias para artículos desconocidos potencialmente interesantes para un usuario.

Los SR actuales utlizan algoritmos avanzados y se invierten grandes recuersos en su desarrollo.

El interés en esta área permanece alto debido a que constituye un problema rico en investigación y a la abundancia de aplicaciones prácticas que ayuden a los usuarios a lidiar con sobrecarga de información.

Las grandes compañías de medios fueron las primeras en invertir en SR. En 2006 Netflix lanzó una competencia para mejorar su entoces sistema de recomendación en al menos un 10 % de efectividad.

El rol clave de los sistemas de recomendación es que se basan en mineria de datos, rama de la computación que se encuentra en auge y que utiliza algoritmos de inteligencia artificial obteniendo conocimiento a partir de la información.

Sin embargo, a pesar de todos estos avances, la actual generación de sistemas de recomendación evolucioana rapidamente presentando retos imposibles de resolver por la arquitectura computacional requerida como consecuencia esto a llevado a una competencia entre las empresas por vender la 'mejor solución'. Lo que se busca al diseñar un SR es que se adapte a la inmensa cantidad de información generada por los usuarios optimizando la memoria requerida. Los SR se aplican a un rango amplio de casos como recomendaciones vacacionales, ciertos tipos de servicios bancarios o de financiamiento a inversionistas, y productos a ser vendidos en una tienda creada por un 'carrito inteligente'. Estas mejoras incluyen mejores métodos para representar comportamiento y la información acerca de los artículos ha ser adquiridos, métodos avanzados de recomendación, incorporación de información contextual y utilización de ratings multicriterio, además del desarrollo de métodos menos intrusivos que también se apoyan en métricas para determinar desempeño de los sistemas de recomendación.

- La recolección de preferencias de los usuarios: **No tiene nada que ver con los perfiles de usuario** ya que esto se realiza a través de una encuesta que permite conocer las preferencias de los usuarios, algunos de ellos mencionan las características deseables de un artículo específico.
- Análisis: **En esta etapa se detectan patrones en las opciones seleccionadas por los usuarios. -**
- Generación de opciones: Los SR se modifican continuamente debido a que el usuario interacciona con el catalogo de artículos y el SR debe adaptarse dinámicamente a dichos cambios.
- Artículos Recomendados: Los artículos pueden ser en general cualquier bien o servicio requerido por un usuario específico. No se requiere que el usuario tenga experiencia previa con el uso del sistema principal. Sin embargo sus selecciones son tomadas en cuenta para mejorar la precisión de la recomendación próxima.

2.1. Carácteristicas clave que un SR debería cumplir

- **Incrementar el número de artículos vendidos**: Debería ser capaz de vender un conjunto de artículos de modo que puedan ser comprados sin la intervención de los SR, es decir puede tener su propia meta de venta (** ningún visitante se puede ir sin comprar **).
- **Vender artículos diversos**: Se prefiere la diversidad de artículos al ofertar productos ya que las empresas buscan que los usuarios (clientes) detecten productos en los que ni siquiera han pensado adquirir. Con frecuencia

se dan descuentos o rebajas en ellos lo que ocasiona que las recomendaciones de los usuarios impacten su venta.

- **Incrementar la satisfacción del usuario**: Un SR bien diseñado cambia la interfaz de usuario según las preferencias de los mejores clientes, ofreciendo objetivos resaltados y posibilidad de que en base a los cambios de la interfase se crean grupos de interés para ofertar productos.
- **Mejor entendimiento de lo que el usuario quiere**: El sondeo adecuado de las preferencias del usuario, permite afinar los parámetros del SR con el fin de acertar en el "mejor" producto.
- **Incrementar la fidelidad del usuario**: La interacción por parte del usuario con el sitio permite que la información sea dinámica (contenido que mantenga la atención) con frecuencias las sugerencias y reseñas de un producto mantienen al usuario mas tiempo en el sitio lo que se aprovecha dando mas opciones de compra.

2.2. Clasificación de los SR

Los SR usualmente son clasificados en las siguientes categorías:

- **Recomendaciones Basadas en contenido**: Al usuario le serían recomendados artículos similares a los que selecciona en el pasado.
- **Recomendaciones Colaborativas**: Al usuario le serían recomendados artículos que gustan a las personas con preferencias y gustos similares en el pasado.
- **Aproximación Híbrida**: Estos métodos combinan métodos colaborativos y basados en contenido.

Adicionalmente los sistemas de recomendación que predicen valores absolutos de rating que usuarios individualmente no han marcado aun en artículos no conocidos, se les conoce como *filtrado basado en preferencias* .

3. Importancia de los SR

El interés de las empresas y universidades en los SR se basa en que constituye un problema rico en investigación, debido la abundancia de aplicaciones prácticas que ayuden a los usuarios a lidiar con la sobrecarga de información; y además el beneficio económico que se puede obtener de ellas.

Las grandes compañías de medios fueron las primeras en invertir en máquinas de aprendizaje comerciales. En 2006, Netflix anunció su máquina de aprendizaje y la competencia de minería de datos Netflix Prize con 1 millón de dólares para el equipo que logrará mejorar su entonces recomendador, el cual fue reclamado en 2009, con toda la atención de los medios, lo que se conoció como 'Recomendaciones de Netflix: Más allá de las 5 estrellas', reveló que la ciencia de los datos (Science Data) puede ser una grán inversión para cualquier negocio en linea y que las ciencias de la computación en la rama de aprendizaje automático tienen algoritmos que pueden ser adaptados a sistemas comerciales. La meta de Netflix

Prize fue fondear un algoritmo de recomendaciones que pudiera entregar 10 % de mejora en precisión de predicción sobre el sistema existente para ello utilizaron herramientas conocidas como Deep Learning. Apple basa su sistema de recomendaciones de estrenos en el sistema de crítica Rotten Tomatoes. Google Play Store en un sistema de ranking de aplicaciones.

Sin embargo, a pesar de todos estos avances, la actual generación de sistemas de recomendación aún requieren mejoras para realizar métodos más efectivos y que sean aplicables a un rango amplio de casos, tales como recomendaciones vacacionales, servicios bancarios o de financiamiento a inversionistas, y productos a ser vendidos en una tienda creada por un "carrito inteligente". Estas mejoras incluyen métodos para representar el comportamiento de los usuarios y la información acerca de los artículos que van a ser adquiridos; métodos avanzados de recomendación; incorporación de información contextual (Ubicación Geográfica, Usuarios satisfechos del producto, etc) y utilización de ratings multicriterio. Además se deben desarrollar métodos no intrusivos que también se apoyan en métricas para determinar desempeño de los sistemas de recomendación.

4. Antecedentes

Las raíces de los sistemas de recomendación inician con trabajos en ciencia cognitiva, recuperación de información y algunas conexiones con administración científica, emergen como un área independiente a mediados de 1990 cuando los investigadores se enfocan en problemas de recomendación que explícitamente se basaban en una estructura de rating. Intuitivamente, esta estimación es usualmente basada en la escala definida por un usuario acerca de una breve información. A partir del rating de algunos artículos se puede determinar el rating de algunos que no han sido seleccionados, con el **rating superior estimado** . De manera formal el problema de recomendación puede ser formulado como sigue: Sea C el conjunto de todos los usuarios y sea S el conjunto de los posibles artículos que pueden ser recomendados tales como libros, películas o restaurantes. El espacio S de los posibles artículos puede ser muy amplio, alcanzando los cientos de millones de artículos. Similarmente el espacio del usuario puede ser bastante amplio. Sea u la función de utilidad que mide el beneficio de un articulo s al usuario. De modo que $C \times S \rightarrow R$, donde R es la totalidad de un conjunto ordenado. Entonces, para cada usuario $c \in C$, queremos seleccionar tal $s' \in S$ que maximiza la utilidad del usuario. De manera simplificada tenemos que: $\forall c \in C, s' = argmaxu(c, s)$

En un sistema de recomendación la utilidad de un artículo es usualmente representada por un *rating* el cual indica como a un usuario le gusta un artículo en particular. Por ejemplo: Juan Perez le dio a "Harry Potter" el rating de 7 (en escala de 1 a 10).

Ratings. Rotten Tomatoes (Tomatómetro): El rating del tomatómetro se basa en las opciones publicadas por críticos de cine y televisión, es una medida confiable de la calidad de una película y representa el porcentaje de reseñas positivas dadas a una película.

Filtrado Colaborativo: La idea detrás del filtrado colaborativo es que se pueden usar los rating de los usuarios que comparten gustos similares para predecir los que aún no han sido definidos. Para obtener intuición, se comparan los ratings por pares del usuario

4.1. Ejemplos de SR:

- Airbnb. Sitio de recomendación de hospedaje. Promueve el hospedaje en casas o departamentos de particulares que ofrecen habitaciones a bajo precio donde además de mostrar ubicaciones disponibles por fecha y ubicación preferida, se incluye información de reseñas de clientes previos y se mezclan con comentarios de redes sociales relacionadas al perfil del usuario.

- Yelp. Recomendación de restaurantes. Los usuarios publican reseñas de sitios como: Restaurantes, Tiendas, Servicios (Taxi, Tintoreria, Lavanderia) y van construyendo confianza en los proveedores o vendedores de cierto bien, después se publican en un portal donde se localizan ubicandolos por ubicación cercana al cliente.

- Los SR de grandes empresas como Google Play, Apple Movies utilizan un sistema de ranking para sugerir aplicaciones. Un sistema de ranking puede utilizar elementos como el número de descargas, el tamaño de la aplicación, y la ubicación geográfica del vendedor para mostrar en primer lugar las aplicaciones que tienen la misma categoría y que han sido seleccionadas por otros usuarios en compras previas. De manera similar al ranking de las busquedas de google, se cuentan las palabras clave (keywords) y el numero de links a un sitio específico. De ahi que sólo se muestran los primeros 10 sitios más visitados.

5. Descripción del Problema

Se considerá una empresa dedicada a la venta y renta de inmuebles, cuyos canales de venta són entrevistas y llamadas telefónicas. Ha decidido invertir en un sitio web donde publica el listado de propiedades a ofertar. Como en la mayoria de sitios web, el diseño se centra únicamente en un portal informativo. Con frecuencia buscando llamar la atención del usuario utilizando elementos dinámicos basados en imágenes o videos, de tal manera que no es posible identificar información relevante o es imposible realizar una búsqueda en texto para localizar un artículo especifico. Por consecuencia los clientes que visitan el sitio (ver info de visitas) observan que resulta muy complicado localizar alguna propiedad relevante, o simplemente toma demasiado tiempo navegar pagina a página para encontrar lo que se requiere.

Este ejemplo es el caso de la mayoria de negocios mexicanos que utilizan un portal web para anunciarse y no aprovechan la interacción con sus clientes. Dentro de la ciudad de Morelia, se han detectado poco más de 30 inmobiliarias que utilizan portales para promoción, incluso las principales constructoras (Arko, Habicasa) utilizan portales de tipo informativo. Muy pocos son los que realizan alguna encuentas o registro para conocer a sus clientes. Por lo que este proyecto busca incrementar la cantidad de información que un usuario promedio recibe al consultar una propiedad. Para ello se incluyen sugerencias de propiedades similares con el fin de mantener al usuario más tiempo en el sitio y por consecuencia incrementar la utilidad de la información. Adicionalmente para mejorar las recomendaciones, durante la navegación se buscará conocer las preferencias del usuario utilizando breves cuestionarios sobre la información que se

está consultando para identificar potenciales opciones.

5.1. Metodología

Nuestra propuesta de diseño se centra en 4 puntos.

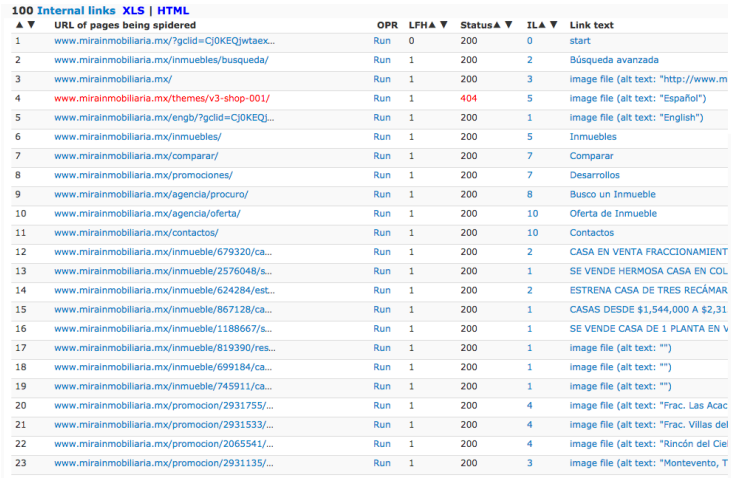
1. Recolección de Datos.
2. Filtrado y Clasificación de los datos.
3. Proporcionar recomendaciones cercanas a las seleccionadas por el usuario.
4. Retroalimentación a través de encuestas de satisfacción al cliente.

5.2. Recolección de los datos

Para la recolección, seleccionamos 20 inmobiliarias de la ciudad de Morelia, algunos portales no presentaban listados accesibles y se descartaros. Del conjunto del que era posible consultar propiedades a traves de una url directa se decidió utilizar un crawler público[6] que nos permitiera indexar las paginas web de las inmobiliarias, para obtener una lista de links (ver figura 1), se filtro la lista dejando unicamente las que se refieren a propiedades; almacenando posteriormente los documentos en formato HTML.

5.3. Filtrado y Clasificación

Se extraerá el cuerpo principal de los documento usando el lenguaje de programación python, almacenado cada propiedad en un nuevo conjunto de datos que convertiremos a un archivo csv para posteriomente realizar la clasificación de las propiedades. Es importante destacar que para este proyecto no se requiere almacenar en una base de datos relacional la información, debido a que trabajar con texto plano simplifica al análisis y no se requiere interacción por parte del usuario.



100 Internal links XLS | HTML

▲▼	URL of pages being spidered	OPR	LFH▲▼	Status▲▼	IL▲▼	Link text
1	www.mirainmobiliaria.mx/?gclid=Cj0KEQjwtaex...	Run	0	200	0	start
2	www.mirainmobiliaria.mx/inmuebles/busqueda/	Run	1	200	2	Búsqueda avanzada
3	www.mirainmobiliaria.mx/	Run	1	200	3	image file (alt text: "http://www.m
4	www.mirainmobiliaria.mx/themes/v3-shop-001/	Run	1	404	5	image file (alt text: "Español")
5	www.mirainmobiliaria.mx/engb/?gclid=Cj0KEQj...	Run	1	200	1	image file (alt text: "English")
6	www.mirainmobiliaria.mx/inmuebles/	Run	1	200	5	Inmuebles
7	www.mirainmobiliaria.mx/comparar/	Run	1	200	7	Comparar
8	www.mirainmobiliaria.mx/promociones/	Run	1	200	7	Desarrollos
9	www.mirainmobiliaria.mx/agencia/procuro/	Run	1	200	8	Busco un Inmueble
10	www.mirainmobiliaria.mx/agencia/oferta/	Run	1	200	10	Oferta de Inmueble
11	www.mirainmobiliaria.mx/contactos/	Run	1	200	10	Contactos
12	www.mirainmobiliaria.mx/inmueble/679320/ca...	Run	1	200	2	CASA EN VENTA FRACCIONAMIENT
13	www.mirainmobiliaria.mx/inmueble/2576048/s...	Run	1	200	1	SE VENDE HERMOSA CASA EN COL
14	www.mirainmobiliaria.mx/inmueble/624284/est...	Run	1	200	2	ESTRENA CASA DE TRES RECÁMAR
15	www.mirainmobiliaria.mx/inmueble/867128/ca...	Run	1	200	1	CASAS DESDE $1,544,000 A $2,31
16	www.mirainmobiliaria.mx/inmueble/1188667/s...	Run	1	200	1	SE VENDE CASA DE 1 PLANTA EN V
17	www.mirainmobiliaria.mx/inmueble/819390/res...	Run	1	200	1	image file (alt text: "")
18	www.mirainmobiliaria.mx/inmueble/699184/ca...	Run	1	200	1	image file (alt text: "")
19	www.mirainmobiliaria.mx/inmueble/745911/ca...	Run	1	200	1	image file (alt text: "")
20	www.mirainmobiliaria.mx/promocion/2931755/...	Run	1	200	4	image file (alt text: "Frac. Las Acac
21	www.mirainmobiliaria.mx/promocion/2931533/...	Run	1	200	4	image file (alt text: "Frac. Villas del
22	www.mirainmobiliaria.mx/promocion/2065541/...	Run	1	200	4	image file (alt text: "Rincón del Cie
23	www.mirainmobiliaria.mx/promocion/2931135/...	Run	1	200	3	image file (alt text: "Montevento, T

Figura 1: Listado de Artículos relacionados con propiedades

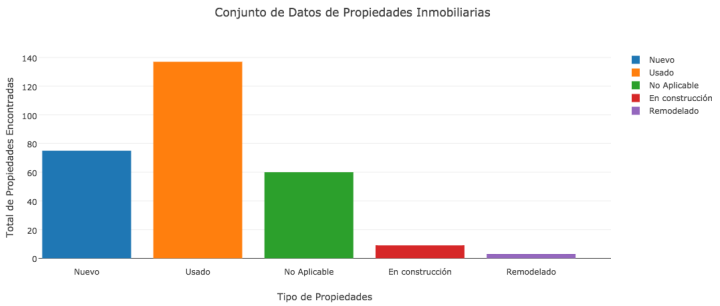


Figura 2: Distribución de Propiedades Tipo de Propiedad

	Total
Bodega	5
Casa	168
Departamento	24
Edificio	1
Local	13
Oficinas	1
Terreno	72

Figura 3: Tabla de Propiedades por Inmueble

5.4. Clasificación de las propiedades

Para el procesamiento y clasificación de los datos se utilizara el lenguaje de programación R. Las propiedades que descargamos de los portales web se integrarán en un solo listado con aproximadamente 284 registros, las cuales vienen listadas por Estado de la propiedades (Usado, Nuevo, Construcción), Área construida (m^2), Zona (Colonia o barrio de Referencia), Precio, Latitud, Longitud y el tipo de propiedad (ver figura 4).

El algoritmo de clasificación es el de vecinos cercanos (KNN), ya que estamos trabajando con variables categóricas. Nuestro objetivo es ofrecer opciones con carácteristicas similares dentro de cada categoria correspondiente.

Latitud	Estado	Superficie Uti
19.66721110000	Nuevo	60
19.67036430000	Usado	199.36
19.6599095000%	No Aplicable	360
19.60531230000	No Aplicable	1.25 ha
19.65956219969	Usado	140
19.56690721050	No Aplicable	160
19.73287613516	Nuevo	151
19.67198296710	Nuevo	118
19.70408386499	Usado	350
19.70945930000	Usado	110

Figura 4: Dataset de Propiedades Inmobiliarias

5.5. KNN - Nearest Neighbor

También conocido como (vecino cercano); es el más simple de los algoritmos de aprendizaje máquina que existe y el más utilizado si se trabaja con variables categóricas. Se basa en identificar los k registros en el conjunto de entrenamiento que son *cercanos* en similaridad. La prueba asigna una clase a la mayoria de los vecinos cercanos similares. El KNN trata las características como coordenadas multidimensionales. Después tomaremos uno o más registros del conjunto de prueba, los cuales serán las selecciones del usuario.

5.6. Métrica de similaridad con distancia.

El KNN requiere una función distancia, o una métrica que mida la similaridad entre dos instancias. El KNN utiliza la distancia euclideana, que es la distancia de un punto con respecto a otro conectados por una línea recta. La distancia euclideana se expresa de la siguiente manera:

$$dist(x, y) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} \tag{1}$$

El KNN requiere elegir una cantidad de vecinos cercanos (K) y determinar que tan bien el modelo clasifica nuevos valores. El balance entre sobre ajustar o subajustar el valor de K para el conjunto de entrenamiento es un problema conocido como **ajuste de la varianza**. En la figura 5 se observa que si se elige una K muy grande el conjunto se parte dejando algunos puntos clasificados incorrectamente, lo cual se toma como falsos positivos; en cambio si la K se toma muy pequeña se tienen falsos falsos. La seleccion de la K se considera en base a la tolerancia de error (la queda a elección del programador) que se pueda perimitir el clasificador, ajustando así hasta que se encuentra una K entre estos dos extremos.

Cuando se seleccionan los valores de K el impacto de la varianza causa datos ruidosos, los cuales pueden ser incluidos o puede sean ignorados por el sistema.

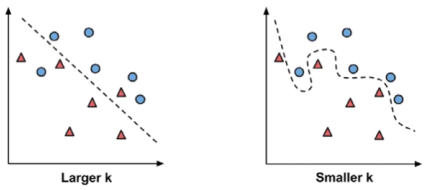


Figura 5: Efectos de seleccionar diferentes valores de la K en KNN

	Casas_test_pred				
Casas_test_labels	Casa	Departamento	Local	Terreno	Row Total
Bodega	0	0	1	0	1
	0.000	0.000	1.000	0.000	0.012
	0.000	0.000	0.250	0.000	
	0.000	0.000	0.012	0.000	
Casa	36	4	0	8	48
	0.750	0.083	0.000	0.167	0.571
	0.837	0.500	0.000	0.276	
	0.429	0.048	0.000	0.095	
Departamento	2	1	1	0	4
	0.500	0.250	0.250	0.000	0.048
	0.047	0.125	0.250	0.000	
	0.024	0.012	0.012	0.000	
Local	0	3	1	0	4
	0.000	0.750	0.250	0.000	0.048
	0.000	0.375	0.250	0.000	
	0.000	0.036	0.012	0.000	
Oficinas	0	0	1	0	1
	0.000	0.000	1.000	0.000	0.012
	0.000	0.000	0.250	0.000	
	0.000	0.000	0.012	0.000	
Terreno	5	0	0	21	26
	0.192	0.000	0.000	0.808	0.310
	0.116	0.000	0.000	0.724	
	0.060	0.000	0.000	0.250	
Column Total	43	8	4	29	84
	0.512	0.095	0.048	0.345	

Figura 6: Propiedades recomendadas según la selección del Usuario

6. Resultados

A continuación se describe en lenguaje R el algoritmo utilizado para la clasificación utilizando KNN.

```
#Clasificador de Casas segun la eleccion
CasasCompletas <- read.csv("DataSet1.csv",header=TRUE)
View(CasasCompletas)
#casasClasificar <- CasasCompletas[,c(1,4,5,7,8,9)]
#View(casasClasificar)
CasasCompletas$Tipo <- factor(CasasCompletas$Inmueble,
             levels=c("Casa","Departamento","Oficinas",
             "Local","Terreno","Bodega"),
             labels=c("Casa","Departamento","Oficinas",
             "Local","Terreno","Bodega"))
CasasCompletas$Estado <- factor(CasasCompletas$Estado,
             levels=c("Nuevo","Usado","No Aplicable",
             "En construccion","Remodelado"),
             labels=c("Nuevo","Usado","T", "PC","RM"))
CasasCompletas$Oferta <-factor(CasasCompletas$Oferta,
             levels=c("renta","venta"),
             labels=c("renta","venta"))
CasasCompletas$Inmueble <- factor(CasasCompletas$Inmueble,
```

```
             levels = c("Casas","Departamento", "Habitacion",
             "Oficina","Local","Terreno","Edificio"))


round(prop.table(table(CasasCompletas$Estado))*100,digits = 1)

normalizacion <- function(x) {
  return ((x - min(x)) / (max(x) - min(x)))
}

CasasNorm <- as.data.frame(
            lapply(CasasCompletas[c(3,5,6)],normalizacion))

# Clasificacion por KNN

Casas_train <- CasasNorm[1:200,]
Casas_prop <- CasasNorm[201:284,]
Casas_train_labels <- CasasCompletas[1:200,1]
Casas_test_labels <- CasasCompletas[201:284,1]
library('class')
library('gmodels')

Casas_test_pred <- knn(train=Casas_train,test=Casas_prop,
                       cl=Casas_train_labels,k = 6)
Casas_test_pred

# La tabla cruzada nos permite comparar como se clasificaron
# los datos de prueba
CrossTable(x = Casas_test_labels,
        y = Casas_test_pred, prop.chisq = FALSE)
```

De la figura 6 las celdas que caen en la diagonal contienen el numero de registros donde el KNN encontró propiedades que son similares con la muestra seleccionada. Se puede observar que el conjunto de propiedades sugeridas se obtiene al seleccionar una K con valor de 6, basándonos en que tenemos 6 posibles clases distintas. Probamos distintos valores de K para refinar la consulta. Al implementar este algoritmo utilizamos una función de normalización para que los datos se describan en la misma escala [0-1], el KNN se ve afectado si la varianza es diferente en cada variable, afectando el resultado final.

7. Conclusiones

La aplicación de KNN a nuestro conjunto de Datos mostró que es posible sugerir cualquier documento relacionado, esto puede aplicarse a diversos dominios. Una de las desventajas que usar KNN es que el conjunto de entrenamiento requiere ser amplio, en nuestro caso reducimos de 400 a 200 debido a que teniamos campos faltantes o valores vacios en algunos casos. Por desgracia KNN no puede manejar valores ausentes o vacios.Para perfeccionar la clasificación es posible utilizar también Arboles de clasificación que son mucho más flexibles y

no requieren que los datos sean normalizados. Este proyecto encuentra viable implementarlo en un PlugIn. El cual puede ser aceptado por los distintos Manejadores de Contenido del Mercado Web (Wordpress, Joomla, Drupal). Para futuros desarrollos dejamos la posibilidad de utilizar python en todo el proceso.

8. Bibliografía

[1] Adomavicius, G., and A. Tuzhilin. "Toward the next Generation of Recommender Systems: A Survey of the State-of-the-art and Possible Extensions."

IEEE Trans. Knowl. Data Eng. IEEE Transactions on Knowledge and Data En- gineering: 734-49. Print.

[2] Sauter, Vicki Lynn, and Vicki Lynn Sauter. Decision Support Systems for Business Intelligence. 2nd ed. Hoboken, N.J.: Wiley, 2010. Print.

[3] R.Bell, Y. Koren and C. Volinsky. The BellKor 2008 Solution to the Netflix Prize. 2008

[4] Said, Alan, Brijnesh J. Jain, Sascha Narr, and Till Plumbaum. "Users and Noise: The Magic Barrier of Recommender Systems." User Modeling, Adaptation, and Personalization Lecture Notes in Computer Science: 237-48. Print

[5] Lantz, B. (n.d.). Machine learning with R: Learn how to use R to apply powerful machine learning methods and gain an insight into real-world applications.

[6] Harrington, P. (2012). Machine learning in action. Shelter Island, NY: Manning Publications.

[7] Toomey, D. (n.d.). R for data science.

[8] Teutonico, D. (2015). Ggplot2 essentials explore the full range of ggplot2 plotting capabilities to create meaningful and spectacular graphs. Birmingham, England: Packt Publishing.

[9] Leipzig, J., & Li, X. (2009). Data mashups in R. Sebastopol, Calif.: O'Reilly.

[10] Ricci, F. (2011). Recommender systems handbook. New York: Springer.

[11] Russell, S., & Norvig, P. (1995). Artificial intelligence: A modern approach. Englewood Cliffs, N.J.: Prentice Hall.

[12] Website Crawler Tool and Google Sitemap Generator
http://freetools.webmasterworld.com/tools/crawler-google-sitemap-generator/