FACULTAD DE INGENIERÍA

UNIVERSIDAD DE BUENOS AIRES

75.50 – INTRODUCCIÓN A LOS SISTEMAS INTELIGENTES

TRABAJO PRÁCTICO:

1er cuatrimestre 2018

93115 – Capon Paul, Lucia Tamara

93158 – Graffe, Fabrizio Sebastian

Indice

1. Introducción

El objetivo del trabajo presentado es aplicar la metodología CRISP sobre la base de datos de postulantes a empleos a través de una plataforma online, realizando un análisis sobre los datos. Ésta base de datos es provista por el sitio Kaggle[1].

2. Marco teórico

2.1. CRISP

La metodología **CRISP** se estructura en seis fases o etapas:

2.1.1. Entendimiento del negocio

Objetivos y requerimientos desde una perspectiva no técnica. Es probablemente la más importante y aglutina las tareas de comprensión de los objetivos y requisitos del proyecto desde una perspectiva empresarial o institucional, con el fin de convertirlos en objetivos técnicos y en un plan de proyecto

2.1.2. Entendimiento de los datos

Comprende la recolección inicial de datos, con el objetivo de establecer un primer contacto con el problema, familiarizándose con ellos, identificar su calidad y establecer las relaciones más evidentes que permitan definir las primeras hipótesis

2.1.3. Preparación de los datos

Una vez efectuada la recolección inicial de datos, se procede a su preparación para adaptarlos a las técnicas de Data Mining que se utilicen posteriormente, tales como técnicas de visualización de datos, de búsqueda de relaciones entre variables u otras medidas para exploración de los datos. Esta preparación incluye las tareas generales de selección de datos a los que se va a aplicar una determinada técnica de modelado, limpieza de datos, generación de variables adicionales, integración de diferentes orígenes de datos y cambios de formato.

2.1.4. Modelado

Se seleccionan las técnicas de modelado más apropiadas para el proyecto de Data Mining específico

2.1.5. Evaluación

En esta fase se evalúa el modelo, teniendo en cuenta el cumplimiento de los criterios de éxito del problema. Debe considerarse además, que la fiabilidad calculada para el modelo se aplica solamente para los datos sobre los que se realizó el análisis

[1] https://www.kaggle.com/

2.1.6. Explotación

Una vez que el modelo ha sido construido y validado, se transforma el conocimiento obtenido en acciones dentro del proceso de negocio, ya sea que el analista recomiende acciones basadas en la observación del modelo y sus resultados, ya sea aplicando el modelo a diferentes conjuntos de datos o como parte del proceso

Las fases se pueden apreciar en la siguiente imagen:

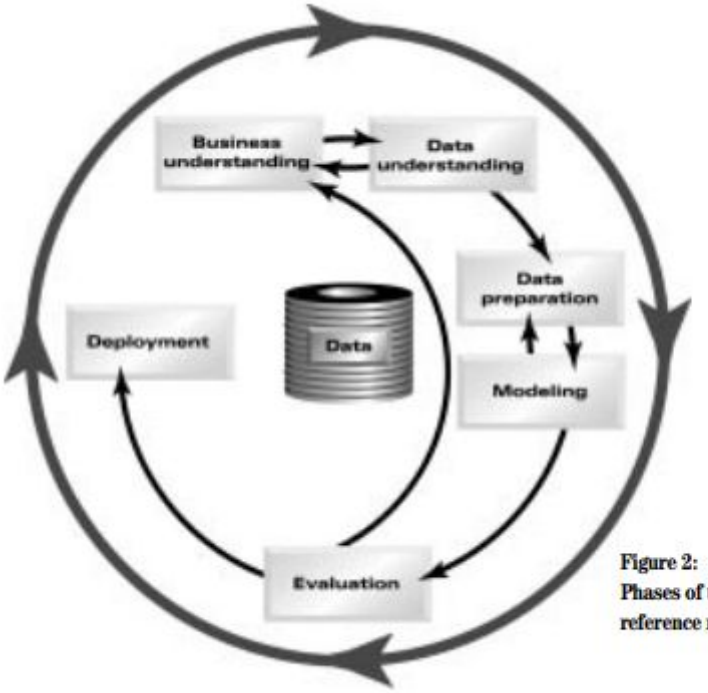


Figure 2:
Phases of the CRISP-DM reference model

CRISP-DM 1.0 Step-by-step data mining guide

3. Fase 1: Entendimiento del negocio

3.1. Determinar los objetivos del negocio

3.1.1. Contexto

Navent es la empresa dueña de los portales de búsquedas laborales Boomerang, ZonaJobs, UniversoBit, HiringRoom, Laborum, Multitrabajos, Konzerta. Uno de los principales desafíos que enfrenta la empresa es identificar cuales avisos son más relevantes para cada usuario que visita sus portales, de manera de aumentar la conversión de usuarios a postulantes.

3.1.2. Objetivos del negocio

Aumentar la conversión de usuarios a postulantes, mediante la detección temprana del interés de un usuario a cierta oferta laboral.

3.1.3. Criterios de éxito del negocio

Como criterio de éxito del negocio, se define la capacidad de reconocer con un 70% de exactitud, según las métricas de accuracy y F1 score (definidas mas adelante), si un postulante se postulará a cierta oferta laboral, a partir de los datos del postulante, y del aviso en sí.

3.2. Evaluación de la situación

3.2.1. Inventario de recursos

- Data set con los datos sobre avisos de trabajo, postulantes, y las postulaciones efectivas.
- Hardware necesario para ejecutar en tiempo razonable los algoritmos de minería de datos.
- Software adaptado a dichos algoritmos, preparado para ingresar parámetros fijos y ejecutarse sobre la base de datos.
- Profesionales capacitados en el uso de dichos algoritmos.

3.2.2. Requerimientos, supuestos y restricciones

Requerimientos:

- Debe aplicarse la metodología CRISP y, al finalizar el proyecto, entregarse la documentación que se obtuvo como resultado.

Supuestos

- Los datos son actuales.
- Los datos son reales.

Restricciones

- La base de datos nombrada anteriormente debe ser la única fuente de información utilizada.

3.2.3. Riesgos y contingencias

Riesgos

- Ante el hipotético caso de que los datos almacenados en la base no resultaren válidos al compararlos con la realidad, entonces los resultados obtenidos del presente trabajo no serían válidos.

Contingencias

- En dicho caso se procederá a recolectar nuevos datos, de calidad confiable, y de no ser posible se tomará la decisión de abandonar el proyecto.

3.2.4. Terminología

A continuación se define la terminología a utilizar, tanto respectiva al negocio involucrado como al proceso de minería de datos.

Términos	Definición
Dataset	El conjunto de datos (o dataset) es la fuente de información que se explotará en un proceso de minería de datos, a partir de la metodología CRISP descrita en la introducción. Dicho set de datos está compuesto por atributos, que representan las distintas dimensiones o características de la información, y registros.
Registro	Es una observación o unidad de información, que cuenta con un valor para cada uno de los atributos.
Postulante	Usuario que accede al sitio en búsqueda de una oferta laboral a

	la cual postularse.
Postulación	Representación de la transacción de un postulante al aplicar a cierta oferta laboral.
Aviso de trabajo	Publicación del sitio de cierta oferta laboral, a la cual pueden aplicar los postulantes.

3.2.5. Costos y beneficios

Debido a que la base de datos que utilizaremos es pública, y por lo tanto gratuita, sumado al hecho de que se utilizarán herramientas libres que no requieren de ningún tipo de hardware en particular, el costo económico del plan es ínfimo (únicamente se debería tener en cuenta el costo en horas hombre de los analistas involucrados en el proyecto).
El beneficio inicial de una mayor reputación es el respeto y la atención dentro de la comunidad, volviéndose posiblemente una autoridad en diversos temas y un pilar de ayuda entre los pares. Luego, también brinda a acceso a información valiosa para analizar el sector de mercado que son los programadores, considerando también que es un sector que tiende a crecer año tras año.

3.3. Determinar los objetivos de la minería de datos

3.3.1. Objetivos de la minería de datos

El objetivo de la minería de datos es determinar a partir de perfil de una persona (postulante) si este se postulará para cierta oferta laboral y con qué probabilidad.

3.3.2. Criterios de éxito de la minería de datos

Como criterio de éxito, se define la obtención de un modelo capaz de determinar con un al menos el 70% de exactitud si un postulante se postulará a cierta oferta laboral.

3.4. Producir el plan de proyecto

3.4.1. Plan de proyecto

A continuación se detalla, para cada una de las tareas a llevar a cabo para el presente proyecto, la cantidad de horas requeridas para su ejecución. La lista de tareas está ordenada secuencialmente en términos temporales.

a. Recolección de datos: 5 horas.
b. Preparación de datos: 10 horas.
c. Ejecución de algoritmos: 3 horas.
d. Análisis de resultados de algoritmos: 6 horas.
e. Combinación de resultados: 5 horas.
f. Elaboración de reporte: 10 horas.

3.4.2. Evaluación inicial de herramientas y técnicas

Se utilizarán herramientas, bibliotecas y software de distribución libre y gratuita:

- Python
- Sklearn
- Pandas
- PySpark

4. Fase 2: Entendimiento de los datos

4.1. Recolección de los datos iniciales

4.1.1. Informe de colección de datos inicial

El dataset seleccionado es una colección de información estructurada sobre los postulantes, avisos de trabajo y postulaciones que se encuentran en los portales de búsqueda de empleo que maneja la empresa.

- URL: **https://www.kaggle.com/c/navent/data**

4.2. Descripción de los datos

4.2.1. Informe de descripción de los datos

El conjunto de datos está compuesto por seis tablas.

- **fiuba_1_postulantes_educacion**:
  - Descripción: *datos sobre la educación de los usuarios que vieron o se postularon a un aviso.*
  - *Campos:*

Nombre	Tipo	Descripción	Valores permitidos
idpostulante	TEXT	Identificación del usuario	Sin restricc.
nombre	TEXT	Nombre nivel educativo.	Posgrado, Universitario, Master, Otro, Terciario Técnico, Doctorado, Secundario
estado	TEXT	Estado del nivel educativo	En Curso, Graduado, Abandonado

- **fiuba_2_postulantes_genero_y_edad**:

- Descripción: *datos sobre el género y edad de los usuarios de los portales.*
- Campos:

Nombre	Tipo	Descripción	Valores permitidos
idpostulante	TEXT	Identificación del usuario	Sin restricc.
fechanacimiento	TEXT	Fecha de nacimiento del postulante	Texto con formato: yyyy-MM-dd
sexo	TEXT	Sexo del usuario.	FEM, MASC

- **fiuba_3_vistas**:
  - Descripción: *Vistas sobre avisos de trabajo que realizaron los usuarios*
  - Campos:

Nombre	Tipo	Descripción	Valores permitidos
idAviso	INTEGER	Identificación del aviso	Sin restricc.
timestamp	DATE	Fecha, y hora del momento en el que el aviso fue visto por el postulante	Sin restricc.
idpostulante	TEXT	Identificación del usuario	Sin restricc.

- **fiuba_4_postulaciones**:
  - Descripción: *Postulaciones de usuarios sobre los avisos de trabajo.*
  - Campos:

Nombre	Tipo	Descripción	Valores permitidos
idaviso	INTEGER	Identificación del aviso	Sin restricc.

idpostulante	TEXT	Identificación del usuario	Sin restricc.
fechapostulacion	TEXT	Fecha de postulación del usuario al aviso.	Texto con formato: yyyy-MM-dd

- **fiuba_5_avisos_online**:
  - Descripción: *Avisos disponibles en los portales*
  - Campos:

Nombre	Tipo	Descripción	Valores permitidos
idaviso	INTEGER	Identificación del aviso	Sin restricc.

- **fiuba_6_avisos_detalle**:
  - Descripción: *Detalle de los avisos disponibles en los portales*
  - Campos:

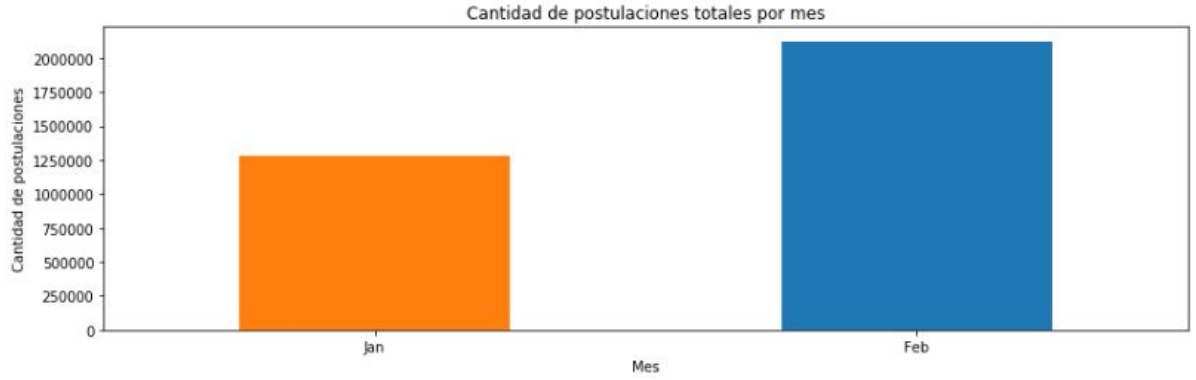
Nombre	Tipo	Descripción	Valores permitidos
idaviso	INTEGER	Identificación del aviso	Sin restricc.
idpais	INTEGER	Identificación del pais origen del aviso	1 (Argentina)
titulo	TEXT	Título del aviso	Sin restricc.
descripcion	TEXT	Descripción del aviso	Sin restricc. (en formato HTML)
nombre_zona	TEXT	Nombre de la zona donde se encuentra la empresa que realiza el aviso	Gran Buenos Aires, Capital Federal, Buenos Aires (fuera de GBA), GBA Oeste, NaN
ciudad	TEXT	Nombre de la ciudad donde se encuentra la	Buenos Aires, Capital Federal, caba, paternal,

		empresa que realiza el aviso	Argentina, San Isidro, Santa Rosa, Tortuguitas, CABA, La Plata, Zárate, Campana, Escobar, Mendoza, Barracas, República Argentina, Buenos Aires Province, Parque Patricios, Vicente Lopez, Microcentro, NaN
mapacalle	TEXT	Dirección donde se encuentra la empresa que realiza el aviso	Sin restricc. Formato recomendado nombre_calle + espacio + nro
tipo_de_trabajo	TEXT	Tipo de puesto que busca el aviso	Full-time, Part-time, Por Horas, Temporario, Fines de Semana, Pasantia, Teletrabajo, Por Contrato, Primer empleo
nivel_laboral	TEXT	Seniority que busca el aviso	Senior / Semi-Senior, Junior, Jefe / Supervisor / Responsable, Otro, Gerencia / Alta Gerencia / Dirección
nombre_area	TEXT	Área para el cual aplica el puesto del aviso	En el anexo: *Nombre de áreas laborales*
denominacion_empresa	TEXT	Denominación fiscal de la empresa	Sin restricc.

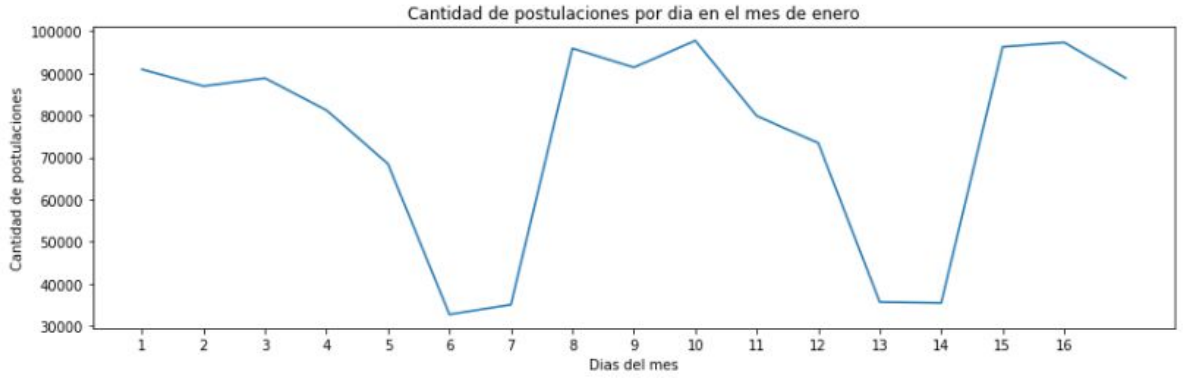
4.3. Explorar los datos

4.3.1. Informe de exploración de los datos

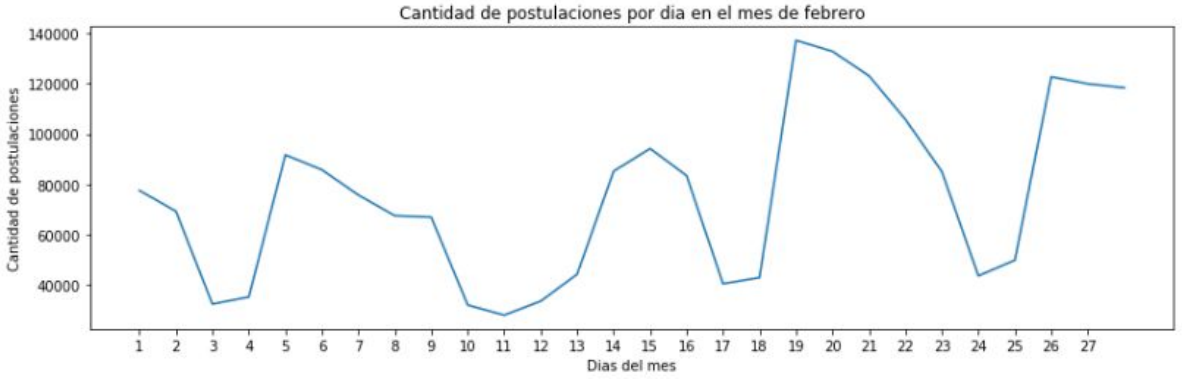
Mediante el uso de Pandas y Spark exploramos el dataset, confeccionando gráficos con el fin de comprender mejor la información provista por el data set.



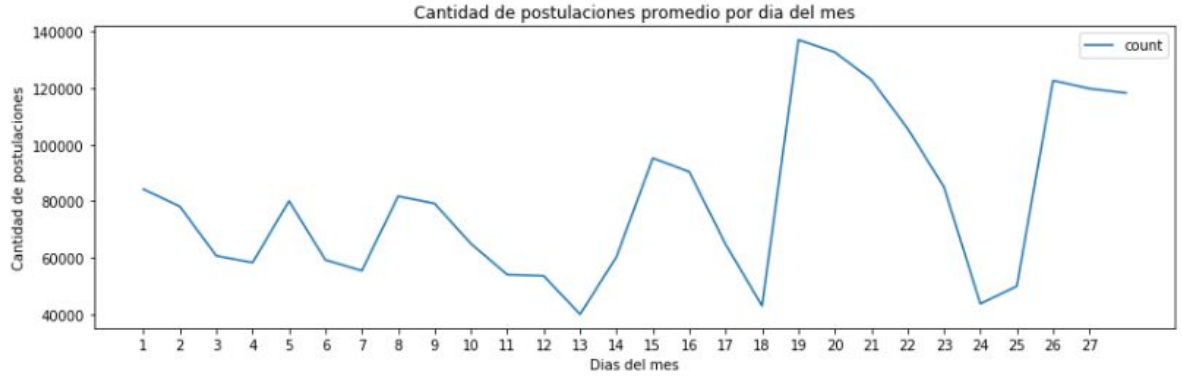
En este gráfico se puede observar que durante el mes de febrero hay una diferencia importante en la cantidad de postulaciones. Una teoría que se puede extraer de este hecho es que esta diferencia se dé debido a que durante el mes de enero hay más cantidad de personas de vacaciones y éstas empiezan o continúan sus búsquedas laborales al volver de éstas.



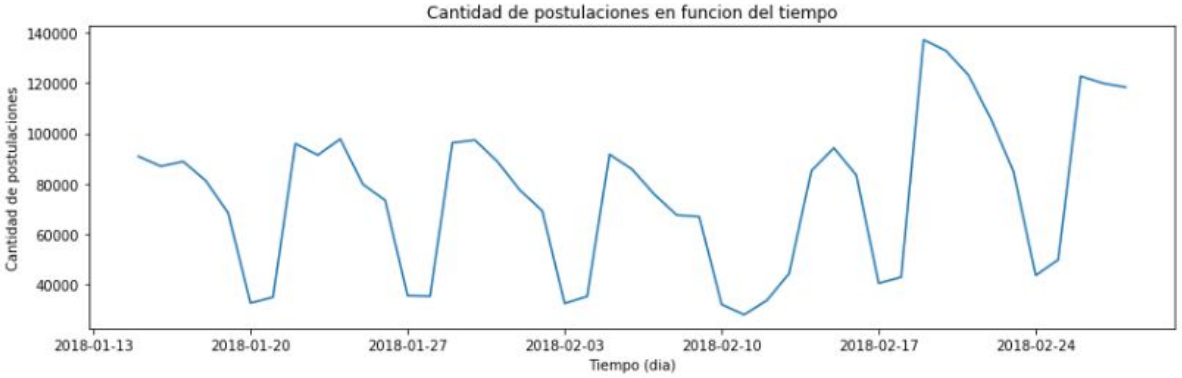
Podemos observar que dentro de las postulaciones realizadas en el mes de enero, tenemos caídas abruptas en la cantidad de postulaciones durante los fines de semana.
También se puede observar que el dataset no contiene datos de postulaciones para la segunda quincena.



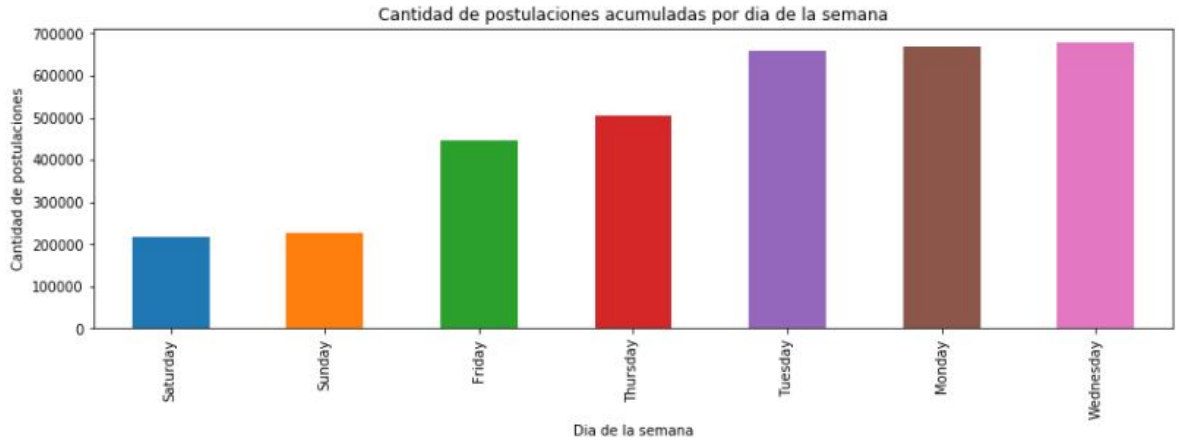
Se observa el mismo patrón durante el mes de febrero. Si bien no se tienen datos de los meses subsiguientes, se podría inferir que las personas realizan sus búsquedas laborales durante la semana, posiblemente parte durante el horario laboral del trabajo en el que se encuentran actualmente.



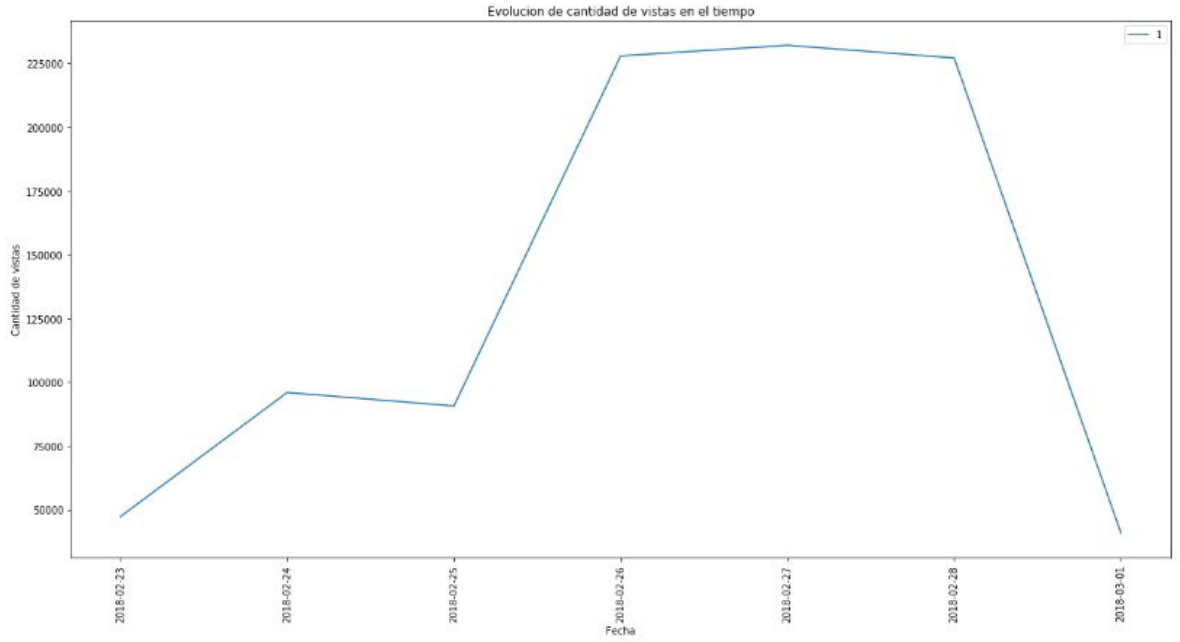
Se puede observar en este gráfico, que durante la segunda mitad de los meses se produce un aumento en la actividad de búsquedas laborales, con ciertas caídas producto de fines de semana.



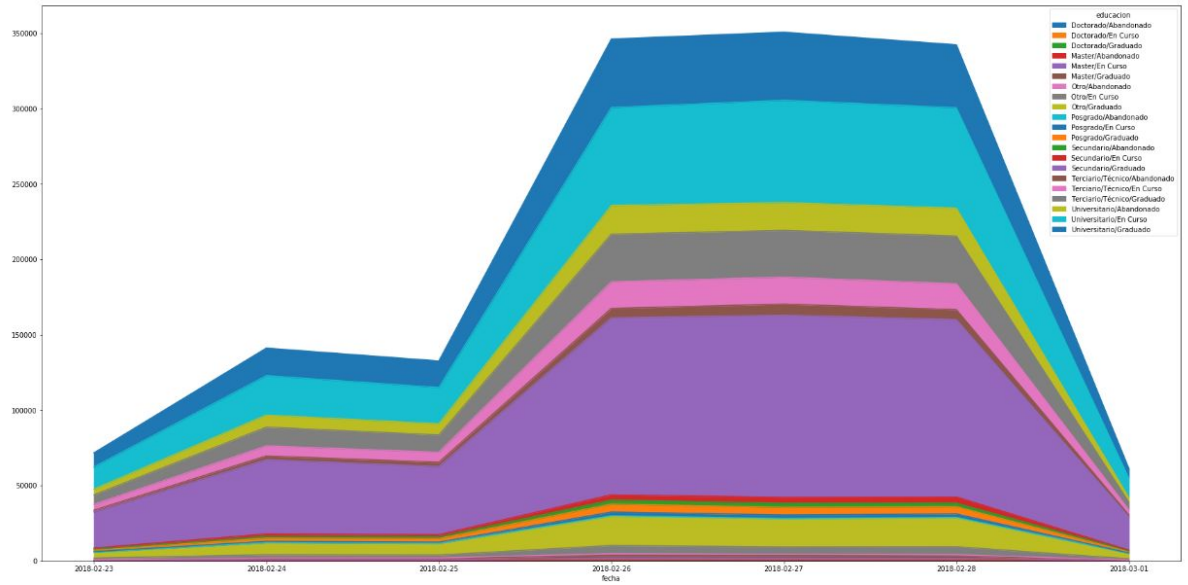
A través de este gráfico observamos que, si bien la cantidad de postulaciones baja durante ciertos periodos debido a los fines de semana, observamos una tendencia estable a lo largo de los meses.



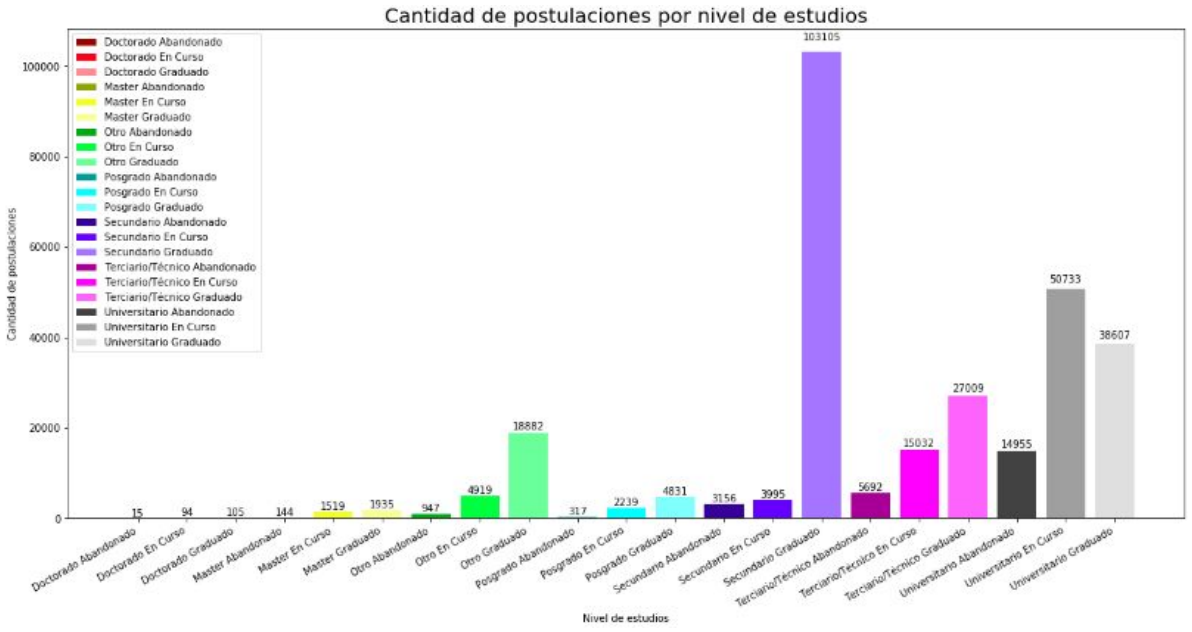
Se observa que los días de mayor actividad son los primeros días de la semana. Luego, a partir del jueves la cantidad de postulaciones decae considerablemente, y durante el fin de semana se tiene el mínimo de actividad.



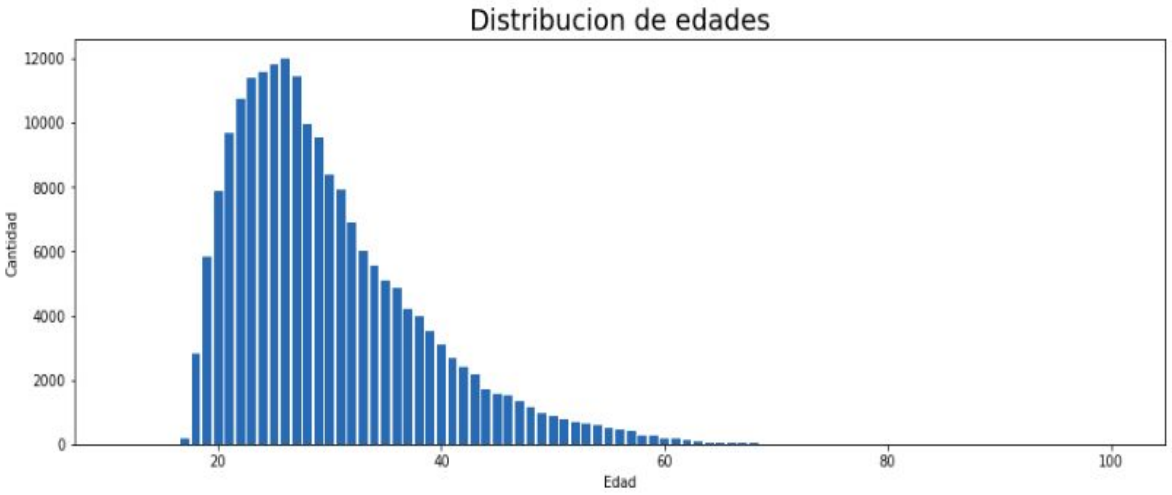
La cantidad de vistas de avisos crece a fines de febrero, lo cual nos indica que podríamos suponer que existe cierta relación entre la cantidad de vistas y cantidad de postulaciones.



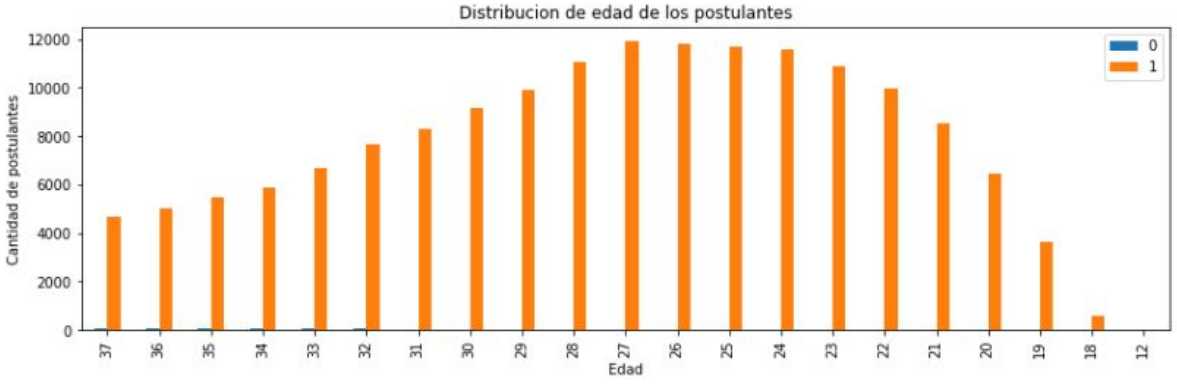
Teniendo en cuenta el nivel educativo de los postulantes que vieron los avisos, encontramos que para todos los niveles, se sigue la misma curva de vistas, siendo menos pronunciada sobre los niveles más altos. Por otro lado, vemos que los más activos son los usuarios con secundario completo.

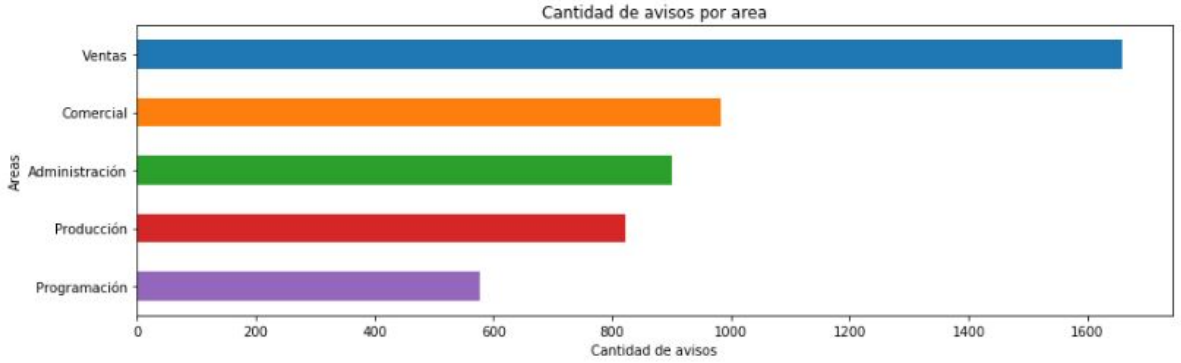


Como podíamos suponer del gráfico anterior, observamos que la mayor cantidad de postulaciones son de postulantes con secundario completo, siguiendo con universitarios (en curso y completo).



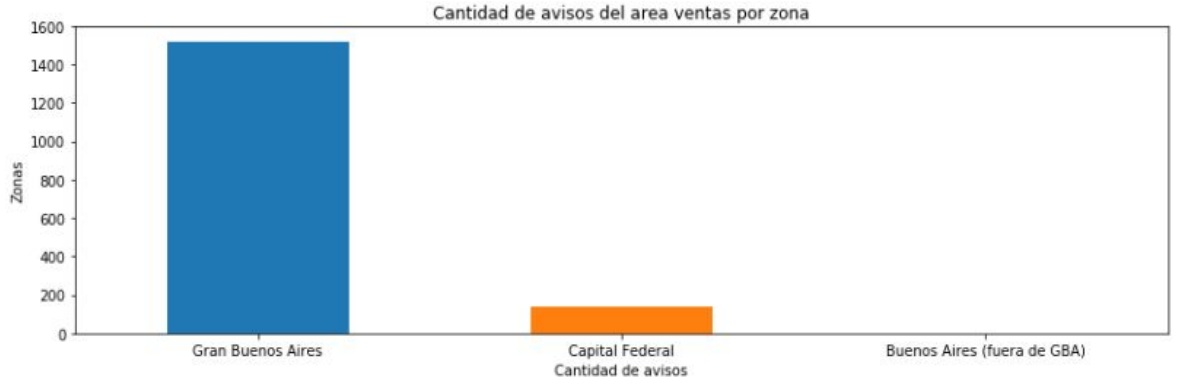
Sobre los postulantes encontramos una distribución χ^2 (chi-cuadrado). Encontrando su máximo entre los 20 y 30 años, lo cual tiene sentido teniendo en cuenta que la mayor cantidad de postulaciones son dadas por personas con secundario completo. Realizando un zoom sobre este pico:



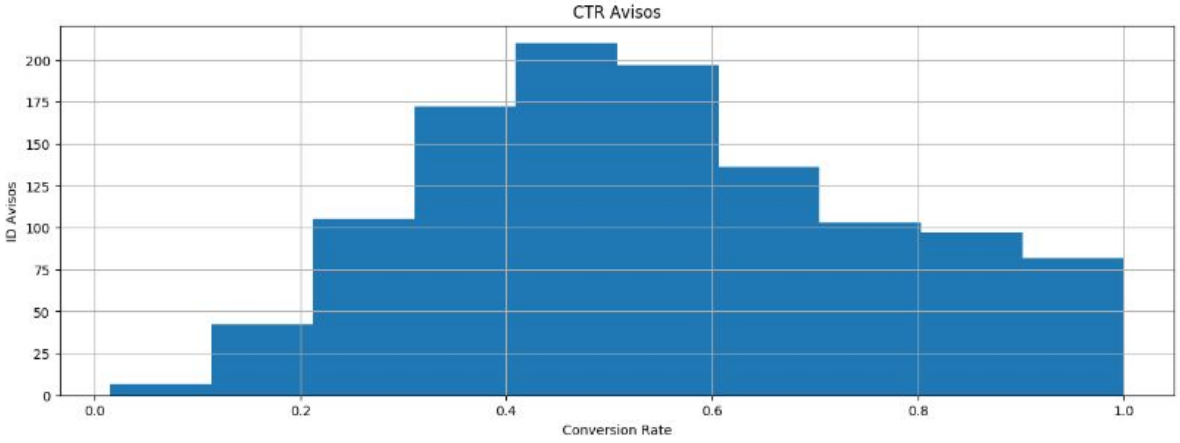


Debido a la cantidad de áreas involucradas, tomamos las 5 áreas con mayor cantidad de avisos. Con esto observamos que hay una significante mayoría de ofertas de avisos del área de ventas, seguida por el área comercial, cayendo aproximadamente 33% de la primera.

Aparentemente la mayor parte de los avisos provienen de empresas radicadas en la zona de Gran Buenos Aires, siendo las de capital muchas menos que las anteriores y las de Buenos Aires provincia y GBA Oeste casi nulas.



Se observa que para el área con mayor cantidad de avisos la distribución de cantidad de avisos por zona se mantiene.



Se observa aquí la distribución de la tasa de conversión de los avisos laborales (CTR[2] - Click through rate). Extrañamente hay gran cantidad de

[2] https://en.wikipedia.org/wiki/Click-through_rate

anuncios con CTR alto, esto hace dudar de los datos, ya que en general esta métrica da valores porcentuales bajos.
Esta métrica es el cociente entre la cantidad de postulaciones a un aviso y la cantidad de vistas que tuvo dicho aviso.

4.4. Verificar la calidad de los datos

4.4.1. Informe de la calidad de los datos

El set de datos cuenta con un nivel de calidad de datos aceptable, teniendo en cuenta que son datos únicamente de los meses de enero y febrero. Además, no se encuentra evidencia de errores en los datos. Por otro lado, proviene de Kaggle, que consideramos una fuente de información confiable.

5. Fase 3: Preparación de los datos

5.1. Selección de los datos

5.1.1. Razonamiento para la exclusión/inclusión de los datos

Para poder realizar una buena selección de los datos primero se tuvo que hacer una combinación de los datos disponibles ya que se encontraba en distintas tablas:

- **fiuba_1_postulantes_educacion**
- **fiuba_2_postulantes_genero_y_edad**
- **fiuba_3_vistas**
- **fiuba_4_postulaciones**
- **fiuba_6_avisos_detalle**

Excluimos la tabla **fiuba_5_avisos_online**, dado que consideramos no aportaba mayor información, dado que no nos interesan los avisos cuyo detalle no exista (un aviso cuyo **idaviso** no se encontrara en la tabla **fiuba_6_avisos_detalle**).

Dentro de la tabla de **fiuba_1_postulantes_educacion**, en caso que un postulante tenga más de un nivel de educación, nos quedaremos con el registro que contengan el mayor nivel alcanzado por el mismo.

Luego, para **fiuba_2_postulantes_genero_y_edad**, solo conservaremos los registros cuyos postulantes sean mayores de 18 años (los cuales pueden trabajar sin necesidad de consentimiento de padres o tutores). Para ello se utilizará el campo **fechanacimiento**, del cual podremos calcular dicha edad.

En el caso de **fiuba_3_vistas**, eliminaremos del dataset la información proveniente de los siguientes campos:

1. **idpais:** Gracias al análisis previo de los datos, encontramos que el data set solo contiene registros con **idpais = 1**.
2. **titulo** y **descripcion:** Dado que son campos con valores libres, no creemos que aporten al análisis.
3. **ciudad** y **mapacalle:** Debido a la análisis previo, encontramos que muchos registros contienen estos campos con valores nulos, por lo tanto no optamos por excluirlos.
4. **denominacion_empresa**: Dado que es un campo con valor libre, no creemos que aporte al análisis.

Por último, de **fiuba_4_postulaciones** se elimina el atributo **fechapostulacion** debido a que no encontramos interés sobre la información que aporta.

5.2. Limpieza de los datos

5.2.1. Informe de limpieza de datos

Para esto utilizando las librerías de python Pandas y PySpark cargamos los datos en DataFrames y RDDs.
El tratamiento de los campos con datos nulos, en este caso, fue la eliminación de la fila que contenía dicho valor. Otras formas de manejar estos casos pueden ser insertar el promedio o la moda de los valores de la columna a la que pertenece. A su vez, se trataron los datos según el apartado 5.1, excluyendo los campos allí señalados.

5.3. Construcción de los datos

5.3.1. Atributos derivados

Al dataset final, se le agregaron los siguientes atributos derivados:

- **edad:** Derivado de **fechanacimiento**, de la tabla **fiuba_2_postulantes_genero_y_edad**.
- **vistas_count:** Obtenido a partir de la cantidad de registros en **fiuba_3_vistas**, con el mismo número de aviso.
- **posts_count:** Obtenido a partir de la cantidad de registros en **fiuba_4_postulaciones**, con el mismo número de aviso.
- **ctr:** Obtenido a partir de la relación entre **posts_count** y **vistas_count**.

5.3.2. Registros generados

No se generaron registros auxiliares.

5.4. Integración de los datos

5.4.1. Combinación de datos

Para la unificación de los datos, se tuvo en cuenta los identificadores de avisos (**idaviso**) y de los postulantes (**idpostulante**). Dado que nuestro interés recae en la relación entre estos.
Dado que necesitaremos la información sobre las postulaciones y las no postulaciones (es decir, que un usuario vio un aviso, pero no postuló a él), la integración de datos resultará en una tabla con la información

relevante de las tablas existentes (ver apartado 5.1), y un campo extra con el cual identificamos si el registro concluyó en una postulación o no.

Para determinar si un aviso (**idaviso**) tuvo o no una postulación de algún postulante, se evaluará si en **fiuba_3_vistas** existe un registro con el aviso, que en **fiuba_4_postulaciones** no. En caso de no existir tal aviso, asumimos que no tuvo ninguna postulación. En caso de existir, tuvo por al menos una postulación.

Luego del filtrado y unificación de los datos, obtuvimos un nuevo dataset de 5510581 registros, y 10 columnas. Sobre estos registros tenemos confianza que todos sus datos son relevantes, y útiles sobre nuestro objetivo.

5.5. Formateo de los datos

5.5.1. Datos re formateados

Debido a la necesidad de identificar los datos mediante categorías, con la biblioteca de Pandas proporcionada por Python, se formatearon los siguientes campos categóricos (inicialmente representados por texto) a categorías identificables por Pandas y finalmente mediante se codificaron usando OneHotEncoding para evitar introducir relaciones de orden en campos categóricos:

- Nombre_zona
- Tipo_de_trabajo
- Nivel_laboral
- Nombre_area
- Nombre
- Estado
- Sexo

A su vez, el nuevo atributo derivado **edad**, se debió normalizar.

5.5.2. Descripción del dataset

Nombre	Tipo	Descripción	Valores permitidos
idaviso	INTEGER	Identificación del aviso	Sin restricc.
idpostulante	TEXT	Identificación del usuario	Sin restricc.
nombre_zona	CATEGORIA	Nombre de la	Gran Buenos Aires,

		zona donde se encuentra la empresa que realiza el aviso	Capital Federal, Buenos Aires (fuera de GBA), GBA Oeste
tipo_de_trabajo	CATEGORIA	Tipo de puesto que busca el aviso	Full-time, Part-time, Por Horas, Temporario, Fines de Semana, Pasantia, Teletrabajo, Por Contrato, Primer empleo
nivel_laboral	CATEGORIA	Seniority que busca el aviso	Senior / Semi-Senior, Junior, Jefe / Supervisor / Responsable, Otro, Gerencia / Alta Gerencia / Dirección
nombre_area	CATEGORIA	Área para el cual aplica el puesto del aviso	En el anexo: *Nombre de áreas laborales*
nombre	CATEGORIA	Nombre nivel educativo.	Posgrado, Universitario, Master, Otro, Terciario Técnico, Doctorado, Secundario
estado	CATEGORIA	Estado del nivel educativo	En Curso, Graduado, Abandonado
sexo	CATEGORIA	Sexo del usuario.	FEM, MASC
edad	INTEGER	Fecha de nacimiento del postulante	Rango de valores: [0, 1]
se_postula	CATEGORIA	Indicativo si el aviso tuvo alguna postulación	0: El aviso no tuvo ninguna postulación 1: El aviso tuvo alguna postulación

Extracto del dataset

idaviso	nombre_zona	tipo_de_trabajo	nivel_laboral	nombre_area	nombre	estado	sexo	edad	se_postula
17903700	Gran Buenos Aires	Full-time	Senior / Semi-Senior	Salud	Otro	Graduado	FEM	41	0
17903700	Gran Buenos Aires	Full-time	Senior / Semi-Senior	Salud	Terciario/Técnico	Graduado	FEM	41	0
1112352260	Gran Buenos Aires	Full-time	Senior / Semi-Senior	Medicina	Otro	Graduado	FEM	41	0
1112352260	Gran Buenos Aires	Full-time	Senior / Semi-Senior	Medicina	Terciario/Técnico	Graduado	FEM	41	0
1112352260	Gran Buenos Aires	Full-time	Senior / Semi-Senior	Medicina	Otro	Graduado	FEM	41	0

Extracto del dataset luego de aplicar OneHotEncoding a features categóricos

edad	se_postula	nombre_zona_Buenos Aires (fuera de GBA)	nombre_zona_Capital Federal	nombre_zona_GBA Oeste	nombre_zona_Gran Buenos Aires	tipo_de_trabajo_Fines de Semana
0.326087	1	0	0	0	1	0
0.326087	1	0	0	0	1	0
0.326087	1	0	0	0	1	0
0.152174	1	0	0	0	1	0
0.152174	1	0	0	0	1	0

6. Fase 4: Modelado

6.1. Selección de la técnica de modelado

6.1.1. Ténica de modelado

Para el modelado se utilizaran 3 modelos de clasificadores basados en Naive Bayes, árboles de decisión, y Perceptrón Multicapa, provistos por Sklearn.

- DecisionTreeClassifier: http://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html
- MultinomialNB: http://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html#sklearn.naive_bayes.MultinomialNB
- MLPClassifier: http://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

6.1.2. Presunciones de modelado

- No hay atributos nulos.
- Los atributos cuentan con una clasificación.

6.2. Generar el diseño de prueba

6.2.1. Diseño de la prueba

Para validar el entrenamiento del modelado, el set de datos fue dividido en 2. Por un lado, un set de entrenamiento del algoritmo, y por otro lado un set para validar el mismo. Para ello se utilizó la función **train_test_split** de la librería de sklearn que por defecto divide el dataset en las proporciones 80% y 20% respectivamente.

6.3. Construir el modelo

6.3.1. Parámetro de ajustes

Todos los algoritmos se ejecutaron con los parámetros por defecto que provee la biblioteca.
A continuación se menciona para cada modelo algunos de ellos:

- DecisionTreeClassifier

Parámetro	Valor	Descripción
criterion	“gini”	Criterio para medir la calidad de la división.
splitter	“best”	Estrategia para elegir la división de un nodo
max_depth	10	Máxima profundidad del árbol

- MultinomialNB

Parámetro	Valor	Descripción
alpha	1.0	Constante de Laplace smoothing
fit_prior	True	Ajustar considerando la probabilidad anterior

- MLPClassifier

Parámetro	Valor	Descripción
hidden_layer_sizes	30	Cantidad de capas intermedias
activation	“logistic”	Función de activación
solver	“adam”	Función usada para actualizar los pesos de las entradas de las neuronas. “Adam” utiliza una función basada en stochastic gradient-based
max_iter	200	Cantidad de iteraciones máximas que hará el **solver** para lograr la convergencia
shuffle	True	Mezclar los registros de entrada por cada iteración

6.3.2. Descripción del modelo

Para la siguiente combinación aviso-usuario extraída del set de entrenamiento se puede observar que el valor del campo se_postula es 1, esto quiere decir que se trata de un usuario que se postuló al aviso.

	idaviso	nombre_zona	tipo_de_trabajo	nivel_laboral	nombre_area	nombre	estado	sexo	edad	se_postula
0	1000610287	Gran Buenos Aires	Full-time	Senior / Semi-Senior	Transporte	Secundario	En Curso	MASC	33	1

Luego de entrenar los modelos, se pidió una predicción para este usuario con la función *predict* de sklearn. El resultado fue la predicción del valor 1 para el campo se_postula, por lo que se predijo la clase correcta.

Luego, se pidió con qué probabilidad se realizó esta predicción, utilizando la función *predict_proba* dando como resultado array([[0.24203928, 0.75796072]])

Esto se interpreta de la siguiente manera:

- P(se_postula = 0) = 0.24203928
- P(se_postula = 1) = 0.75796072

De esta manera se observa cómo obtener las probabilidades para la predicción de las clases para cada usuario-aviso.

6.4. Evaluar el modelo

6.4.1. Modelo evaluado

Se utilizaron las siguientes métricas para evaluar los modelos entrenados:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{precision} = \frac{tp}{tp + fp},$$

$$\text{recall} = \frac{tp}{tp + fn},$$

$$F_\beta = (1 + \beta^2)\frac{\text{precision} \times \text{recall}}{\beta^2 \text{precision} + \text{recall}}.$$

Donde:

- **tp** = true positives (verdaderos positivos)
- **fp** = false positives (falsos positivos)
- **tn** = true negatives (verdaderos negativos)
- **fn** = false negatives (falsos negativos)
- **Precision**: intenta responder a la siguiente pregunta: ¿Qué proporción de identificaciones positivas fue correcta?
- **Recall**: intenta responder a la siguiente pregunta: ¿Qué proporción de positivos reales se identificó correctamente?
- **F-measure**: métrica que combina las nociones de precision y recall.
- **F1 score**: F_β (F-measure) con $\beta = 1$

Resultados según las métricas utilizadas

- **DecisionTreeClassifier**
  - Accuracy (%): 74.41
  - F1 score (%): 85.24
  - Tiempo empleado para el entrenamiento (seg): 51.35

- **MultinomialNB**
  - Accuracy (%): 74.26
  - F1 score (%): 85.21
  - Tiempo empleado para el entrenamiento (seg): 0.68 seg

- **MLPClassifier (Perceptron Multicapa)**
  - Accuracy (%): 74.68
  - F1 score (%): 85.27
  - Tiempo empleado para el entrenamiento (seg): 243.80 seg

- **Perceptron (1 neurona)**
  - Accuracy (%): 74.07
  - F1 score (%): 85.07
  - Tiempo empleado para el entrenamiento (seg): 6.74 seg

- **Baseline (Random)**
  - Accuracy (%): 50.03
  - F1 score (%): 59.75
  - Tiempo empleado para el entrenamiento (seg): 0.18 seg

7. Fase 5: Evaluación

7.1. Evaluar los resultados

7.1.1. Evaluación de los resultados del Data Mining con respecto a los criterios de éxito

Según lo expuesto en la sección *6.4.1 Modelo Evaluado,* los modelos entrenados entregan predicciones muy por encima del baseline impuesto.
El Baseline es un modelo que predice en base a una distribución uniforme, por lo que para una clasificación binaria, cómo es el caso, predice correctamente en un 50% de los casos. Los modelos entrenados lograron predecir correctamente alrededor del 75% de las veces.

Esto significa que, luego del análisis y las técnicas utilizadas, se pudieron obtener modelos entrenados que sirven para determinar si un usuario se postulará a un aviso laboral teniendo en su información demográfica y sobre el puesto del aviso con una confianza mayor al 70%.

7.1.2. Modelos aprobados

Dado que se obtuvieron resultados confiables, se aprueban:

- El modelo de Red Bayesiana (MultinomialNB) entrenado utilizando la librería de sklearn

- El modelo de Red Neuronal (Multi Layer Perceptron) entrenado utilizando la librería de sklearn.

7.2. Proceso de revisión

7.2.1. Revisión de proceso

Las técnicas realizadas durante el proceso, así como el preprocesamiento de los datos fueron cruciales para obtener resultados que concuerdan con la realidad y siguen el objetivo. Por lo tanto, no se considera que deban sufrir ninguna modificación.

7.3. Determinar los siguientes pasos

7.3.1. Listado de posibles acciones

Las posibles acciones a realizar son:

- Realizar un análisis más exhaustivo sobre la correlación de los features del dataset entre sí y con la variable de clase, para descartar features altamente correlacionados (o sea, que tienen alta dependencia y no aportan nueva información) e informarnos de cuáles son los features que tienen mayor correlación con la clase predicha.
- Realizar alguna etapa previa de clustering para agrupar las áreas laborales en grupos más generales pero que sigan sirviendo como variable predictiva.
- Extender el análisis realizado en el presente proyecto a un dataset con datos históricos desde el inicio de los tiempos, ya que el dataset utilizado solo posee datos de los meses de enero y febrero del corriente año.
- Analizar si las clases están desbalanceadas y balancearlas añadiendo copias de la clase minoritaria o eliminando registros de la clase mayoritaria.
- Finalizar el proyecto
- Investigar nuevas técnicas para preprocesar los datos antes de entrenar el modelo y experimentar una mayor cantidad de tiempo con los parámetros de los modelos para mejorar sus predicciones.

7.3.2. Decisión

Conclusiones:

A partir del Dataset utilizado se pudo llevar a cabo el objetivo que era predecir para un usuario de los portales de Navent, si éste se postulará a un determinado aviso laboral y con qué probabilidad. Se obtuvo un conjunto de modelos que se compararon para determinar cual entregaba mejores resultados. Estos modelos toman en cuenta datos demográficos del usuario como son su edad, género y nivel educativo alcanzado y datos del aviso tales como de qué zona es la empresa que lo publicó, el área laboral del puesto ofrecido, el tipo de jornada y el nivel jerárquico del puesto. Esta información puede ser utilizada por la empresa Navent para mostrar avisos laborales a sus usuarios de una manera mas efectiva, es decir, aumentar su conversión y así poder aumentar sus ingresos.

8. Referencias

1. Link al desafío en kaggle donde se encuentra el dataset: https://www.kaggle.com/c/navent
2. Cómo codificar las variables predictoras según si son categoricas o no, y si tienen orden o no:
3. https://stats.stackexchange.com/questions/225395/where-to-find-a-guide-to-encoding-categorical-features
4. Librería de machine learning de python: http://scikit-learn.org/
5. Librería de manejo de datos en python: https://pandas.pydata.org/
6. Librería para gráficar: https://matplotlib.org/
7. Métricas de Precision y Recall: https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall?hl=es-419
8. Métrica Accuracy: https://developers.google.com/machine-learning/crash-course/classification/accuracy
9. Eligiendo la cantidad de capas ocultas para la red neuronal multicapa: https://stats.stackexchange.com/questions/181/how-to-choose-the-number-of-hidden-layers-and-nodes-in-a-feedforward-neural-netw
10.

9. Anexo - Nombres de áreas laborales

'Comercial', 'Salud', 'Transporte', 'Producci\xc3\xb3n', 'Ventas', 'Atenci\xc3\xb3n al Cliente', 'Mantenimiento', 'Ingenier\xc3\xada El\xc3\xa9ctrica y Electr\xc3\xb3nica', 'Abastecimiento', 'Administraci\xc3\xb3n', 'Contabilidad', 'Distribuci\xc3\xb3n', 'Redes', 'Soporte T\xc3\xa9cnico', 'Calidad', 'Oficios y Profesiones', 'Farmac\xc3\xa9utica', 'Liderazgo de Proyecto', 'Telecomunicaciones', 'Construcci\xc3\xb3n', 'Pr\xc3\xa1cticas Profesionales', 'Call Center', 'Recursos Humanos', 'Almac\xc3\xa9n / Dep\xc3\xb3sito / Expedici\xc3\xb3n', 'Tecnologia / Sistemas', 'Consultor\xc3\xadas Comercio Exterior', 'Selecci\xc3\xb3n', 'Comunicacion', 'Administraci\xc3\xb3n de Personal', 'Planeamiento comercial', 'Gastronomia', 'Tesorer\xc3\xada', 'Educaci\xc3\xb3n', 'Auditor\xc3\xada', 'Programaci\xc3\xb3n', 'Cr\xc3\xa9ditos y Cobranzas', 'Tecnico de Seguros', 'Asistente', 'Facturaci\xc3\xb3n', 'Medicina', 'Telemarketing', 'Ingenier\xc3\xada Mec\xc3\xa1nica', 'Planeamiento econ\xc3\xb3mico-financiero', 'Capacitaci\xc3\xb3n', 'J\xc3\xb3venes Profesionales', 'Recepcionista', 'Secretaria', 'An\xc3\xa1lisis Funcional', 'Compras', 'Administracion de Seguros', 'Dise\xc3\xb1o Textil e Indumentaria', 'Mineria/Petroleo/Gas', 'Marketing', 'Servicios', 'Compras Internacionales/Importaci\xc3\xb3n', 'Seguridad', 'Laboratorio', 'Log\xc3\xadstica', 'Responsabilidad Social', 'Ingenier\xc3\xada Industrial', 'Caja', 'Data Entry', 'Ventas Internacionales/Exportaci\xc3\xb3n', 'Direcci\xc3\xb3n de Obra', 'Compensaci\xc3\xb3n y Planilla', 'Otros', 'Ingenier\xc3\xada Electromec\xc3\xa1nica', 'Camareros', 'Back Office', 'Ingenier\xc3\xada Qu\xc3\xadmica', 'Mantenimiento y Limpieza', 'Ingenier\xc3\xada Oficina T\xc3\xa9cnica / Proyecto', 'Desarrollo de Negocios', 'Ingenier\xc3\xada Civil', 'Hoteler\xc3\xada', 'Arquitectura', 'Programaci\xc3\xb3n de producci\xc3\xb3n', 'Seguridad e Higiene', 'Otras Ingenierias', 'Impuestos', 'Ingenier\xc3\xada Automotriz', 'Ingenier\xc3\xada de Ventas', 'Operaciones', 'Legal', 'Finanzas', 'Educaci\xc3\xb3n/ Docentes', 'Control de Gesti\xc3\xb3n', 'An\xc3\xa1lisis de Riesgos', 'Gerencia / Direcci\xc3\xb3n General', 'Producto', 'Creatividad', 'Administraci\xc3\xb3n de Base de Datos', 'Dise\xc3\xb1o de Interiores / Decoraci\xc3\xb3n', 'Sistemas', 'Promotoras/es', 'Cuentas Corrientes', 'Ingenier\xc3\xada de Producto', 'Dise\xc3\xb1o Gr\xc3\xa1fico', 'Pasant\xc3\xada / Trainee', 'Corporate Finance / Banca Inversi\xc3\xb3n', 'Community Management', 'Consultoria', 'Tecnolog\xc3\xadas de la Informaci\xc3\xb3n', 'Testing / QA / QC', 'Venta de Seguros', 'Business Intelligence', 'Cadeter\xc3\xada', 'Mercadotecnia Internacional', 'Data Warehousing', 'Apoderado Aduanal', 'Bioqu\xc3\xadmica', 'E-commerce', 'Ingenier\xc3\xada de Procesos', 'Seguridad Inform\xc3\xa1tica', 'Internet', 'Investigaci\xc3\xb3n y Desarrollo', 'Medio Ambiente', 'Topograf\xc3\xada', 'Infraestructura', 'Ingenier\xc3\xada en Alimentos', 'Comercio Exterior', 'Seguros', 'Est\xc3\xa9tica y Cuidado Personal', 'Evaluaci\xc3\xb3n Econ\xc3\xb3mica', 'Finanzas Internacionales', 'Asistente de Tr\xc3\xa1fico', 'Negocios Internacionales', 'Seguridad Industrial', 'Relaciones Institucionales/Publicas', 'Dise\xc3\xb1o', 'Organizaci\xc3\xb3n y M\xc3\xa9todos', 'Planeamiento', 'Turismo', 'Ingenier\xc3\xada Agr\xc3\xb3noma', 'Dise\xc3\xb1o Industrial', 'Ingenier\xc3\xada Metalurgica', 'Ingenier\xc3\xada en Minas', 'Veterinaria', 'Trabajo Social', 'Multimedia', 'Asesor\xc3\xada Legal Internacional', 'Media Planning', 'Direcci\xc3\xb3n', 'Comunicaciones Internas', 'Dise\xc3\xb1o Web', 'Independientes', 'Qu\xc3\xadmica', 'Dise\xc3\xb1o Multimedia', 'Instrumentaci\xc3\xb3n', 'Educaci\xc3\xb3n especial', 'Inversiones / Proyectos de Inversi\xc3\xb3n', 'Telefonista', 'Urbanismo', 'Exploraci\xc3\xb3n Minera y Petroquimica', 'Otras \xc3\xa1reas t\xc3\xa9cnicas en salud', 'Farmacia hospitalaria', 'Emergentolog\xc3\xada', 'Arte y Cultura', 'Farmacia industrial', 'Farmacia comercial', 'Medicina Laboral', 'Otras Especialidades m\xc3\xa9dicas', 'Quimica', 'Auditor\xc3\xada M\xc3\xa9dica', 'Trabajo social', 'Ingenier\xc3\xada en Petr\xc3\xb3leo y Petroqu\xc3\xadmica', 'Dise\xc3\xb1o 3D', 'Ingenier\xc3\xada Geol\xc3\xb3gica', 'Bienestar Estudiantil', 'Instrumentaci\xc3\xb3n quir\xc3\xbargica', 'Idiomas', 'Traduccion', 'Comunicaciones Externas'

10. Anexo: Código relativo al modelo (Entrenamiento, predicción y métricas)

```
from sklearn.metrics import accuracy_score
from sklearn.metrics import f1_score
from sklearn.dummy import DummyClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.neural_network import MLPClassifier
from sklearn.naive_bayes import MultinomialNB
from sklearn.linear_model import Perceptron
import time

classifiers = ({'Baseline' :               DummyClassifier(strategy='uniform'),
                'Decision Tree' :          DecisionTreeClassifier(max_depth=10),
                'Naive Bayes' :            MultinomialNB(),
                'Multi Layer Perceptron' : MLPClassifier(hidden_layer_sizes=30,
activation='logistic'),
                'Perceptron' :             Perceptron(penalty='l2')
                })
results = {}

for (clf_label, clf) in classifiers.items():
    t0 = time.time()
    clf.fit(X_train, y_train)
    t1 = time.time()
    predicted = clf.predict(X_test)
    print("Params", clf.get_params())
    print("{} Classifier score on training set: {}".format(clf_label, clf.score(X_train, y_train)))
    print("{} Classifier score on validation set: {}".format(clf_label, clf.score(X_test, y_test)))
    print("{} Classifier correctly predicted: {}".format(clf_label, accuracy_score(y_test,
predicted, normalize=True)))
    print("{} F1-score for validation set: {}".format(clf_label, f1_score(y_test, predicted)))
    print("{} Classifier time needed to train: {}".format(clf_label, t1-t0))
    print()
```