

# Trabajo Práctico 1

[75.06 / 95.58] Organización de Datos  
Primer Cuatrimestre de 2018

Alumno	Padrón	E-mail
Soro, Lucas Gustavo	95665	lugusor@gmail.com
Hazan, Pablo Nehuén	96522	phazan.fiuba@gmail.com
Núñez Leyes, Javier Damián	94455	javier.nunezleyes@gmail.com
Montes, Marcelo	81397	mdmontes@gmail.com

## **Índice**

### **1\_ Introducción**

### **2\_ Pre-procesamiento del set de datos**

## 1\_ Introducción

El informe expone los resultados del análisis exploratorio de los datos del registro histórico de avisos de búsquedas laborales en <https://www.zonajobs.com.ar>

Link al set de datos:

[https://drive.google.com/file/d/1K4uRag5nmGtfuvzyJV9RL\\_73lzsh\\_iTO/view?usp=sharing](https://drive.google.com/file/d/1K4uRag5nmGtfuvzyJV9RL_73lzsh_iTO/view?usp=sharing)

Link al repositorio con los notebook en GitHub:

<https://github.com/javierleyes/OrganizacionDeDatos>

## 2\_ Pre-procesamiento del set de datos

Para cada set de datos analizamos los tipos de datos, la presencia de valores nulos o no validos, los identificadores usados y la posibilidad de relacionar los datos entre distintos sets con estos identificadores.

Se encontraron casos de fechas nulas y con formato no valido (personas menores de edad y mayores a 100 años) por ser una cantidad despreciable se descartaron los casos irregulares.

Para el caso de las columnas donde existian muy pocos datos validos para usar, se opto por descartar la columna.

### 1. Educacion de postulantes

Inspeccion rapida: Forma y calidad

# %timeit sirve para evaluar el tiempo de ejecucion

```
df_temp =  
pd.read_csv('../csv/datos_navent_fiuba/fiuba_1_postulantes_educacion.csv')  
df_temp.head()
```

	idpostulante	nombre	estado
0	NdJl	Posgrado	En Curso
1	8BkL	Universitario	En Curso
2	1d2B	Universitario	En Curso
3	NPBx	Universitario	En Curso
4	NPBx	Master	En Curso

```
df_temp['nombre'].value_counts()
```

Secundario	110256
Universitario	104295
Terciario/Técnico	47733
Otro	24748
Posgrado	7387
Master	3598
Doctorado	214

Name: nombre, dtype: int64

```
df_temp['estado'].value_counts()
```

Graduado	194474
En Curso	78531
Abandonado	25226

Name: estado, dtype: int64

```
df_temp.isnull().any()
```

```
idpostulante    False
nombre          False
estado          False
dtype: bool
```

```
(df_temp['idpostulante'].value_counts() > 1).any()
```

```
True
```

Bitacora: Todos los datos sanos y bien categorizados. Algunos postulantes tienen varios niveles de educacion.

## 2. Genero y edad de postulantes

Inspeccion rapida: Forma y calidad

```
df_temp =
pd.read_csv('../csv/datos_navent_fiuba/fiuba_2_postulantes_genero_y_edad.csv')
df_temp.head()
```

	idpostulante	fechanacimiento	sexo
0	NM5M	1970-12-03	FEM
1	5awk	1962-12-04	FEM
2	Za05	1978-08-10	FEM
3	NdJl	1969-05-09	MASC
4	eo2p	1981-02-16	MASC

```
df_temp['sexo'].value_counts()
```

```
FEM          101981
MASC          94339
NO_DECLARA    4568
Name: sexo, dtype: int64
```

```
df_temp.isnull().any()
```

```
idpostulante    False
fechanacimiento  True
sexo            False
dtype: bool
```

```
df_temp.isnull().sum()
```

```
idpostulante      0
fechanacimiento   4750
sexo              0
dtype: int64
```

```
# ok, miro cuales son las fechas malas no nulas
```

```
df_temp[
    df_temp['fechanacimiento'].notnull()[
        (pd.to_datetime(df_temp['fechanacimiento']).dropna(),
        errors='coerce').isnull())]
```

idpostulante	fechanacimiento	sexo
56206	xkPwXwY 0031-12-11	FEM
71458	LN85Y3b 0029-05-11	MASC
130846	8M2R6pz 0024-02-09	FEM
141832	A36Npjj 0033-09-14	FEM
145683	dYjV0rb 0012-11-04	NO_DECLARA
148638	GNZ0vAv 0004-07-19	MASC
149653	1QPQ8QL 0011-03-08	MASC

```
# Las fechas malas pueden descartarse
```

```
df_temp['fechanacimiento'] = pd.to_datetime(df_temp['fechanacimiento'],
errors='coerce')
```

```
# Considero fechas anteriores al siglo XX como invalidas
```

```
df_temp.loc[df_temp['fechanacimiento'] < '1900-01-01', 'fechanacimiento'] =
pd.NaT
```

```
# Considero fechas que implican edades menores a 15 años como invalidas
```

```
df_temp.loc[df_temp['fechanacimiento'] > '2003-01-01', 'fechanacimiento'] =
pd.NaT
```

```
(df_temp['idpostulante'].value_counts() > 1).any()
```

False

```
df_temp.to_csv('../csv/datos_navent_fiuba/fiuba_2_postulantes_genero_y_edad_fix.
csv')
```

Bitacora: Se encontraron fechas nulas. Tambien una pequena cantidad de fechas no validas, que fueron borradas. Una fecha de nacimiento era anterior a 1900 y tambien fue borrada. Se exporta el csv corregido.

### 3. Vista de avisos online y offline

Inspeccion rapida: Forma y calidad

```
df_temp = pd.read_csv('../csv/datos_navent_fiuba/fiuba_3_vistas.csv')
df_temp.head()
```

	idAviso	timestamp	idpostulante	
0	1111780242	2018-02-23T13:38:13.187-0500	YjVJQ6Z	
1	1112263876	2018-02-23T13:38:14.296-0500	BmVpYoR	
2	1112327963	2018-02-23T13:38:14.329-0500	wVkBzZd	
3	1112318643	2018-02-23T13:38:17.921-0500	OqmP9pv	
4	1111903673	2018-02-23T13:38:18.973-0500	DrpbXDP	

```
df_temp.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 961897 entries, 0 to 961896
Data columns (total 3 columns):
idAviso          961897 non-null int64
timestamp        961897 non-null object
idpostulante     961897 non-null object
dtypes: int64(1), object(2)
memory usage: 22.0+ MB
```

```
pd.to_datetime(df_temp['timestamp']).sort_values().head(10)
```

```
2373    2018-02-23 18:38:10.808
1041    2018-02-23 18:38:12.173
1352    2018-02-23 18:38:12.581
1691    2018-02-23 18:38:12.790
1692    2018-02-23 18:38:12.945
0        2018-02-23 18:38:13.187
2029    2018-02-23 18:38:13.269
2030    2018-02-23 18:38:13.343
351     2018-02-23 18:38:13.849
1        2018-02-23 18:38:14.296
Name: timestamp, dtype: datetime64[ns]
```

Bitacora: Todos los datos conservados. No hay errores.

#### 4. Postulaciones hasta 1 de marzo

Inspeccion rapida: Forma y calidad

```
df_temp = pd.read_csv('../csv/datos_navent_fiuba/fiuba_4_postulaciones.csv')
df_temp.head()
```

idaviso	idpostulante	fechapostulacion
0	1112257047	NM5M 2018-01-15 16:22:34
1	1111920714	NM5M 2018-02-06 09:04:50
2	1112346945	NM5M 2018-02-22 09:04:47
3	1112345547	NM5M 2018-02-22 09:04:59
4	1112237522	5awk 2018-01-25 18:55:03

```
df_temp.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3401623 entries, 0 to 3401622
Data columns (total 3 columns):
idaviso          int64
idpostulante     object
fechapostulacion object
dtypes: int64(1), object(2)
memory usage: 77.9+ MB
```

```
pd.to_datetime(df_temp['fechapostulacion']).sort_values().head(10)
```

```
1525012    2018-01-15 00:00:01
1269880    2018-01-15 00:00:02
1842775    2018-01-15 00:00:09
1525013    2018-01-15 00:00:10
3348905    2018-01-15 00:00:11
222799     2018-01-15 00:00:16
1812230    2018-01-15 00:00:16
1558135    2018-01-15 00:00:16
2435961    2018-01-15 00:00:16
3159078    2018-01-15 00:00:18
```

Name: fechapostulacion, dtype: datetime64[ns]

```
df_temp['idpostulante'].apply(len).value_counts()
```

```
7    2763243
6    632698
5     5278
4     404
```

Name: idpostulante, dtype: int64

Bitacora: Todos los datos conservados. No hay errores.

## 5. Avisos online al 8 de marzo

Inspeccion rapida: Forma y calidad

```
df_temp = pd.read_csv('../csv/datos_navent_fiuba/fiuba_5_avisos_online.csv')
df_temp.head()
```

```
      idaviso
0    1112355872
1    1112335374
2    1112374842
3    1111984070
4    1111822480
```

```
df_temp.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5028 entries, 0 to 5027
Data columns (total 1 columns):
idaviso      5028 non-null int64
dtypes: int64(1)
memory usage: 39.4 KB
```

```
df_temp['idaviso'].isnull().any()
```

False

Bitacora: Todos los datos conservados. No hay errores.

## 6. Detalle de avisos online y offline

Inspeccion rapida: Forma y calidad

```
df_temp = pd.read_csv('../csv/datos_navent_fiuba/fiuba_6_avisos_detalle.csv')
```

```
df_temp.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 13534 entries, 0 to 13533
Data columns (total 11 columns):
idaviso                13534 non-null int64
idpais                13534 non-null int64
titulo                13534 non-null object
descripcion            13534 non-null object
nombre_zona           13534 non-null object
ciudad                47 non-null object
mapacalle             872 non-null object
tipo_de_trabajo       13534 non-null object
nivel_laboral         13534 non-null object
nombre_area           13534 non-null object
denominacion_empresa  13529 non-null object
dtypes: int64(2), object(9)
memory usage: 1.1+ MB
```

```
df_temp.isnull().sum()
```

```
idaviso                0
idpais                0
titulo                0
descripcion            0
nombre_zona           0
ciudad               13487
mapacalle            12662
tipo_de_trabajo       0
nivel_laboral         0
nombre_area           0
denominacion_empresa  5
dtype: int64
```

```
df_temp['idpais'].value_counts()
```

```
1    13534
Name: idpais, dtype: int64
```

```
df_temp['nombre_zona'].value_counts()
```

```
Gran Buenos Aires    12654
Capital Federal      876
Buenos Aires (fuera de GBA)  2
GBA Oeste             2
Name: nombre_zona, dtype: int64
```



```
df_temp['ciudad'].value_counts()
```

Buenos Aires	14
Argentina	13
CABA	3
San Isidro	2
Capital Federal	2
paternal	1
Santa Rosa	1
Microcentro	1
República Argentina	1
Tortuguitas	1
Buenos Aires Province	1
Parque Patricios	1
La Plata	1
Barracas	1
Mendoza	1
caba	1
Vicente Lopez	1
Zarate, Campana, Escobar	1

Name: ciudad, dtype: int64

```
df_temp['tipo_de_trabajo'].value_counts()
```

Full-time	12339
Part-time	863
Teletrabajo	110
Pasantia	63
Por Horas	63
Temporario	42
Por Contrato	37
Fines de Semana	14
Primer empleo	3

Name: tipo\_de\_trabajo, dtype: int64

```
df_temp['nivel_laboral'].value_counts()
```

Senior / Semi-Senior	9407
Junior	2216
Otro	921
Jefe / Supervisor / Responsable	809
Gerencia / Alta Gerencia / Dirección	181

Name: nivel\_laboral, dtype: int64

```
df_temp['nombre_area'].value_counts()
```

Ventas	1659
Comercial	983
Administración	901
Producción	821
Programación	576
Contabilidad	416
Tecnología / Sistemas	388
Atención al Cliente	347
Mantenimiento	324
Recursos Humanos	235
Gastronomía	234
Oficios y Profesiones	209
Soporte Técnico	203
Logística	200
Call Center	191
Almacén / Depósito / Expedición	184
Compras	170
Marketing	153

Otros	153
Administración de Personal	152
Recepcionista	151
Transporte	148
Mantenimiento y Limpieza	141
Telemarketing	138
Finanzas	138
Tesorería	137
Créditos y Cobranzas	132
Salud	127
Desarrollo de Negocios	126
Medicina	119

Auditoría Médica	3
Instrumentación	2
Topografía	2
Data Warehousing	2
Educación especial	2
Trabajo Social	2
Trabajo social	2
Diseño Multimedia	2
Mercadotecnia Internacional	2
Otras áreas técnicas en salud	2
Ingeniería Geológica	2
Diseño 3D	2
Medicina Laboral	2
Dirección	2
Responsabilidad Social	2
Farmacia comercial	2
Bienestar Estudiantil	1
Urbanismo	1
Comunicaciones Externas	1
Farmacia hospitalaria	1
Traducción	1
Idiomas	1
Exploración Minera y Petroquímica	1
Otras Especialidades médicas	1
Emergentología	1
Arte y Cultura	1
Telefonista	1
Instrumentación quirúrgica	1
Química	1
Ingeniería en Petróleo y Petroquímica	1

Name: nombre\_area, Length: 173, dtype: int64

```
df_temp['nombre_area'].str.upper().value_counts().count()
```

172

Bitacora:

Hay problemas de categorizacion en ciudad. Ej: "Republica Argentina"  
En nombre\_area hay nombres repetidos con diferencia de mayusculas.