Solución Propuesta

Creemos que, como muestran Banko y Brill[1], el factor decisivo a la hora de predecir los tipos de crímenes no va a ser el algoritmo que elijamos, sino la información que encontremos y podamos usar para entrenar nuestro algoritmo.

El último mes el grupo se abocó a la tarea de exploración de datos y al agregado de información de distintas fuentes. Se tuvieron en cuenta factores como la luz del día[2], el clima[3], y planeamos seguir incorporando fuentes de datos como feriados, media de ingresos familiares por distrito, eventos masivos como partidos de *baseball*. Incorporaremos nuevas fuentes de datos a medida que probemos que nuestro algoritmo puede procesarlas en un tiempo razonable.

Luego de leer y discutir las ventajas y desventajas de algunos algoritmos, decidimos avanzar con Perceptron Multiclase.
Habiendo seleccionado este algoritmo, todavía quedaba definir si se utilizará la variante One vs All o One vs One.
Según lo leído, One vs One podría ser más eficiente a la hora de predecir las probabilidades para las categorías de los crímenes, pero necesita de muchísimos más recursos. Sin tener en cuenta el entrenamiento, para definir las probabilidades de un solo registro se tendría que aplicar 917 Perceptrones binarios (uno por cada par de las 39 categorías posibles) y luego analizar la cantidad de triunfos. Con los volúmenes de datos que estamos analizando, esta solución parece inviable. Por este motivo, nos centraremos en One vs All, en donde el grueso del tiempo se da en el entrenamiento, mientras que la definición de las probabilidades es instantánea.

Entrenaremos los 39 Perceptrones binarios (uno por categoría de la forma pertenece/no pertenece) con nuestro nuevo set de entrenamiento. Esta es la tarea que consumirá más tiempo. Luego, para clasificar a los distintos registros, multiplicaremos cada vector de peso obtenido en la etapa anterior. Las probabilidades serán asignadas en base al resultado de esta operación y las diferencias con el resto de los resultados.

Sin embargo, antes de poder aplicar el algoritmo, debemos analizar dos cuestiones muy importantes: cómo vamos a definir el umbral de pertenencia y, en caso de no llegar a una diferenciación perfecta mediante el entrenamiento, cuántas iteraciones serán suficientes para definir los pesos?

[1] http://research.microsoft.com/pubs/66840/acl2001.pdf
[2] ver Incorporación del atributo Daylight
[3] ver Incorporación de información sobre clima (Weather)

Exploración de datos

Análisis de la estructura de datos en train.csv

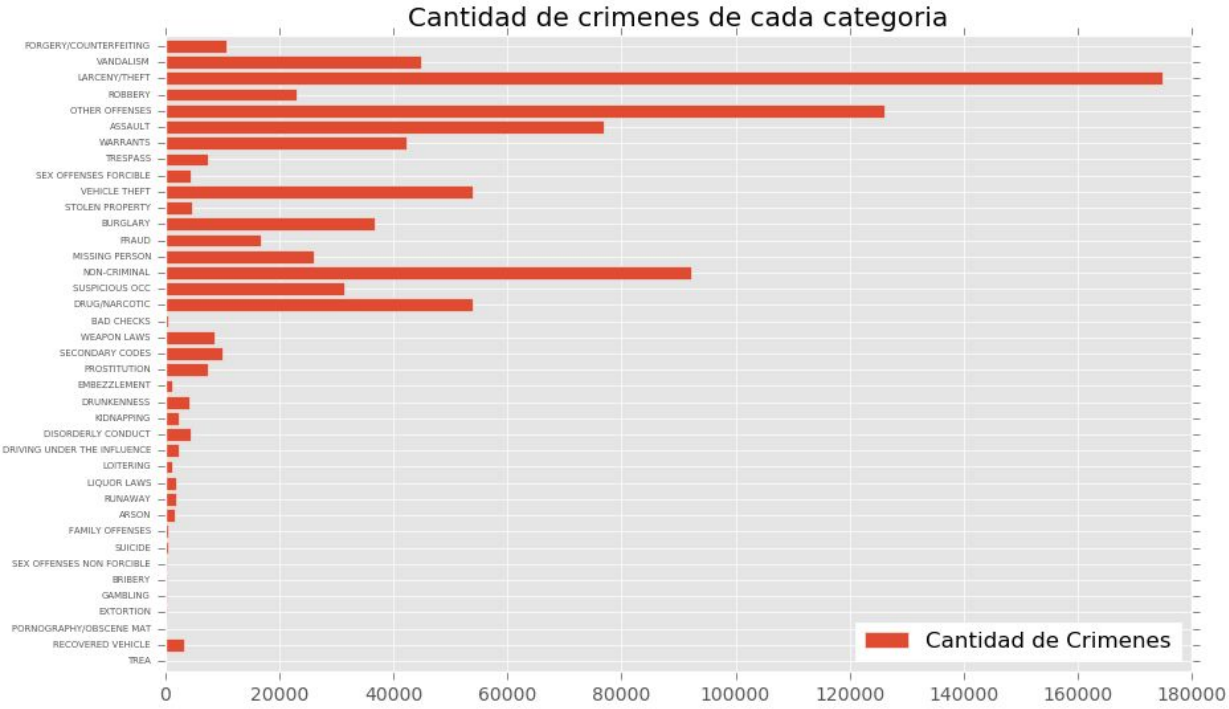
- Cantidad de observaciones: 878049
- Cada observación tiene 8 atributos más la clase Category

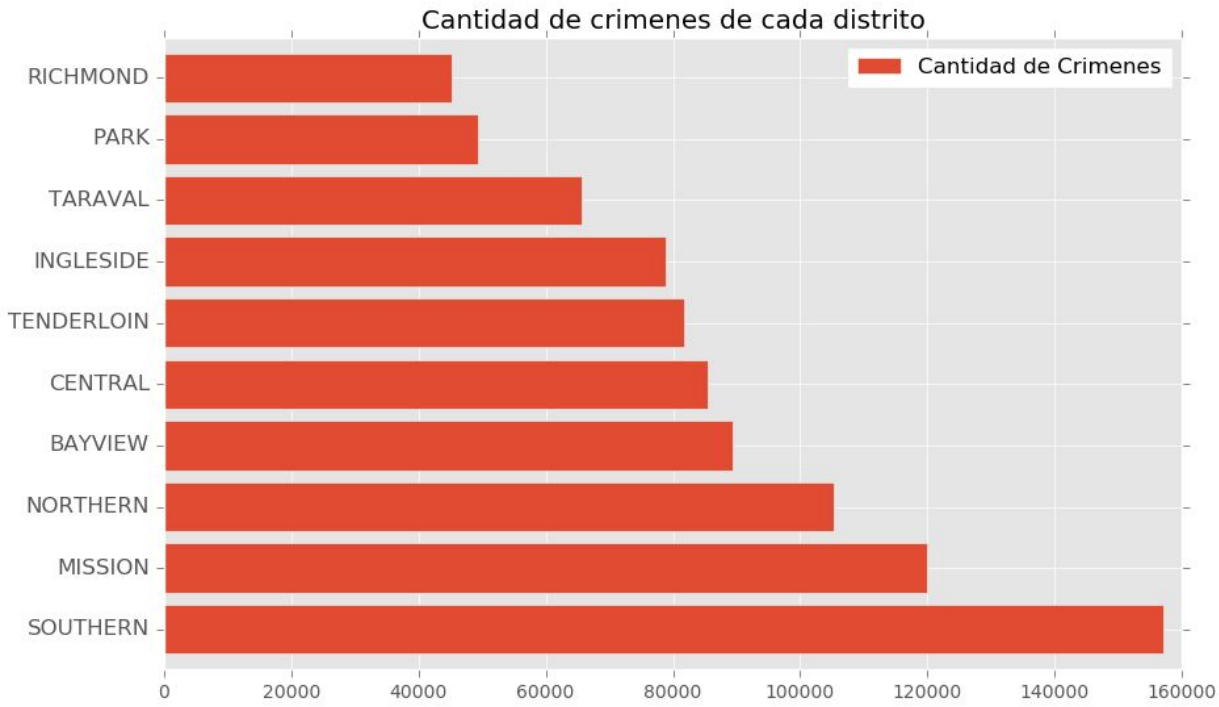
Atributos Train.csv	**Tipo de dato**	**#factores**	**min**	**max**	**avg**	**std**
*Dates [%Y-%m-%d %H:%M:%S]	factor	4511				
Category (clase)	factor	39				
Descript	factor	879				
DayOfWeek	factor	7				
PdDistrict	factor	10				
Resolution	factor	17				
Address	factor	23218				
X [ºLongitud]	float number		-122.514	-120.5	-122.423	0.03
Y [ºLatitud]	float number		37.708	90	37.771	0.457

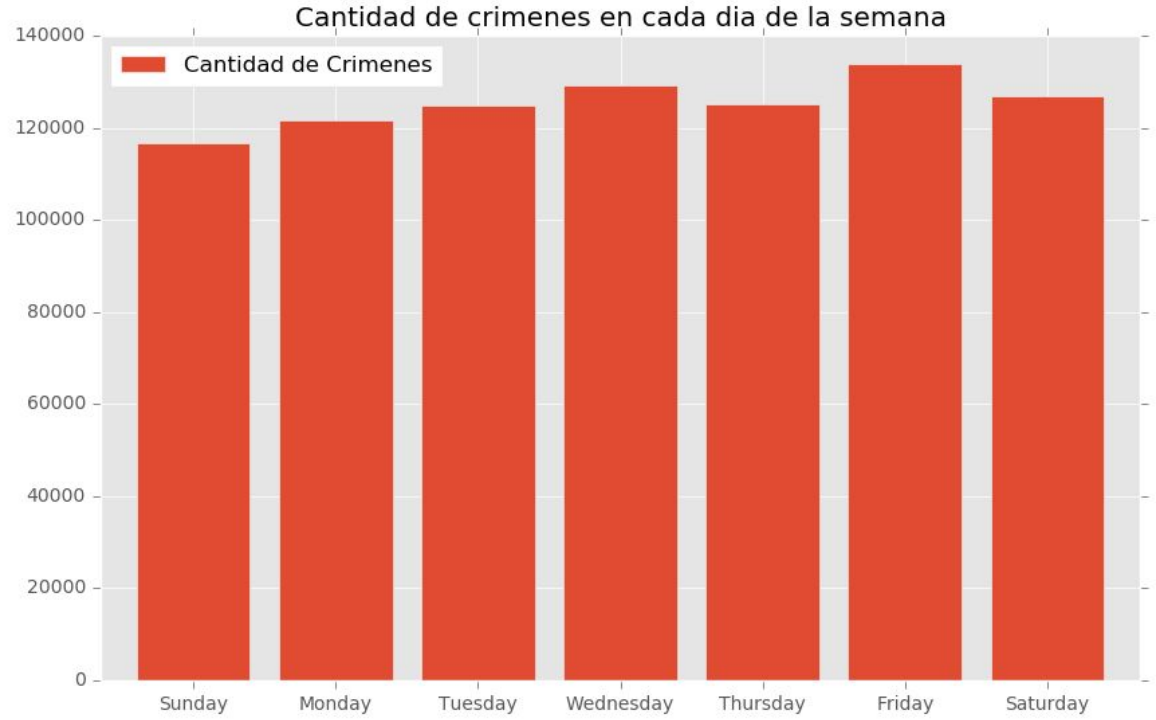
*"Min" 2003-01-06 00:01:00
*"Max" 2015-05-13 23:59:59

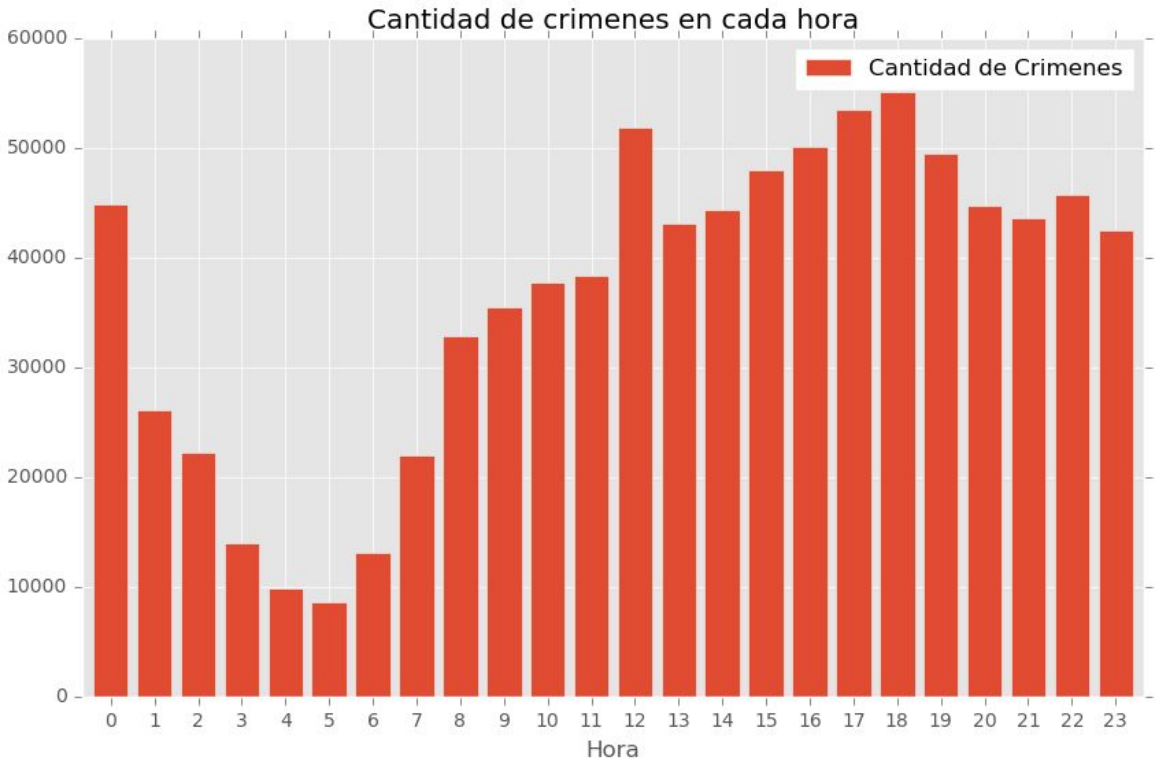
Los atributos X Y presentan 69 valores anómalos. Los cuales representan el 0,0076306% del set.

Algunos plots del train.csv






Cantidad de crimenes en cada dia de la semana
Cantidad de Crimenes
140000
120000
100000
80000
60000
40000
20000
0
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday


Cantidad de crimenes en cada hora
Cantidad de Crimenes
60000
50000
40000
30000
20000
10000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hora

Postprocesado de datos

Primeras instancias

Al comenzar este trabajo, quisimos analizar cómo se comportaba KNN solamente observando la distancia física, siempre teniendo presente que, seguramente, la efectividad iba a ser baja.
Al implementar este algoritmo, no sólo comprobamos nuestra hipótesis inicial, sino que también pudimos darnos cuenta de lo importante que iba a ser la efectividad del algoritmo, ya que luego de alrededor de 2 horas, el *script* no había logrado analizar más del 5% del archivo.
En segunda instancia, limitamos la comparación a crímenes dentro del mismo distrito. Nuevamente, los resultados fueron desalentadores. Finalmente, decidimos ordenar todos las entradas según la distancia al punto (0,0) y, mediante búsqueda binaria, agilizar los procesos de comparación. Recién con esta modificación logramos procesar todos los registros en un lapso de tiempo aceptable (aproximadamente 1 minuto), logrando una efectividad en la predicción de ~30% (se utilizó el mismo set de entrenamiento como set de testeo, descartando el propio crimen a analizar a la hora de realizar comparaciones)

Hasta el momento

El proceso actual solamente realiza procesos de mapeo. Es decir, la cantidad de filas de nuestro nuevo archivo postprocesado es igual a la cantidad original. Se deja abierta la posibilidad de realizar un proceso de filtrado como podría ser *backward selection*, para reducir el set de datos manteniendo sus características.
De ser necesario se aplicará un PCA (Análisis de Componentes Principales) para reducir dimensiones.

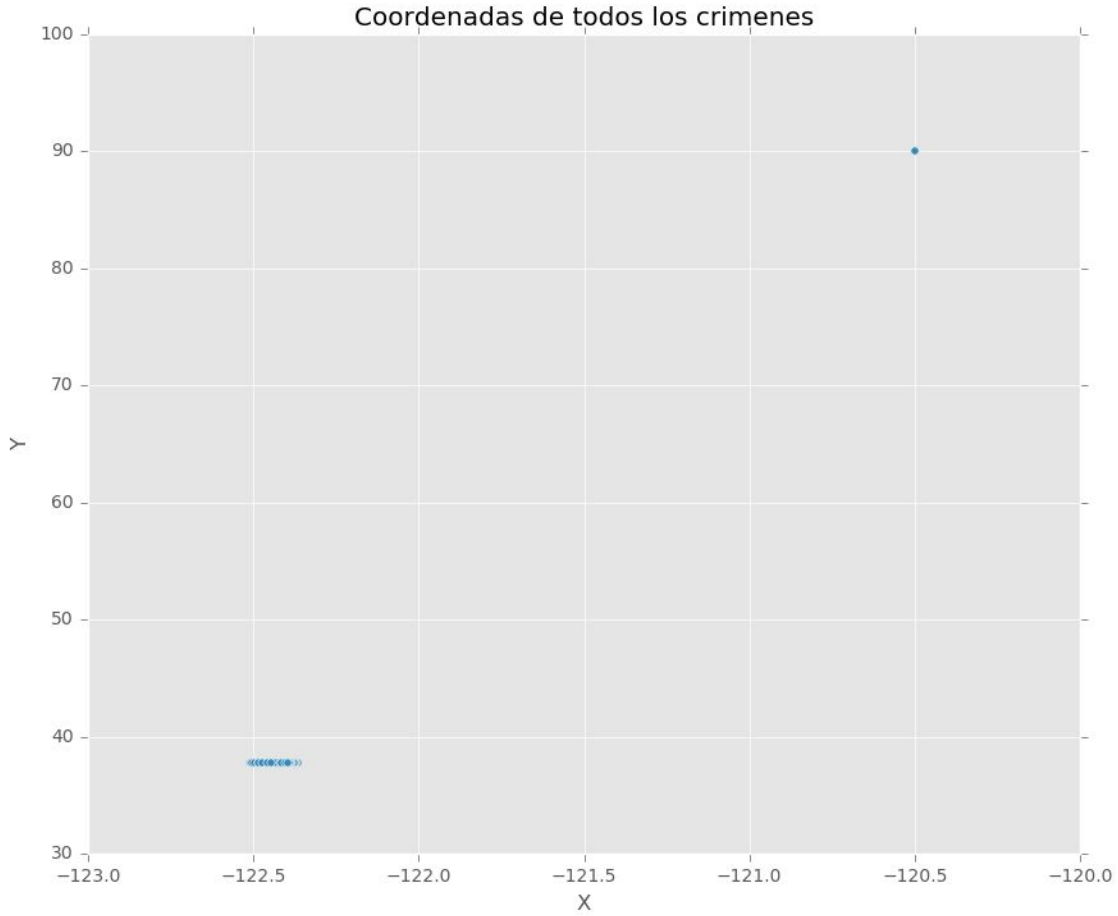
El proceso de mapping sobre cada *row* consiste en:

1. Descarte de los atributos Descript y Resolution ya que no aparecen en el set *test*
2. Descarte de los atributos PdDistrict y Address, ya que postulamos que están contenidos dentro de la información de latitud y longitud (queda a corroborar con PCA y backward selection)
3. Parseo de la fecha y creación de atributos específicos por componente:
    a. Month
    b. Day
    c. Time
4. Conversión del atributo Date a formato *Unix timestamp*, cuya naturaleza numérica lo hace más apropiado para su análisis
5. Creación de columnas correspondientes al día de la semana, con un 1 en aquella que corresponda al día de la *row*
6. Corrección de coordenadas anómalas
7. Normalización de coordenadas
8. Adición del atributo *Holiday* indicando si el día del crimen fue feriado

9. Adición del atributo *Daylight* en base a la hora del crimen
10. Adición de atributos binarios indicando la presencia de eventos climáticos
11. Adición de los atributos relacionados a la temperatura y precipitación

Coordenadas X,Y anómalas

Se detectaron 69 instancias anómalas. Las cuales representan el 0,0076306% del set. Todas comparten la misma coordenada. (X, Y) = (-120.5, 90). Es decir, el polo norte!



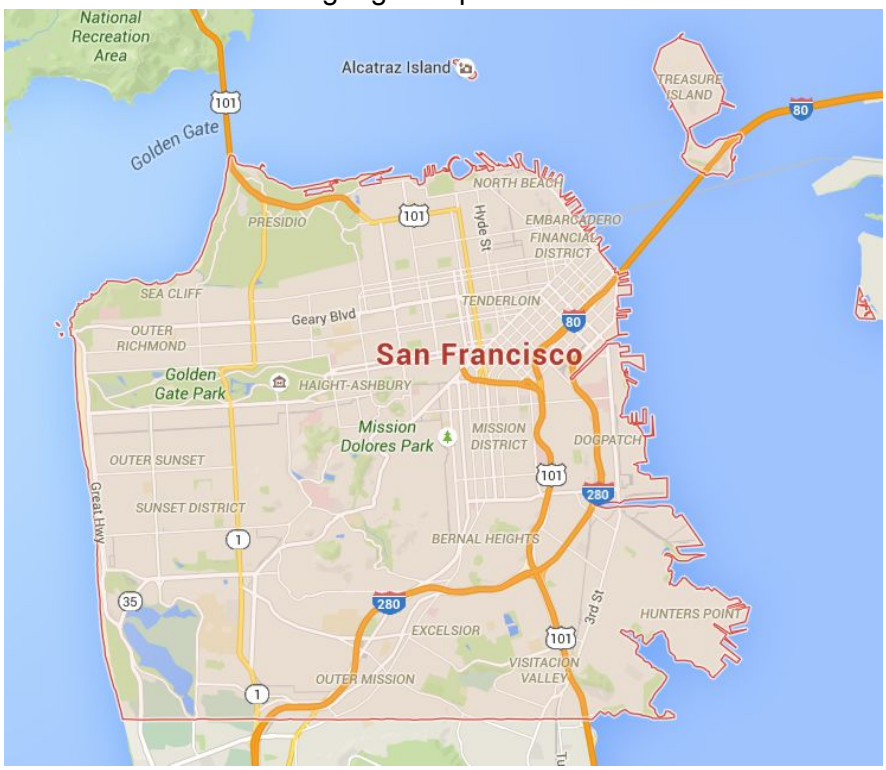
Los puntos azules arriba a la derecha son las coordenadas anómalas.

Opciones para solucionar el problema:
1. Eliminar observaciones con atributos anómalos
2. Imputación de datos
3. Utilización de algoritmos de regresión o clasificación

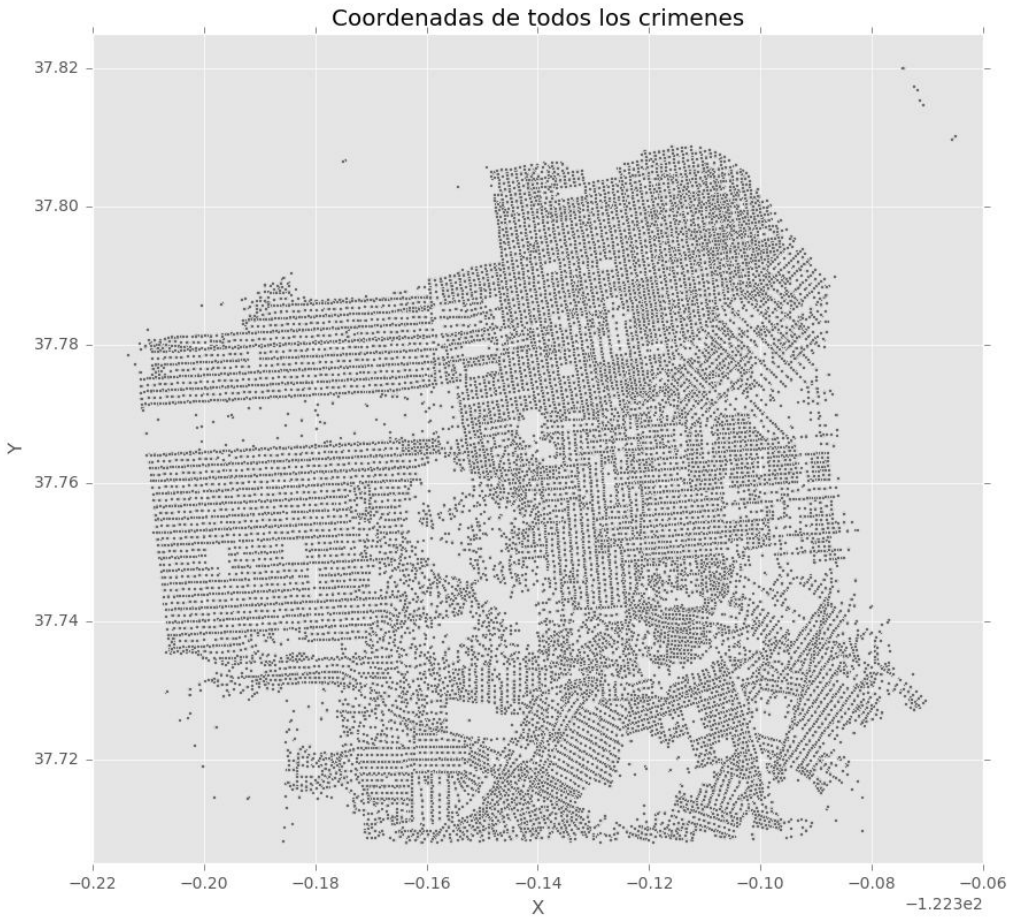
Se optó por imputar los datos anómalos de los atributos X e Y a partir de obtener el promedio (de X e Y) de los distritos a los que pertenece cada observación (sin tener en cuenta el desvío generado por las observaciones anómalas).

Atributos Act.	**Tipo de dato**	**min**	**max**	**avg**	**std**
X [ºLongitud]	float number	-122.514	-122.365	-122.423	0.025
Y [ºLatitud]	float number	37.708	37.82	37.767	0.024

Mapa de San Francisco obtenido de google maps:



Resultado de la corrección de las coordenadas anómalas:



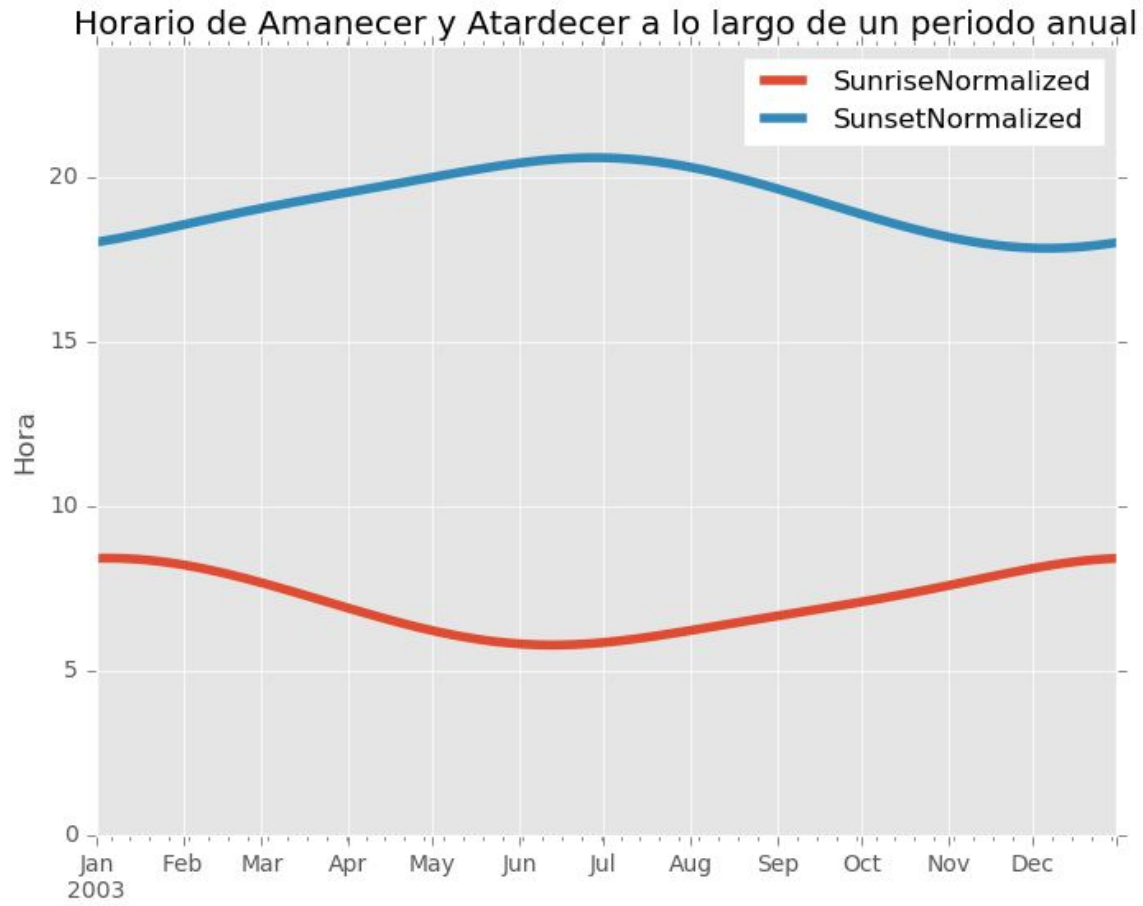
Incorporación del atributo Daylight

Se decidió incorporar un atributo (Daylight) que nos permita saber si un crimen se cometió de día o de noche.

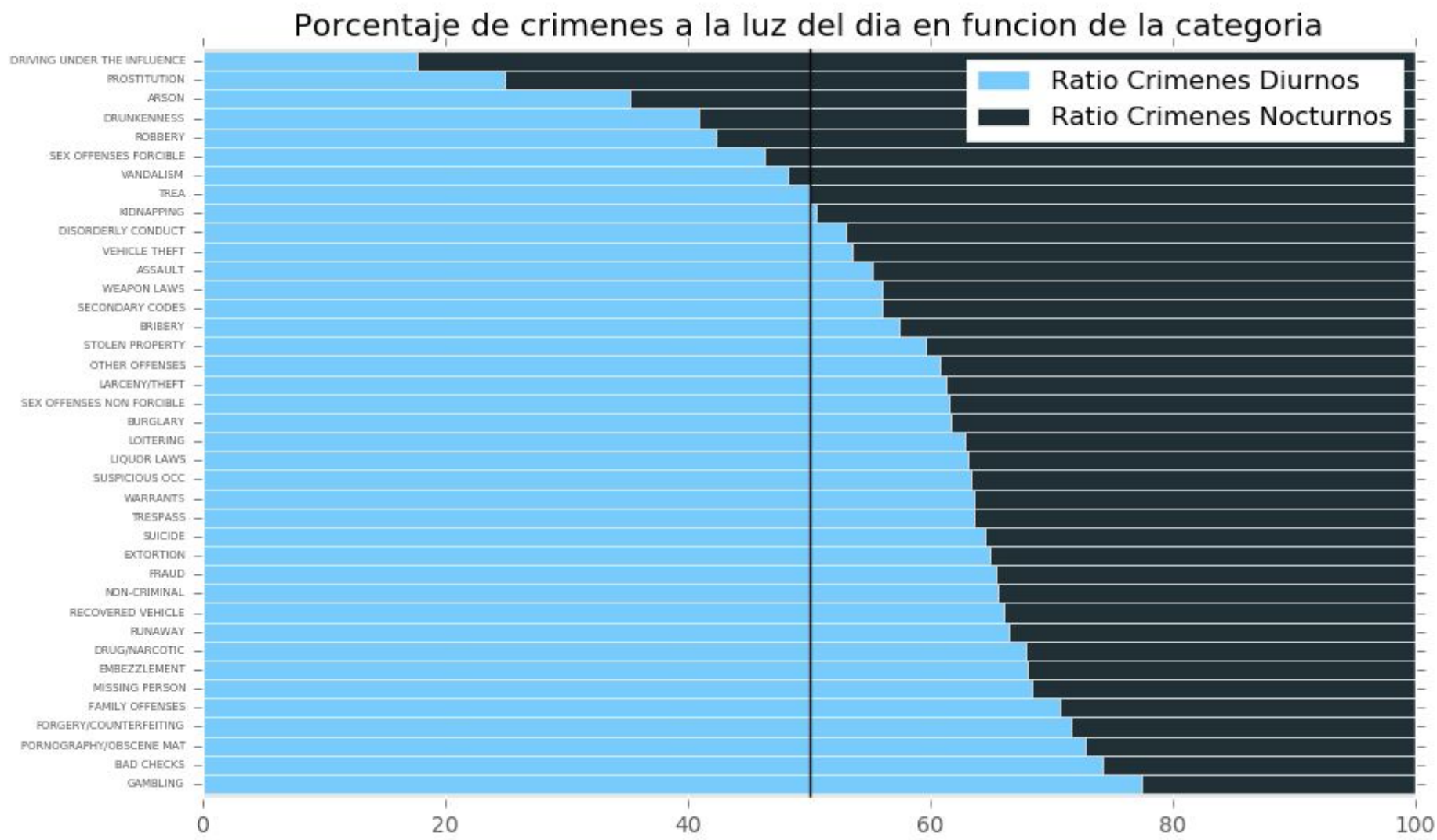
Se creo un script que generara el archivo ssrc.csv a partir de hacer peticiones a la página http://sunrise-sunset.org/.

Este csv consta del siguiente formato:

Atributo	**Ejemplo de Formato**
Date [%Y-%m-%d]	2003-01-01
Sunrise [%H:%M:%S]	08:25:08
Sunset [%H:%M:%S]	18:01:51



A partir de esta información se pudo determinar si un crimen se cometió durante el día o durante la noche y se lo incorporó dicha información al train-filtered.csv como un atributo nuevo llamado Daylight.


Porcentaje de crimenes a la luz del dia en funcion de la categoria
Ratio Crimenes Diurnos
Ratio Crimenes Nocturnos
DRIVING UNDER THE INFLUENCE
PROSTITUTION
ARSON
DRUNKENNESS
ROBBERY
SEX OFFENSES FORCIBLE
VANDALISM
TREA
KIDNAPPING
DISORDERLY CONDUCT
VEHICLE THEFT
ASSAULT
WEAPON LAWS
SECONDARY CODES
BRIBERY
STOLEN PROPERTY
OTHER OFFENSES
LARCENY/THEFT
SEX OFFENSES NON FORCIBLE
BURGLARY
LOITERING
LIQUOR LAWS
SUSPICIOUS OCC
WARRANTS
TRESPASS
SUICIDE
EXTORTION
FRAUD
NON-CRIMINAL
RECOVERED VEHICLE
RUNAWAY
DRUG/NARCOTIC
EMBEZZLEMENT
MISSING PERSON
FAMILY OFFENSES
FORGERY/COUNTERFEITING
PORNOGRAPHY/OBSCENE MAT
BAD CHECKS
GAMBLING
0
20
40
60
80
100

Incorporación de información sobre clima (Weather)

Se decidió incorporar información sobre el clima. Creemos que puede repercutir enormemente en diferentes categorías de crímenes. Ya sea desde accidentes automovilísticos, desvanecimientos por olas de calor, suicidios, etc.

Las fuentes de información son las siguientes:
http://www.crh.noaa.gov/
http://www.ncdc.noaa.gov/

A partir de ellas se generó un csv con los siguientes atributos:

Atributo [unidad]	**Ejemplo de Formato**
Date [%Y-%m-%d]	2003-01-01
TemperatureMin [ºF]	46
TemperatureMax [ºF]	59
TemperatureAvg [ºF]	52.5
Precipitation [inch]	0
Drizzle [binary]	0
Fog [binary]	1
Ground Fog [binary]	0
Hail [binary]	0
Heavy Fog [binary]	0
Demaging Winds [binary]	0
Mist [binary]	1
Rain [binary]	0
Thunder [binary]	0

Transformación del train.csv al train-filtered.csv

- Cantidad de observaciones: 878049 (no se filtraron observaciones)

Observaciones en crudo

Atributos Train.csv [unidad]	**Tipo de dato**
Dates [%Y-%m-%d %H:%M:%S]	factor
Category	factor
Descript	factor
DayOfWeek	factor
PdDistrict	factor
Resolution	factor
Address	factor
X [ºLongitud]	float number
Y [ºLatitud]	float number

Atributos Daylight [unidad]	**Tipo de dato**
Daylight	factor (2: Day, Night)

Atributos [unidad]	**Tipo de dato**
Holyday	factor (nHolydayTypes: Navidad, Año nuevo, ...)

Atributos Weather [unidad]	**Tipo de dato**
TemperatureMin/Max/Avg [ºF]	3 x float number
Precipitation [inch]	binary
Drizzle [binary]	binary
Fog [binary]	binary
Ground Fog [binary]	binary
Hail [binary]	binary
Heavy Fog [binary]	binary
Demaging Winds [binary]	binary
Mist [binary]	binary
Rain [binary]	binary
Thunder [binary]	binary

Observaciones normalizadas

bin = binario {-1, 1}
# = atributo numérico punto flotante de rango [-1, 1] con media 0 y desvío estándar 1.
-- = atributo eliminado

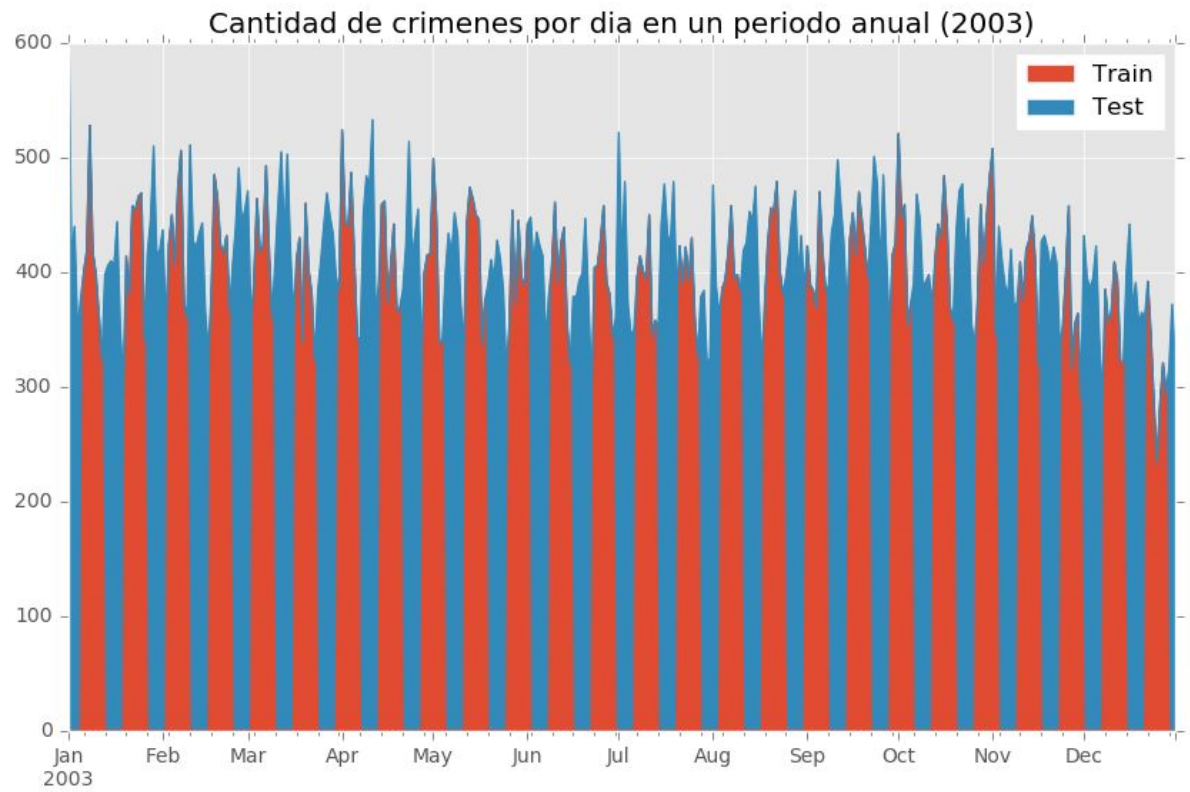
Atributos Normalizados	**Tipo de dato**
Date	1#
Month	12 bin
Day	31 bin
Time	~~24 bin~~ vs 24#
Category	-- **(es la clase)**
Descript	--
PdDistrict	~~7bin~~ vs --
Resolution	--
Adress	~~23218 bin~~ vs --
X	1#
Y	1#
Daylight	1 bin
HolyDay	1 bin vs ~~nHolyDayTypes bin~~
TemperatureMin/Max/Avg	#3
Precipitation	#1
ClimateTypes*	9 bin
TOTAL	**31# + 54 bin = 85 atributos**

*Drizzle, Fog, Ground Fog, Hail, Heavy Fog, Demaging Winds, Mist, Rain, Thunder

Curiosidades y/o patrones detectados

En navidad los crímenes descienden

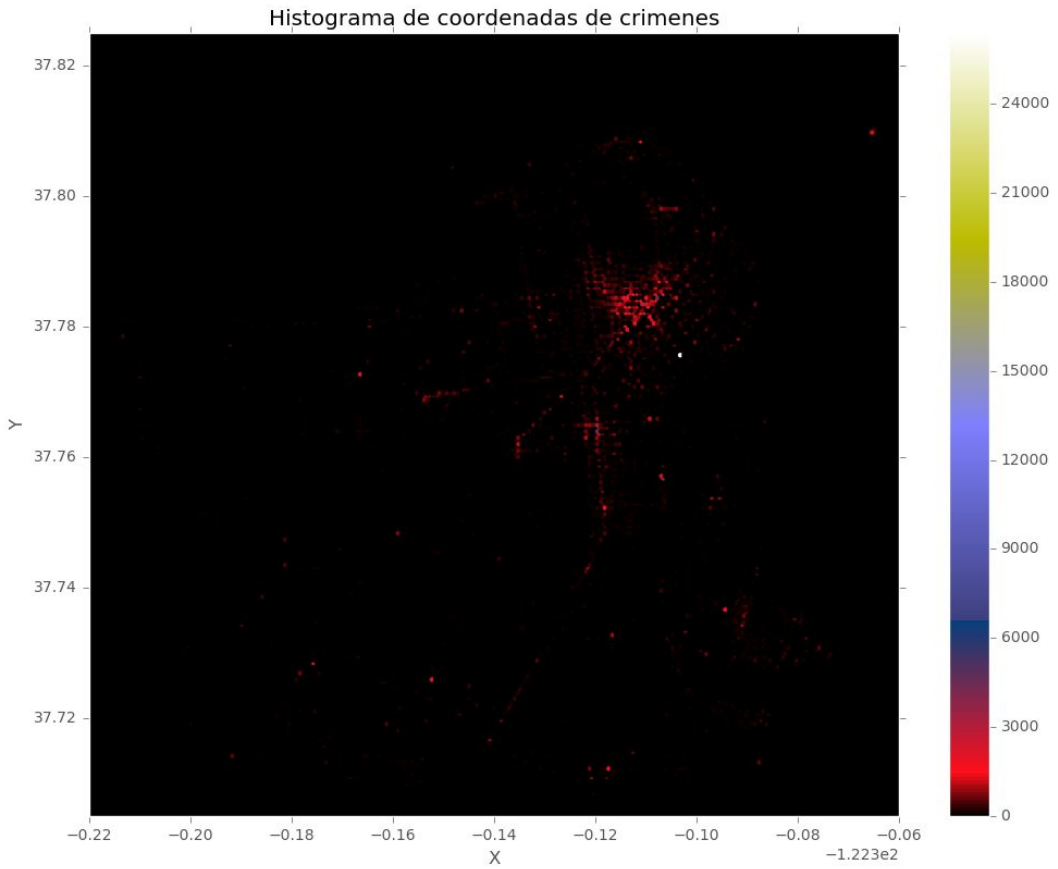
El train.csv contiene una serie de semanas contempladas entre el 2003 y 2015 mientras que el test.csv contiene el complemento de esas semanas.



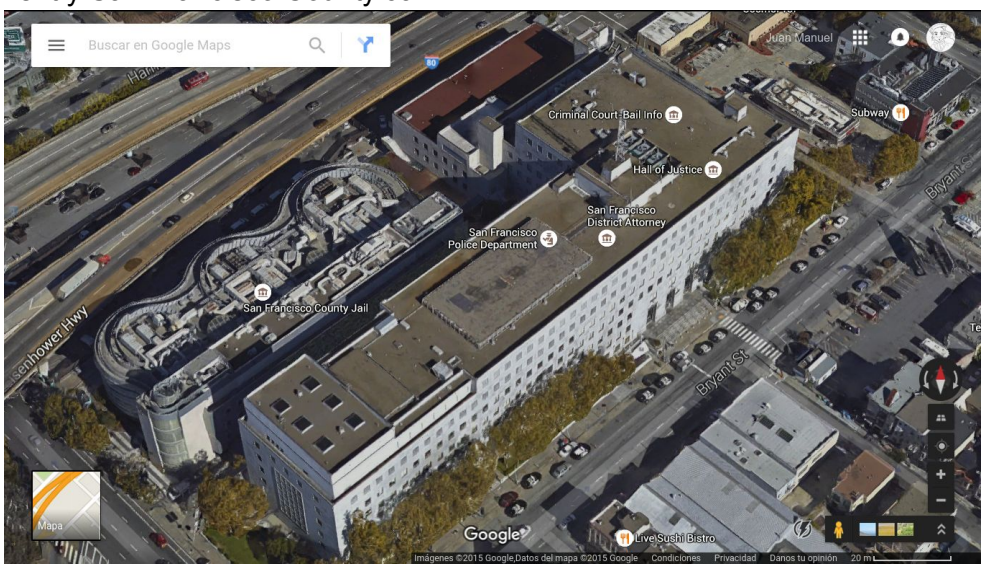
A partir de lo cual también notamos el patrón de que en navidad hay muchos menos crímenes que en el resto de los días del año.
Se repite la misma tendencia año a año.

Crímenes concentrados en un punto

La coordenada (X, Y) = (-122.403404791, 37.7754207067) concentra 26354 crímenes. Lo cual representa el 3% del set de datos (es bastante).
Para tener una mejor noción, la 2da coordenada con más crímenes apenas concentra unos 4000.

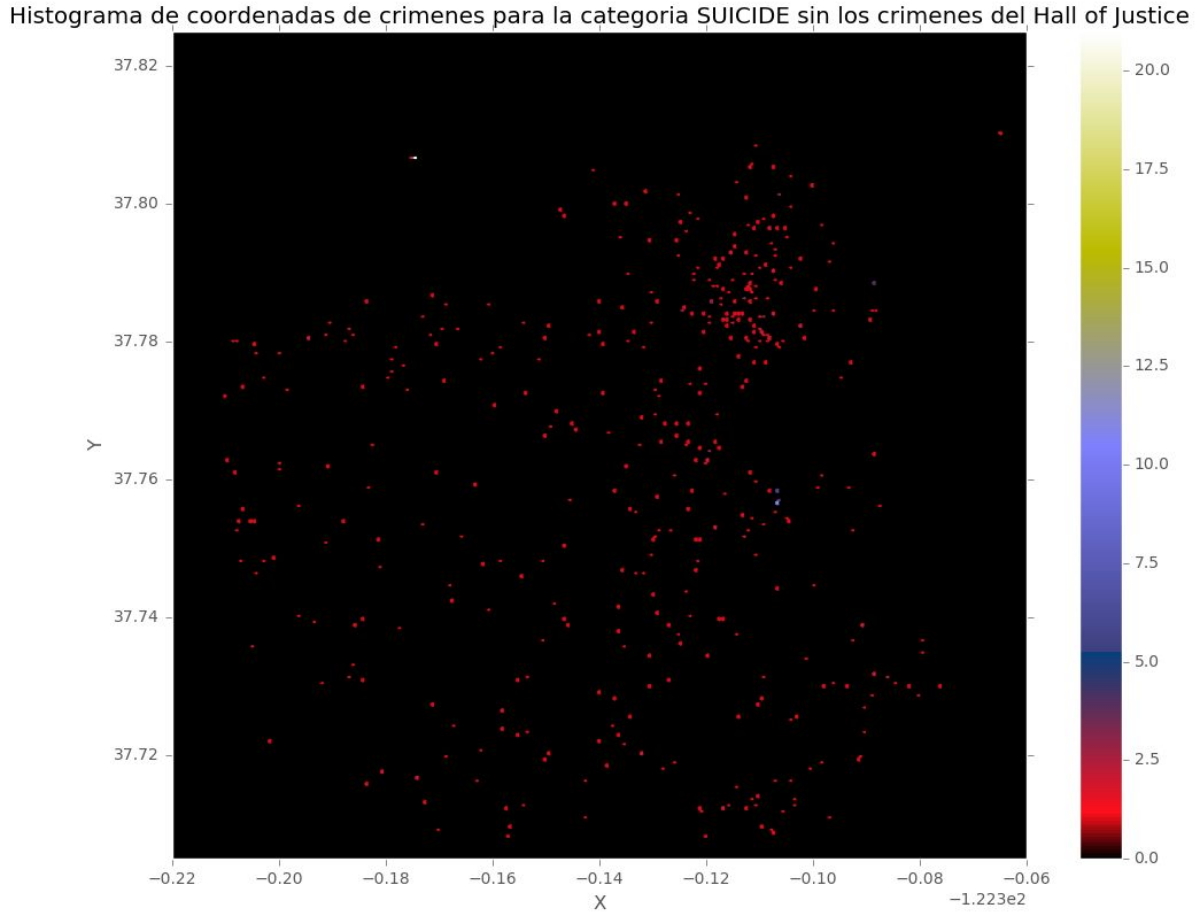


En la imagen aparece como un punto blanco.
Un poco de google maps apunta a que es el Hall of Justice, Police Department-Traffic Enforcement y San Francisco County Jail.



Golden Gate Bridge es un Puente de Suicidios

Analizando los suicidios en función de la coordenada nos topamos un lugar que concentraba muchos suicidios (unos 20), siendo que en el resto estaban absolutamente dispersos.



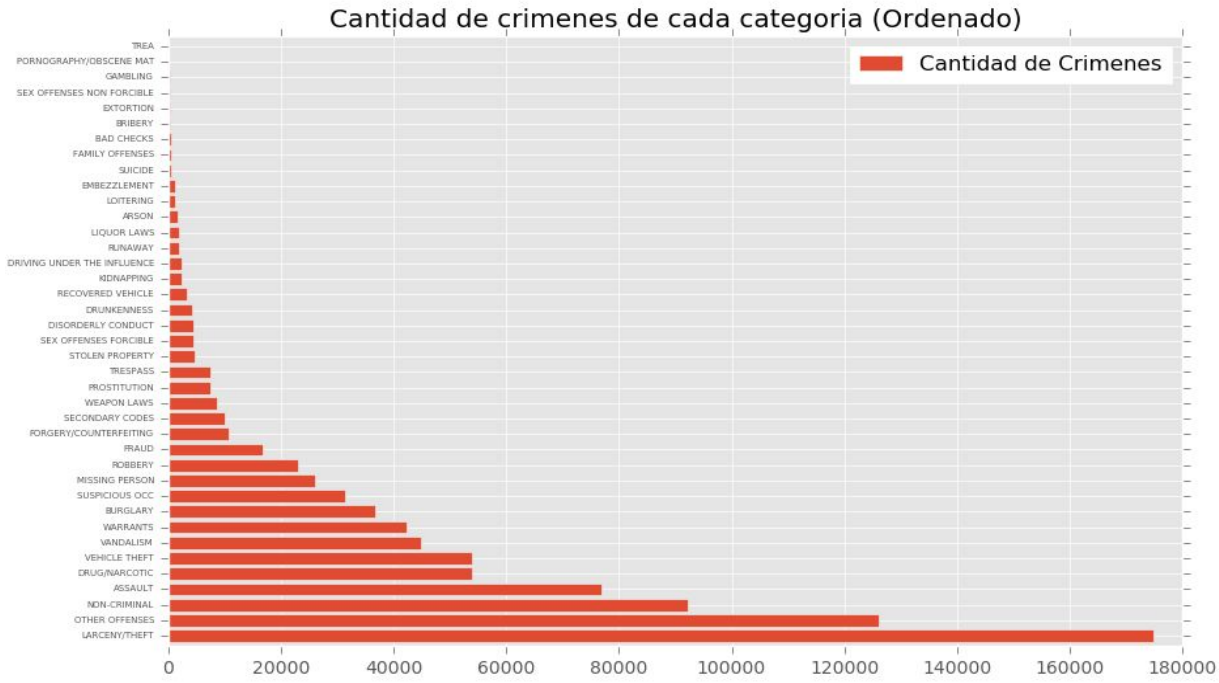
Averiguamos a donde apuntaban esas coordenadas y nos topamos con que coinciden con el Golden Gate Bridge.

Nos llamó la atención, googleamos un poco al respecto y encontramos esto:
https://en.wikipedia.org/wiki/Suicide_bridge

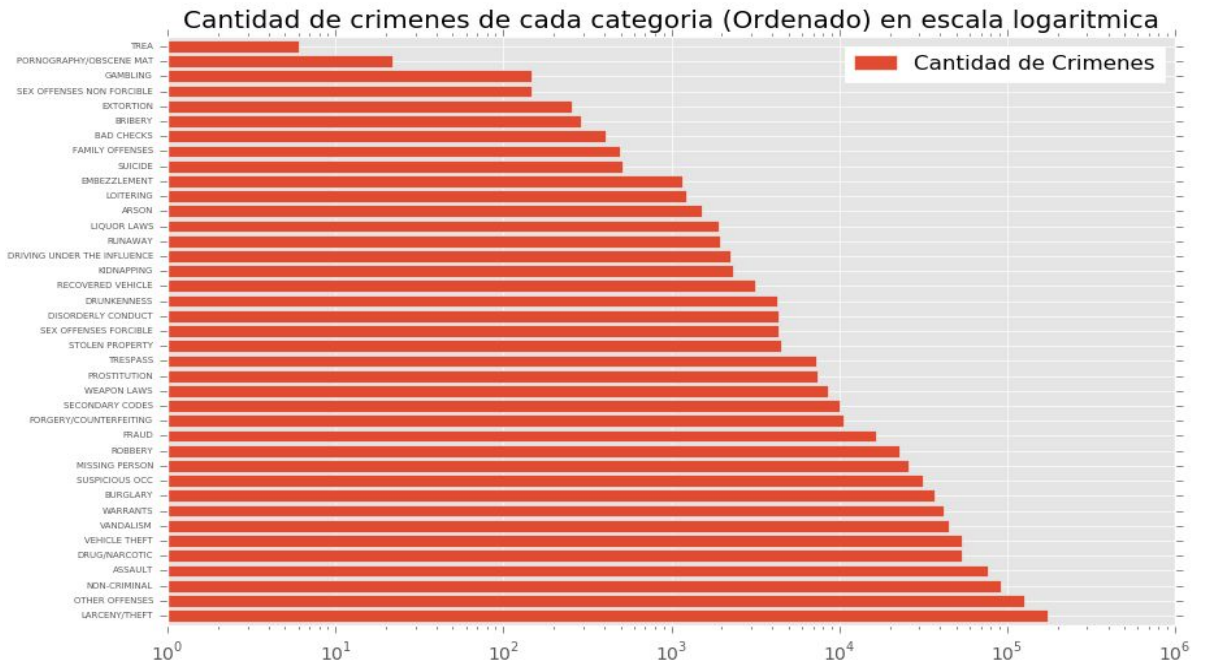
"The Golden Gate Bridge in San Francisco has the second highest number of suicides in the world, after the Nanjing Yangtze River Bridge."

¿La Cantidad de Crímenes en función de la categoría es exponencial?

Observando el gráfico de crímenes por categoría nos hicimos la pregunta de si la cantidad de crímenes por categoría se parece a una función de potencias. Ploteamos para ésto la cantidad de crímenes ordenada, tanto en escala lineal como logarítmica.



Vemos debajo que, salvo por las primeras dos categorías que tienen muy pocos crímenes, el gráfico se puede aproximar por una recta.
Esta información nos puede servir por ejemplo para hacer una aproximación en caso de tener que descartar datos: el Principio de Pareto indica que el 20% de las categorías serán responsables por el 80% de los casos.



Comentarios Adicionales

Resultó difícil para el grupo tomar una decisión sobre qué algoritmo utilizar para resolver este problema. La información encontrada en internet no propone ningún claro vencedor entre los distintos algoritmos (KNN, Perceptron, Decision Tree, etc.), y por lo tanto podríamos cambiar de elección en caso de hacer pruebas y encontrar que uno de ellos se comporta mucho mejor que el resto.

En cuanto a las características del trabajo práctico buena parte del grupo tiene grandes expectativas en cuanto a obtener resultados efectivos.
Creemos también que de todas las opciones que había en Kaggle el tópico de crímenes resultó ser una buena elección. Al menos a nosotros se nos hizo muy interesante lo que nos motiva a realizar las investigaciones pertinentes de forma comprometida. De hecho estamos aprendiendo un montón de cosas sobre la ciudad de San Francisco que antes de iniciar el TP ni sospechábamos.